Vers la conception automatique de filtres d'informations efficaces

Towards the Automatic Design of Efficient Custom Filters

Mathieu Stricker *,**, Frantz Vichot *, Gérard Dreyfus **, Francis Wolinski *

* Informatique-CDC – Groupe Caisse des Dépôts
Direction des Techniques Avancées
4, rue Berthollet
94114 Arcueil cedex – France
Phone : 33 1 40 49 15 69
Fax : 33 1 40 49 15 78
Email : mathieu.stricker@icdc.caissedesdepots.fr

** Ecole Supérieure de Physique et de Chimie Industrielles
Laboratoire d'Electronique
10, rue Vauquelin
75005 Paris – France

Abstract

The large amount of financial information released daily by press agencies makes the design of custom information filters, at a low development cost, an important issue. We present a comprehensive methodology for designing an information filter on a prescribed topic with a statistical approach. By using a neural network in conjunction with a search engine, we are able to automate to a large extent the construction of the training set. Since the performance of a filter depends strongly on the relevance of the selected inputs, we describe a method for performing feature selection. The filter is in routine use to filter news releases concerning company shareholding, with a precision and a recall of 90%.

Résumé

L'abondance d'information financière disponible rend nécessaire l'utilisation de filtres efficaces et facilement adaptables aux besoins de l'utilisateur. Cet article montre comment la mise en œuvre de méthodes statistiques permet de réduire le coût de fabrication de tels filtres, notamment en automatisant, en grande partie, la fabrication d'une base d'apprentissage. Comme les performances d'un filtre dépendent largement de la sélection de descripteurs, nous présentons une méthode pour effectuer cette sélection automatiquement. Notre filtre est utilisé en temps réel pour filtrer des dépêches relatives au thème des prises de participations, avec une précision et un rappel de l'ordre de 90 %.

Mots clés : classification, indexation par le contenu, réseaux de neurones, filtres, apprentissage, sélection de variables

1 Introduction

Afin d'exploiter efficacement le flux des informations disponibles, les utilisateurs ont besoin de filtrer celles-ci de manière à ne conserver que celles relatives à leurs activités, par exemple les prises de participations en capital, les taux d'intérêts, le bogue de l'an 2000… . Le filtrage de documents est un problème de classification à deux classes : il faut distinguer les documents pertinents des documents non pertinents. Un document est pertinent s'il traite du sujet que l'on souhaite filtrer. Cette définition implique donc la construction d'un filtre spécifique pour chaque thème.

La propriété d'apprentissage des réseaux de neurones, et leur capacité à s'adapter à la complexité des problèmes, ont déjà été exploitées par d'autres auteurs dans le domaine de la classification en général, et pour le filtrage de documents en particulier [Wiener *et al,* 1995; Schütze *et al*., 1995]. Pour être efficace, l'apprentissage des réseaux de neurones nécessite (i) de disposer d'une base de documents, préalablement étiquetés comme pertinents ou non pertinents, qui doit être suffisamment grande et représentative ; (ii) une procédure efficace pour sélectionner le plus petit ensemble de descripteurs discriminants.

Pour construire une base d'apprentissage de documents étiquetés, la procédure la plus simple consiste à classer chaque document manuellement. Mais, dans la pratique, cette méthode est quasiment inutilisable, car la base d'apprentissage est constituée de plusieurs milliers de documents, et cette opération doit être renouvelée pour chaque thème que l'on veut apprendre à filtrer. Pour résoudre ce problème, nous proposons la méthodologie suivante : une première base de documents étiquetés est extraite en utilisant un moteur de recherche classique sur une base d'archives couvrant plusieurs mois. Comme le moteur de recherche fait des erreurs de classification, nous estimons la probabilité de pertinence de chaque document afin de détecter les textes dont il faut vérifier manuellement la classe. Nous améliorons ainsi la qualité de la base.

Dans le domaine du traitement automatique de textes, le très grand nombre des descripteurs possibles (un descripteur par mot) constitue une difficulté pour la mise en œuvre de méthodes statistiques. En effet, les modèles statistiques font intervenir des paramètres, ajustables par apprentissage ; or le nombre de paramètres ajustables, pour tout modèle, neuronal ou pas, doit être aussi petit que possible ; comme ce nombre croît avec la dimension du vecteur d'entrée, on doit toujours s'efforcer de réduire celle-ci. Nous présentons une méthode qui permet de classer les mots par ordre de pertinence, et de choisir automatiquement le bon nombre de descripteurs.

Nous montrons comment ces différentes étapes sont utilisées pour construire une application opérationnelle qui permet le filtrage, en temps réel, des dépêches de l'AFP relatives au thème des participations des entreprises, avec une précision et un rappel de l'ordre de 90 %.

La section 2 décrit les travaux antérieurs relatifs au sujet du présent article. Dans la section 3, nous expliquons la méthodologie que nous proposons pour construire, à moindre coût, une base de documents étiquetés. La section 4 décrit la méthode de sélection de variables, et la section 5 présente les résultats expérimentaux. Une brève description de l'application opérationnelle fait l'objet de la section 6. Enfin, nous discutons les résultats obtenus.

2 Travaux antérieurs

Des travaux ont montré que filtrer les dépêches était possible et pouvait être utilisé dans un contexte opérationnel [Vichot *et al*., 1999]. Ces filtres sont fondés sur des systèmes de règles et ont démontré leur efficacité pour filtrer les documents selon des thèmes prédéfinis [Landau *et al*., 1993], mais leur réalisation nécessite un travail minutieux et coûteux étroitement lié à la langue et au thème abordé, ce qui les rend difficiles à transposer à d'autres thèmes. Pour rendre plus souple la conception de ces filtres, [Wiener *et al,* 1995; Schütze *et al*., 1995] ont utilisé des méthodes statistiques à base d'apprentissage, y compris des réseaux de neurones, sur des corpus de documents déjà étiquetés. Leurs différents résultats ont prouvé que ces méthodes conduisaient à des résultats satisfaisants et constituent donc des méthodes alternatives.

Pour construire rapidement une base d'apprentissage adaptée au thème que l'on cherche à apprendre, [Lewis et Gale, 1994] proposent une méthode itérative appelée *uncertainty sampling*. Ils commencent par construire un classifieur probabiliste à partir d'un très petit nombre d'exemples classés à la main. Ils utilisent ensuite ce classifieur sur le reste de la base non étiquetée, et ne classent manuellement que les documents pour lesquels le classifieur est incertain (probabilité autour de 0.5). Ils obtiennent ainsi une base d'apprentissage plus grande et ils itèrent l'opération. Cette méthode donne des résultats efficaces, mais elle nécessite néanmoins beaucoup de classement manuel. De plus, elle nécessite de refaire la sélection de descripteurs plusieurs fois, au fur et à mesure que la base grandit.

Quel que soit le modèle statistique utilisé dans le domaine du filtrage de textes, il est nécessaire d'effectuer une réduction de l'espace initial des descripteurs, celui-ci étant composé de tous les mots différents apparaissant dans un corpus (environ 500 000 mots sur notre corpus). Les performances des filtres dépendent en grande partie des mots retenus lors de la sélection de descripteurs, et du nombre de ces descripteurs [Lewis, 1992]. Ces méthodes reposent généralement sur des méthodes statistiques comme le calcul de l'information mutuelle moyenne et du χ^2 [Wiener *et al,* 1995] ou sur la méthode *latent semantic indexing* [Deerwester *et al.,* 1990]. [Yang et Pedersen, 1997] ont proposé une comparaison intensive de ces différentes approches. Cependant, ils font précéder leurs sélections d'un pré-traitement qui consiste à ne pas considérer les descripteurs qui appartiennent à une liste de mots grammaticaux fréquents (stop words). Or

l'établissement d'une telle liste est délicate, et elle est évidemment liée à la langue utilisée. Il est donc préférable d'utiliser une méthode qui ne nécessite pas l'utilisation d'une telle liste.

3 Construire une base d'apprentissage

Toutes les méthodes qui reposent sur une estimation de paramètres (apprentissage), nécessitent une base de documents étiquetés représentatifs des classes que l'on cherche à discriminer. Des corpus de documents préalablement étiquetés existent (par exemple les corpus utilisés par la conférence TREC [Voorhees and Harman, 1999]). Ils sont utiles afin de comparer les différentes méthodes entre elles, mais ils ne peuvent pas être utilisés pour construire des applications réelles puisque les bases disponibles ne sont pas nécessairement représentatives des classes que l'on cherche à apprendre. Nous proposons dans cette section une méthodologie pour construire une base représentative de documents étiquetés en minimisant le nombre de vérifications manuelles.

Un processus itératif

La construction de la base d'apprentissage que nous proposons nécessite l'estimation, pour chaque document, de la probabilité d'être pertinent relativement au thème que l'on souhaite apprendre à filtrer. Cette probabilité peut être calculée avantageusement avec des réseaux de neurones (voir par exemple [Bishop, 1995]).

L'étape préliminaire de la méthode consiste à extraire, d'une base d'archive couvrant plusieurs mois, un ensemble de documents pertinents et un ensemble de documents non pertinents relativement au thème que l'on veut apprendre à filtrer. Les documents pertinents sont obtenus en utilisant un moteur de recherche commercial et plusieurs requêtes booléennes. Chacune de ces requêtes est construite pour être précise et peu couvrante. Le moteur de recherche utilise des formes défléchies, si bien que la sélection effectuée ne se limite pas uniquement aux requêtes. Les documents non pertinents sont extraits en sélectionnant au hasard des documents dans l'archive après élimination des documents sélectionnés par les requêtes précédentes.

Cette étape initiale permet d'obtenir une première base, qui comporte des erreurs de classification, dues, d'une part, au moteur de recherche, et, d'autre part, au fait que, parmi les dépêches sélectionnées au hasard, certaines sont en fait pertinentes car le moteur de recherche n'a pas extrait toutes les dépêches pertinentes contenues dans l'archive.

La qualité de l'étiquetage de la base est améliorée itérativement comme suit : un réseau de neurones apprend sur cette base, puis estime la probabilité de pertinence de chaque document. Seules sont vérifiées manuellement les dépêches pour lesquelles le réseau de neurones et le moteur de recherche sont en désaccord, c'est-à-dire les dépêches indexées comme pertinentes par le moteur de recherche mais dont l'estimation de la probabilité de pertinence est proche de 0, et les dépêches indexées comme non pertinentes avec une probabilité de pertinence proche de 1.

On renouvelle apprentissage en tenant compte des modifications effectuées, et l'on vérifie à nouveau la classe des documents selon le même critère. Le système garde la trace des documents qui ont été vérifiés manuellement lors des itérations précédentes. Après quelques itérations, il n'existe plus de documents pour lesquels le réseau de neurones et le moteur de recherche sont en désaccord et qui n'ont jamais été vérifiés manuellement.

Comme le nombre de vérifications est limité à quelques centaines pour une base de plusieurs milliers de dépêches, cette méthode permet de construire des bases de taille importante en limitant le coût de fabrication.

Cette méthode nécessite de nouveaux apprentissages à chaque itération ; néanmoins, les réseaux utilisés sont de très petite taille (voire, se réduisent à des séparateurs linéaires), et possèdent peu de descripteurs d'entrée grâce à l'utilisation de la méthode de sélection décrite dans le paragraphe suivant ; en conséquence, les temps de calcul nécessaires ne sont aucunement rédhibitoires.

4 Sélection de descripteurs

Chaque dépêche est représentée par un vecteur de fréquences des mots qui la composent. Si l'on considère l'ensemble des dépêches, le nombre de mots différents est très grand. Il est donc nécessaire de n'utiliser que les fréquences de certains mots choisis en fonction du thème que l'on cherche à filtrer.

Dans la suite, nous appelons *descripteurs* les mots choisis pour représenter les textes.

La sélection de descripteurs est une étape critique de la constitution du filtre, car elle conditionne ses performances quel que soit le modèle utilisé ultérieurement. Cette procédure doit choisir automatiquement, à partir des exemples de l'ensemble d'apprentissage, la définition et la taille du vecteur de descripteurs optimal qu'il faut retenir pour représenter chaque texte.

Pour choisir les descripteurs, on cherche à supprimer les descripteurs très fréquents sur le corpus, car on considère qu'ils n'apportent pas d'information (par exemple les mots *le*, *la*, *les*, …). On cherche également à supprimer les mots très rares, car ils ne sont pas exploitables d'un point de vue statistique.

Plus le nombre de descripteurs est élevé, plus le nombre d'exemples nécessaires à une bonne estimation des paramètres du modèle est important. Cependant, plus on supprime de descripteurs, plus on perd de l'information. La procédure de sélection de descripteurs doit donc trouver un compromis entre ces exigences.

Nous présentons dans la suite une méthode de classement des descripteurs par ordre d'importance décroissante, couplée avec une méthode qui permet de choisir le bon nombre de descripteurs sans avoir à effectuer d'apprentissages.

4.1 Réduction de l'espace initial

Il est nécessaire de faire précéder la classification de descripteurs d'une première réduction de la dimension de l'espace des descripteurs, pour n'avoir plus que quelques centaines de candidats. Cette première sélection permet de définir un ensemble de mots représentant le vocabulaire lié au thème que l'on cherche à filtrer.

Pour chaque mot d'un texte, on calcule sa fréquence divisée par sa fréquence totale dans le corpus. Les mots du texte sont ensuite classés par valeurs décroissantes de ces ratios, et seuls les mots du haut de la liste sont conservés. Ainsi, tous les mots très fréquents du corpus sont éliminés.

Après avoir effectué cette opération sur l'ensemble des textes pertinents, on calcule, sur cet ensemble, la fréquence de chaque mot. Un nouveau classement par ordre décroissant est effectué afin de placer les mots les plus rares en fin de liste.

Cette première étape permet de supprimer les mots grammaticaux fréquents, sans l'utilisation d'une liste annexe, et elle permet de supprimer les mots rares. Les descripteurs sélectionnés représentent en fait le vocabulaire du domaine, mais ne sont pas tous nécessairement discriminants ; de plus, certains peuvent être très corrélés.

4.2 Classement des descripteurs

Nous utilisons la méthode d'orthogonalisation de Gram-Schmidt [Chen *et al.*, 1989] pour classer les descripteurs restants par ordre de pertinence décroissante. La méthode peut être décrite de la manière suivante.

On considère les Q descripteurs candidats et un ensemble d'apprentissage contenant N exemples avec leurs Q descripteurs possibles et la sortie associée (la sortie est ici la classe à laquelle appartient l'exemple). On note $\boldsymbol{tf}^i = {}^{\mathrm{T}}[tf_1^i, tf_2^i, ..., tf_N^i]$ le vecteur des fréquences pour le descripteur i et l'on note $\boldsymbol{y}_p$ le vecteur de dimension N contenant les sorties à modéliser. On considère la matrice (N, Q) $X = [\boldsymbol{tf}^1, \boldsymbol{tf}^2, ..., \boldsymbol{tf}^Q]$; le modèle peut être écrit $\boldsymbol{y} = X\theta$, où θ est le vecteur de paramètres du modèle.

À la première itération, il faut trouver le vecteur de descripteurs $\boldsymbol{tf}^i$ qui "explique" le mieux la sortie. Pour cela, on calcule le carré du cosinus des angles entre le vecteur de sortie et les vecteurs de descripteurs :

$$\cos^2\left(\boldsymbol{tf}^k, \boldsymbol{y}_p\right) = \frac{\left({}^{\mathrm{T}}\boldsymbol{tf}^k \boldsymbol{y}_p\right)^2}{\left({}^{\mathrm{T}}tf^k \boldsymbol{tf}^k\right)\left({}^{\mathrm{T}}\boldsymbol{y}_p \boldsymbol{y}_p\right)}, k = 1, Q$$

Le vecteur sélectionné est celui pour lequel cette quantité est maximale. La contribution du descripteur sélectionné est ensuite éliminée en projetant le vecteur de sortie et tous les vecteurs restants, sur le sous-espace orthogonal au vecteur sélectionné.

Dans ce sous-espace, le descripteur qui "explique" le mieux la sortie projetée est sélectionné et les Q-2 descripteurs restants ainsi que la sortie sont projetés sur le sous-espace orthogonal au sous-espace formé par les deux premiers vecteurs sélectionnés.

Cette méthode a été appliquée avec succès pour la modélisation dans [Duprat *et al.,* 1998] et pour la classification dans [Stoppiglia, 1997] [Oukhellou *et al.*, 1998]. D'autres méthodes de sélectin de variables peuvent être utilisées (voir par exemple Cibas *et al.,* 1996].

4.3 Critère d'arrêt

Une fois que les descripteurs ont été classés par la procédure expliquée ci-dessus, il faut décider du nombre de descripteurs que l'on conserve. Cette opération est effectuée en utilisant un vecteur aléatoire qui est classé par la méthode de Gram-Schmidt exactement comme les autres descripteurs. Les descripteurs classés après ce vecteur aléatoire sont considérés comme non pertinents pour le problème posé.

Dans la pratique, le rang de ce vecteur aléatoire est en fait un nombre aléatoire. Il faut donc calculer la fonction de distribution de probabilité de l'angle entre un vecteur aléatoire et le vecteur de sortie. Le calcul de la probabilité qu'un vecteur aléatoire soit plus pertinent que l'un des n descripteurs sélectionnés après n itérations a été développé dans [Stoppiglia, 1997]. Chaque fois qu'un descripteur est sélectionné, on calcule la probabilité qu'un descripteur aléatoire soit plus pertinent que ce descripteur. Au-delà d'un seuil prédéfini (typiquement 1 % ou 5 %), tous les descripteurs sont éliminés.

4.4 Factorisation des mots

La méthode de sélection de descripteurs précédente est une méthode générale applicable à différentes catégories de problèmes. Le langage naturel présente une spécificité dont nous n'avons pas tiré parti : la flexion des mots. Le français est, en effet, une langue fortement fléchie (par exemple, le verbe *détenir* existe sous plus de 20 formes conjuguées) ; or notre réseau considère les différentes formes d'un même mot comme des mots différents. Par exemple, la forme *détenue* du verbe *détenir* est prise en considération par le réseau alors que la forme *détenues*, qui est beaucoup plus rare sur notre corpus, ne l'est pas.

Une idée intuitive consiste à considérer les différentes formes d'un mot comme un seul descripteur de façon à obtenir de meilleures performances. Cependant, cette opération peut n'avoir pas de sens pour la classification [Jing et Tzoukermann, 1999]. Ainsi le mot *action* au singulier et au pluriel a souvent un sens différent selon le contexte. Par exemples, la phrase "Le jugement est plus nuancé selon le domaine d'*action* du gouvernement." n'est pas liée au thème des participations alors que la phrase "Den Danske Bank a acquis en décembre dernier 90 % des *actions* de Fokus Bank." est une phrase qui traite des participations. Si nous factorisons le mot *action*, nous risquons donc de sélectionner une dépêche qui contient la première phrase.

Nous avons donc choisi de factoriser les formes d'un mot qui ont la même influence discriminante. On dit que deux formes ont la même influence discriminante si elles influencent le classifieur vers la même décision, c'est-à-dire si leurs coefficients de régression linéaire sont de même signe ; or la procédure d'orthogonalisation de

Gram-Schmidt permet d'estimer ce coefficient pour chaque mot, donc les calculs effectués pour la sélection des mots fournissent directement l'information recherchée.

5 Résultats expérimentaux

5.1 Caractéristique de la base créée

Sur le thème des prises de participations, la méthode décrite au paragraphe 3 a permis d'obtenir une base d'apprentissage contenant environ 10000 dépêches dont 2000 dépêches pertinentes. Cette base est divisée en (i) une base d'apprentissage de 8000 exemples, (ii) une base de validation (1000 exemples) utilisée pendant l'apprentissage, (iii) une base de test destinée à la mesure des performances du filtre. Dans chacun des ensembles, les dépêches sur le sujet représentent environ 20 % des dépêches.

5.2 Mesure de la performance

Dans notre application, le filtre doit sélectionner les documents qu'il estime être pertinent pour les présenter à l'utilisateur. Parmi les documents sélectionnés, certains sont effectivement pertinents, d'autres ne le sont pas. On définit alors 4 grandeurs :

a = nombre de documents pertinents sélectionnés.
b = nombre de documents non pertinents sélectionnés.
c = nombre de documents pertinents non sélectionnés.
d = nombre de documents non pertinents non sélectionnés.

On calcule alors la précision et le rappel d'un filtre par :

Précision (P) = $a/(a+b)$.
Rappel (R) = $a/(a+c)$.

Ces deux notions sont liées puisqu'il est facile d'obtenir une précision élevée si le rappel est très faible et il est, de même, facile d'obtenir un rappel élevé au prix d'une précision très faible. Un filtre doit donc être caractérisé à la fois par son rappel et sa précision.

Pour faciliter les comparaisons entre différents filtres, on définit la mesure F [Lewis et Gale, 1994] qui prend en considération les deux indicateurs :

$$F = \frac{2PR}{P+R}$$

Cette mesure donne une importance égale à la précision et au rappel, et, si la précision est égale au rappel, alors $F = P = R$.

Dans notre application, on cherche en général un compromis entre la précision et le rappel sans chercher à favoriser un indicateur par rapport à l'autre. La précision est en effet directement accessible à l'utilisateur, car, s'il lit trop de dépêches non pertinentes, il n'utilisera plus le filtre. D'autre part, s'il ne trouve pas des informations dont il a eu connaissance par ailleurs, sa confiance dans le filtre diminue.

On caractérise donc un filtre par la mesure F dans la suite, puisque cette mesure rend compte de ce que l'utilisateur percevra.

Comme notre classifieur est un classifieur probabiliste, il estime pour chaque document une probabilité de pertinence, et la décision de sélectionner ou non ce document est prise selon un seuil de décision. Pour chaque valeur du seuil de décision, il est possible de calculer une valeur de la précision et une valeur de rappel, donc une valeur de F. Le seuil de décision est choisi pour optimiser cette valeur sur la base d'apprentissage ; la valeur de F qui caractérise la performance du modèle est calculée, avec le seuil ainsi déterminé, à l'aide de la base de test.

5.3 Sélection de descripteurs

Nous appliquons d'abord la première étape de notre sélection de descripteurs sur la base d'apprentissage relative aux participations. Cette étape consiste à supprimer les descripteurs les plus fréquents et les moins fréquents. La Figure 1 montre les performances d'un classifieur linéaire sur la base de test en fonction du nombre de descripteurs retenus, classés par ratio décroissant.

Cette première expérience montre donc que les performances du filtre dépendent du nombre de descripteurs choisi, et que, au-delà d'une certaine limite, les performances n'augmentent plus et peuvent même diminuer.

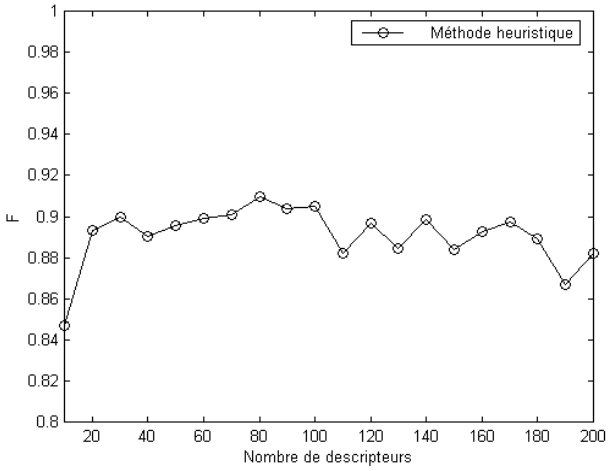

Figure 1 : Evolution des performances en fonction du nombre de descripteurs retenus pour le modèle.

La Figure 2 illustre l'amélioration apportée par rapport à la première étape de sélection. L'ordre des descripteurs proposé par la méthode de Gram-Schmidt est meilleur que celui proposé par la méthode heuristique décrite plus haut puisque, à nombre de descripteurs égal, les performances sont plus élevées. Le critère d'arrêt fixé par le descripteur aléatoire appliqué avec un risque de 1 % sélectionne les 30 premiers descripteurs Ce critère d'arrêt permet de choisir la taille du meilleur sous-ensemble car au-delà les performances n'augmentent plus que de façon négligeable.

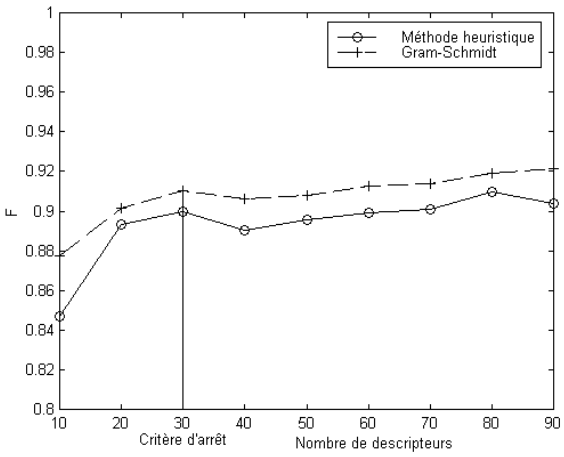

Figure 2 : Résultats de la sélection des descripteurs.

La méthode de sélection de descripteurs a donc permis de réduire l'espace des descripteurs de 500000 mots possibles à 30 mots.

5.4 Apprentissage des réseaux de neurones

Le filtre est réalisé par un réseau de neurones qui est utilisé pour estimer la probabilité qu'une dépêche soit pertinente relativement au thème défini par la base d'apprentissage. Les entrées du réseau sont les descripteurs sélectionnés lors de la phase précédente.
Les réseaux utilisés sont des réseaux à couches avec deux couches de poids, tous les neurones utilisent une fonction d'activation sigmoïde.
Si l'on appelle x_i l'entrée du réseau correspondant au descripteur i et ω_i l'occurrence de ce descripteur dans la dépêche alors :

$$x_i = \begin{cases} -1 \text{ si } \omega_i = 0 \\ \dfrac{\omega_i}{\max(\omega_i)} \text{ si } \omega_i > 0 \end{cases}$$

Max(ω_i) est l'occurrence maximale du descripteur i dans l'ensemble de la base d'apprentissage. Ce codage permet de s'assurer que les entrées sont dans l'intervalle [-1;+1].
L'apprentissage est effectué en minimisant l'erreur quadratique sur l'ensemble d'apprentissage entre la valeur estimée par le réseau de neurones et la sortie désirée. Le gradient de la fonction de coût est calculé sur l'ensemble des exemples par rétropropagation. La minimisation se fait à l'aide de la méthode de quasi-Newton avec l'algorithme BFGS [Bishop, 1995].

Choix de la meilleure architecture
Le choix de l'architecture (i.e. le nombre de neurones cachés) est effectué en réalisant plusieurs apprentissages avec un nombre de neurones cachés croissants, et en optimisant la valeur de F sur la base de validation. Une fois l'architecture choisie, les performances sont estimées sur la base de test qui n'a jamais été utilisée jusqu'ici avec le seuil optimisé sur la base d'apprentissage.

Résultats avec les 30 descripteurs
La meilleure valeur de F sur la base de validation est obtenue avec 2 neurones cachés (93.7 %) ; cependant avec un neurone de sortie à fonction d'activation sigmoïde, sans neurone caché (régression logistique), la performance est de 93.3 %. Comme l'amélioration est négligeable, le modèle le plus parcimonieux est sélectionné.
Les performances du réseau sont alors mesurées sur la base de test **: F = 90.0 %** ce qui correspond à une précision de 91.4 % et un rappel de 88.7 %**.**
La figure 3 montre la courbe rappel-précision obtenue sur la base de test en fonction du seuil de décision.

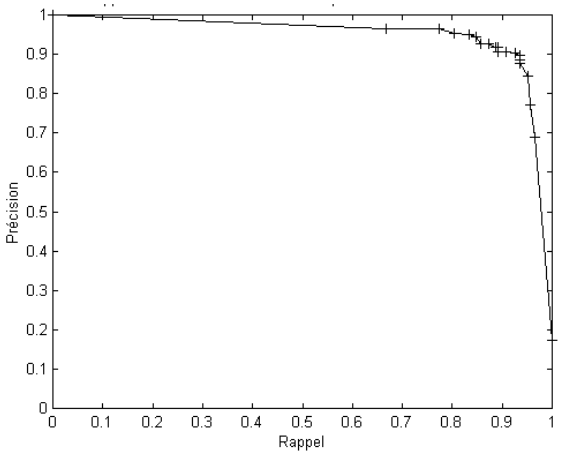

Figure 3 : Courbe Rappel-Précsion sur la base de test. Chaque croix correspond à un seuil différent.

Cette figure montre qu'il faut choisir, pour le fonctionnement opérationnel du filtre, un compromis entre le rappel et la précision. Le choix de ce compromis ce fait en fixant le seuil de décision du filtre de façon a optimiser la valeur de F.

Influence de la factorisation
On effectue une factorisation de certains mots selon les critères indiqués dans la section 4.4. Nous obtenons alors 23 descripteurs, certains d'entre eux représentant plusieurs mots. On ajoute notamment pour les verbes leurs formes fléchies courantes.
Plusieurs architectures sont à nouveau testées et comme précédemment, les meilleurs résultats sont obtenus avec un neurone sigmoïde. La performance sur la base de validation est F= 92.6 % et sur la base de test ***F* = 90.6 %**, qui correspond à une précision de 89.1 % et un rappel de 92.1 %.
Les résultats sont donc comparables à ceux qui sont obtenus avec les 30 descripteurs. Nous commenterons ce point lors de la discussion.

5.5 Difficulté du thème

Pour avoir une idée de la difficulté de ce thème, nous reprenons le critère exposé dans [Wiener *et al,* 1995] : nous calculons les performances du filtre obtenu avec uniquement le meilleur descripteur comme entrée du modèle (le mot *capital*). Sur la base de test, un tel filtre a

une précision de 85.7 %, et un rappel de 64.7 %, soit F = 73.7 %. Le thème choisi n'est donc pas trivial.

6 Une application opérationnelle

Le réseau est intégré dans une application utilisée quotidiennement par les gestionnaires de portefeuilles d'actions. Les dépêches relatives au thème des participations sont sélectionnées en temps réel. Cette application est implémentée dans une structure multi-agents [Wolinski *et al.*, 1998]. Lorsqu'une dépêche arrive, un premier agent la transforme en vecteur de descripteurs, puis un second agent constitué par le classifieur calcule la probabilité que la dépêche soit pertinente. Si cette probabilité dépasse un seuil de décision fixé, alors le titre de la dépêche apparaît en lien hypertexte renvoyant au texte de la dépêche comme le montre la Figure 4.

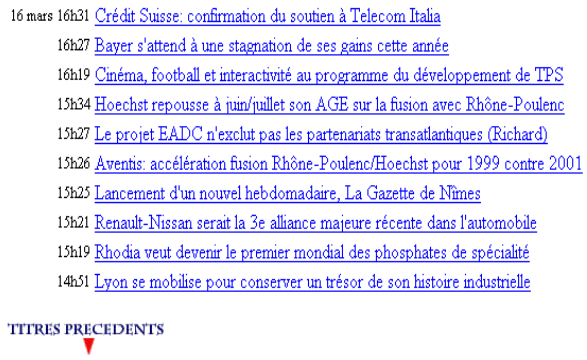

Figure 4 : Exemple de dépêches sélectionnées sur le thème participations

Comme nous l'avons mentionné plus haut, il faut fixer un seuil de décision pour le réseau. La valeur de ce seuil détermine la précision et le rappel de ce filtre. Or les probabilités *a priori* d'appartenance à l'une ou l'autre classes ne sont pas identiques sur la base d'apprentissage et sur le flux réel ; il n'est donc pas possible d'utiliser le seuil optimal trouvé lors des expériences précédentes. Il est néanmoins possible d'effectuer une correction pour tenir compte du fait que les caractéristiques du flux réel de dépêches sont différentes de celles de la base d'apprentissage.

En effet, désignons par C_1 la classe des dépêches pertinentes, et par C_2 son complémentaire ; alors la sortie S du réseau de neurones peut être interprétée comme la probabilité *a posteriori* d'appartenir à la classe C_1 connaissant le vecteur d'entrée x, qui est liée à la probabilité *a priori* des classes par la formule de Bayes :

$$S = P(C_1|x) = \frac{p(x|C_1)P_1}{p(x|C_1)P_1 + p(x|C_2)P_2} \quad (1)$$

P_1 et P_2 sont les probabilités *a priori* sur la base d'apprentissage d'appartenir aux classes C_1 et C_2 respectivement, et où $p(x \mid C_1)$ est la densité de probabilité du vecteur x dans la classe C_1.

Si les probabilités a priori sur le flux réel sont maintenant P'_1 et P'_2, alors il faut estimer S' qui vaut

$$S' = P'(C_1|x) = \frac{p(x|C_1)P'_1}{p(x|C_1)P'_1 + p(x|C_2)P'_2} \quad (2)$$

Or en utilisant (1), on trouve que la probabilité S' s'exprime à partir de S :

$$S' = \frac{P'_1}{P_1} S \frac{1}{\frac{P'_1}{P_1} S + \frac{P'_2}{P_2}(1 - S)} \quad (3)$$

Connaissant le seuil optimal sur la base de test, on peut donc calculer le seuil équivalent sur le flux réel grâce à (3).

7 Discussion

Les résultats que nous avons présentés peuvent être discutés sous trois aspects : la structure du réseau de neurones classifieur, l'influence de la factorisation, et celle de la taille de la base.

7.1 Structure du réseau de neurones

Le modèle utilisé finalement pour notre filtre est un neurone sigmoïde, c'est-à-dire une régression logistique correspondant à une séparation linéaire. Cependant l'approche neuronale reste intéressante ; en effet dans l'exemple présenté, on a montré que l'on n'obtient pas de meilleures performances avec un modèle non linéaire. Néanmoins, un nouveau thème peut nécessiter un modèle plus compliqué : l'approche neuronale sera alors à même de fournir un modèle bien adapté.

7.2 Influence de la factorisation

La factorisation des mots n'améliore les performances globales du filtre que de façon négligeable, malgré les précautions expliquées dans la section 4. Or, dans notre cas, on s'attend à ce que la factorisation augmente le rappel sans trop diminuer la précision puisque peu de mots sont factorisés. En effet, certaines dépêches qui contenaient une forme fléchie d'un verbe non présent dans la liste des mots initiaux (par exemple la forme *détenues*) ont une probabilité de pertinence beaucoup plus élevée lorsque l'on utilise le classifieur avec descripteurs factorisés. Cependant les dépêches pour lesquelles la factorisation ne modifie pas le vecteur des entrées, peuvent voir leur probabilité de pertinence diminuer : par exemple, la forme *détenue* a un poids de 2.24 dans le classifieur sans descripteur factorisé ; or le poids pour toutes les formes du verbe *détenir* est 1.86 avec le classifieur avec descripteurs factorisés. Par conséquent, une dépêche qui ne contient que la forme *détenue* voit sa probabilité de pertinence diminuer si le verbe *détenir* est factorisé. Pour ces dépêches, la probabilité de pertinence peut alors devenir inférieure au seuil de décision, et dans ce cas, le rappel diminue. Ces raisons font que le rappel n'augmente pas autant qu'on pouvait le penser. Comme de plus la précision diminue

légèrement, car de nouvelles dépêches non pertinentes sont sélectionnées, la performance globale mesurée par la valeur de F n'augmente que très peu et peut même diminuer légèrement.
Pour éviter cette diminution de rappel, on peut utiliser le classifieur sans descripteur factorisé lorsque la factorisation n'a amené aucun nouveau mot et utiliser le classifieur avec les descripteurs factorisés dans le second cas. On obtient alors sur la base de test F = 91.5 % qui correspond à une précision de 88.2 % et à un rappel de 95.0 %. Dans ce cas, l'utilisation des deux réseaux permet d'améliorer significativement le rappel sans trop diminuer la précision, si bien que la performance globale du filtre mesurée par la valeur de F augmente.

7.3 Influence de la taille de la base

L'essentiel du coût de la méthode présentée dans cet article réside dans la fabrication semi-automatique de la base d'apprentissage. L'évolution des performances du filtre sur la base de test en fonction de la taille de la base d'apprentissage est reportée Figure 5. Cette expérience montre qu'à partir d'un millier d'exemples de chaque classe environ, les performances du filtre n'augmentent plus.

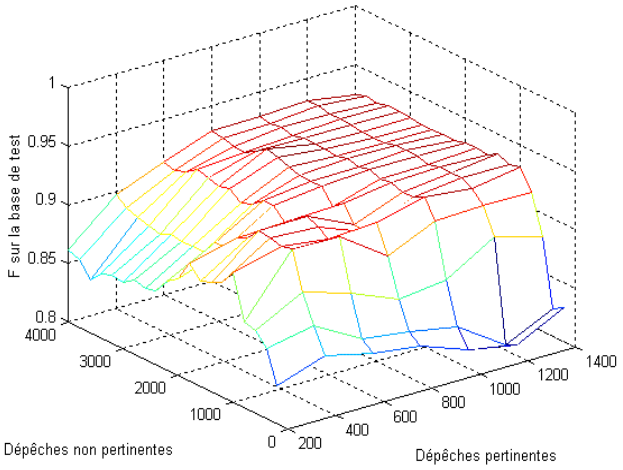

Figure 5 : Performance sur la base de test en fonction de la taille et de la répartition de la base d'apprentissage

Une partie de la base n'apporte donc plus d'informations ; certains reclassements faits à la main étaient donc peut-être inutiles. [Cohn *et al.*, 1994] proposent une méthode appelée active learning qui permet de sélectionner uniquement les exemples dont le réseau a besoin pour mieux apprendre. En associant cette méthode avec la nôtre, on ne pourrait reclasser que les dépêches effectivement utiles pour l'apprentissage du réseau et donc limiter le nombre d'exemples nécessaires.

8 Conclusion

Nous avons montré dans cet article comment réaliser un filtre d'information sur mesure pour un thème quelconque. Notre méthode permet de fabriquer, à moindre coût, une base d'apprentissage de taille importante, adaptée au thème que l'on cherche à apprendre à filtrer. Les deux points clés de la méthode sont :

- l'utilisation d'un moteur de recherche conventionnel pour constituer une base de données initiale, couplé à un réseau de neurones classifieur pour diminuer le taux d'erreur d'étiquetage de cette base ;
- l'utilisation d'une méthode de sélection de descripteurs entièrement automatique dont le critère d'arrêt permet une optimisation efficace de la dimension du vecteur des entrées du classifieur.

Remerciements
Nous tenons à remercier Guillaume Euvrard pour ses conseils et pour la relecture de cet article.

Bibliographie

[Bishop, 1995] C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[Chen *et al.*, 1989] S. Chen, S. A. Billings and W. Luo. Orthogonal least squares methods and their application to non-linear system identification. *International Journal of Control*, Vol. 50, n°5, 1873--1896, 1989.

[Cibas *et al.*, 1996] T. Cibas, F. Fogelman-Soulié, P. Gallinari, S. Raudys. Variable Selection with Neural Networks. *Neurocomputing,* Vol. 12, 223-248, 1996.

[Cohn *et al.*, 1994] D. Cohn, L. Atlas, R. Ladner. Improving Generalization with Active Learning. *Machine Learning*, 5(2), 201--221, 1994.

[Deerwester *et al.,* 1990] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, R. Harshman. Indexing by Latent Semantic Indexing. *Journal of the American Society for Information Science*, 41(6), 391--407, 1990.

[Duprat *et al.*, 1998] A. Duprat, T. Huynh, G. Dreyfus. Towards a Principled Methodology for Neural Network Design and Performance Evaluation in QSAR ; Application to the Prediction of LogP. *J. Chem. Inf. Comp. Sci.*, 38, 586--594, 1998.

[Jing et Tzoukermann, 1999] H. Jing, E. Tzoukermann. Information Retrieval Based on Context Distance and Morphology. *To appear in the Proceedings of SIGIR'99*. 1999.

[Landau *et al* , 1993] M.-C. Landau, *F*. Sillion, *F*. Vichot. Exoseme : a Document Filtering System Based on Conceptual Graphs. *Proceedings of ICCS*, Quebec, Canada, 1993.

[Lewis, 1992] D. D. Lewis. Feature Selection and Feature Extraction for Text Categorization. *Proceedings of Speech and Natural Language Workshop*. DARPA, Morgan Kaufmann, 212--217, 1992.

[Lewis et Gale, 1994] D. D. Lewis, W. A. Gale. A Sequential Algorithm for Training Text Classifiers.

Proceedings of the 17th Annual International ACM/SIGIR Conference, 3--12, 1994.

[Oukhellou *et al.*, 1998] L. Oukhellou, P. Aknin, H. Stoppiglia, G. Dreyfus. A New decision Criterion for Feature Selection: Application to the Classification of Non Destructive Testing Signatures. *European SIgnal Processing COnference (EUSIPCO'98),* 1998.

[Schütze *et al*., 1995] H. Schütze, D. A. Hull, J. O. Pedersen. A Comparison of Classifiers and Document Representations for the Routing Problem. *Proceedings of the 18th Annual ACM SIGIR Conference*, 229--337, 1995.

[Stoppiglia, 1997] H. Stoppiglia. Méthodes statistiques de sélection de modèles neuronaux; applications financières et bancaires. Thèse de l'université Paris VI, 1997.

[Vichot *et al*., 1999] F. Vichot, F. Wolinski, H.-C. Ferri, D. Urbani. Using Information Extraction for Knowledge Entering. *Advances in intelligent systems: concepts, tools and applications.* S. Tzafestas Ed., Kluwer academic publishers, 1999.

[Voorhees et Harman, 1999]. Ellen M. Voorhees and Donna Harman. Overview of the 7th Text Retrieval Conference (TREC-7). In *The Seventh Text Retrieval Conference (TREC-7)*, *NIST Special Publication 500-242*, 1999.

[Wiener *et al,* 1995] E. Wiener, J. O. Pedersen, A. S. Weigend. A Neural Network Approach to Topic Spotting. *Symposium on document analysis and information retrieval,* 317--332, 1995.

[Wolinski *et al.*, 1998] *F*. Wolinski, *F*. Vichot, O. Grémont. Producing NLP-based On-line Contentware. *Natural Language Processing & Industrial Applications*, Moncton, NB, Canada, 1998.

[Yang et Pedersen, 1997] Y. Yang, J. O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. *Proceedings of the 14th International Conference on Machine Learning (ICML'97),* 1997