Classification automatique de documents : théorie et pratique.

Jean Beney
jean.beney@INSA-lyon.fr

Département informatique
INSA de Lyon
22 février 2006

Ceci est un brouillon. N'hésitez pas à me faire vos remarques.

Chapitre 1

Introduction

Les dernières années ont vu une grande amélioration des capacités des ordinateurs et de leurs périphériques : vitesse de calcul, taille des mémoires centrales, rapidité de transfert sur les réseaux et surtout, pour ce qui nous intéresse, taille et vitesse d'accès des moyens de stockage. Tous ces gains de temps et de place permettent maintenant d'envisager de gérer électroniquement toutes sortes de documents textuels même pour des applications où cela ne serait pas vital.

Gérer des documents, c'est d'abord les organiser pour que leur récupération soit facile. Donc, soit définir des classes a priori (en fonction de nos connaissances du domaine, de l'organisation de l'entreprise) et y ranger les documents, soit, par application d'un algorithme adéquat, étudier la répartition des documents pour définir les classes les plus propices à leur répartition efficace (équilibrer les classes, par exemple).

Les classes peuvent être définies en fonction des rôles des acteurs qui vont recevoir et traiter ces documents. Il s'agit alors, non de ranger les documents pour les retrouver plus tard, mais de les envoyer aux personnes concernées : c'est ce qu'on appelle le *routage*.

Lorsqu'il s'agit d'informer des utilisateurs, sans que ceux-ci n'aient d'action particulière à réaliser, il faut éviter de les inonder de documents qui ne les concernent pas : on parle alors de *filtrage* des documents.

La *recherche d'information* (sur le web ou dans une base de documents) est un cas particulier de filtrage où une classe de documents utiles est définie en fonction des besoins immédiats de l'utilisateur. Il n'est donc pas question dans ce cas de préparer le travail en rangeant les documents dans des classes correspondants à ces besoins qui varient avec chaque utilisateur potentiel et même avec chaque centre d'intérêt de chaque utilisateur. On remarquera de plus que dans ce cas là, l'ensemble des documents se divise en 2 classes : celle des documents désirés et celle des autres documents. Cette dernière contient beaucoup plus de documents que la première.

Néanmoins, certains site web de recherche d'information ont prévu des classes correspondant à des rubriques fréquemment utilisées (vacances, finances, sports...) et des pages sélectionnées y ont été placées. Ces classes sont généralement orga-

nisées en taxinomie (classification hiérarchique) pour guider l'explorateur vers des sous-classes contenant un nombre limité de documents.

Cela permet de répondre aux situations les plus classiques mais pose des problèmes pour une page au contenu original qui n'entre pas exactement dans les classes prédéfinies : il faut lui choisir arbitrairement une classe et l'utilisateur intéressé risque de ne pas faire le même choix de classe et de passer à côté de l'information.

Dans toutes ces applications (organisation a priori hiérarchique ou non, filtrage, recherche d'information), un programme peut être utile pour classifier de nouveaux documents dans les classes existantes : c'est ce qu'on appelle un classifieur.

1.1 terminologie

- Nous nous intéressons donc principalement au processus de rangement de documents dans une ou plusieurs *classes* prédéfinies, ces classes étant connues par des *exemples* de documents. C'est la *classification (ou catégorisation) supervisée* de documents.
- Ce processus se base uniquement sur le *contenu* des documents. C'est-à-dire que l'on va extraire des documents-exemples des *termes* et chercher quels sont les termes représentatifs de chaque classe. Les termes peuvent être de différente nature, ce qui est discuté en **??**. Pour fixer les idées, on peut dans un premier temps supposer que chaque mot qui apparaît dans un texte est un terme.
  Ceci a pour corollaire que nous n'utilisons pas d'*index* attachés aux documents. En fait, ces index correspondent à des catégories bibliographiques et peuvent être ajoutés automatiquement par une méthode de classification.
- On distinguera ce problème de la *classification non supervisée* où les classes ne sont connues que par une *description* (liste de mots-clés, requête lors d'une recherche d'information, ...).
  La description peut-être un titre tel que "circuits intégrés", ou des mots-clés choisis par un expert, mais ce titre et ces mots-clés risquent de ne pas être bien utiles pour une classification suivant le contenu car ils n'apparaissent pas (ou pas tels quels quels) dans les documents. Cependant, certaines méthodes de classification supervisée sont définies de manière à pouvoir intégrer les descriptions des classes.
- Un tout autre problème est celui du *groupage (clustering)* qui consiste à définir des classes cohérentes dans un ensemble de documents. Le groupage s'appelle taxinomie lorsqu'il vise à établir une classification systématique et hiérarchisée.
- Pour éviter toute ambiguïté, nous appellerons *exemple* la paire <document, classe> qui informe que ce document appartient à cette classe. Les documents-exemples ont donc été *pré-classifiés* manuellement, c'est-à-dire qu'un ou des experts du domaine ont examiné les documents pour déter-

miner les classes auquel ils appartiennent. C'est cette expertise que nous allons tenter de copier dans la construction du classifieur.

- Donc, dans le cas de la classification supervisée, on se base sur des exemples. Si ceux-ci sont suffisamment nombreux, on les traite tous d'un coup pour obtenir un classifieur définitif. C'est le *traitement par lots*(batch). Si les exemples sont moins nombreux, on considérera que le classifieur doit être amélioré dès que possible, c'est-à-dire lorsque d'autres exemples arrivent. C'est la classification adaptative (ou incrémentale) qui permet entre autre de tenir compte des évolutions du vocabulaire (nouveaux termes techniques, expressions à la mode).
- Nous dirons aussi qu'un document d est *pertinent* pour une classe c s'il a été pré-classifié dans cette classe (la paire <d,c> est un exemple). D'autre part, un document est dit *sélectionné* dans une classe si le classifieur automatique arrive au résultat que ce document appartient à cette classe. Le classifieur sera parfait si tous les documents pertinents sont sélectionnés et qu'aucun document non pertinent n'est sélectionné.
- En général, rien n'empêche qu'un document appartienne à plusieurs classes ; nous parlerons alors de *multiclassification*. Dans certains cas cependant, chaque document n'appartiendra qu'à une seule classe car ces classes ont été définies de façon à ne pas se recouvrir : c'est la *monoclassification*[1].
- Pour l'expérimentation, les exemples seront séparés en 2 : l'*ensemble d'apprentissage* et l'*ensemble de test*. La méthode sera appliquée sur le premier ensemble et son résultat mesuré sur le deuxième. L'idée est de voir comment la classification réussit sur des documents qui n'ont pas été regardés pendant l'apprentissage et donc d'avoir ainsi une idée du résultat que l'on obtiendra sur de nouveaux documents (voir la discussion en 1.4.2).
  Nous dirons aussi que les documents des deux ensembles sont respectivement *vus* et *cachés*.

1.1.1 exemples

1.1.1.1 Demandes de brevets

Les offices de brevets (en France, l'Institut National de la Propriété Industrielle ; pour 36 pays d'Europe, l'EPO : Office Européen des Brevets) sont chargés de délivrer des brevets, c'est-à-dire de certifier l'originalité d'une invention et ainsi de la protéger contre d'autres qui n'en seraient que la copie pure et simple.

Les demandes de brevets sont donc examinées et comparées aux brevets déjà existants. Ce travail ne peut être bien fait par des gens très compétents dans un domaine.

[1] Certains auteurs parlent de monoclassification lorsque l'on ne traite qu'une classe à, la fois, indépendamment des autres classes : un document peut alors appartenir à chacune des classes successivement. Symétriquement, ils parlent de multiclassification lorsque toutes les classes sont traitées ensemble et que chaque document n'est mis que dans une seule de ces classes.

L'office a organisé ses experts en 549 équipes organisées en 44 directoires. Chaque équipe a donc la charge d'un domaine scientifique assez restreint et il faut envoyer chaque demande de brevet a l'équipe ou aux équipes concernées. Ce travail de répartition, bien que pouvant être assimilé à un travail de secrétariat, nécessite également du personnel ayant un très haut niveau de culture scientifique et technique (ainsi 60% du personnel de l'EPO a un diplôme universitaire). Mais l'éventail des connaissances nécessaires pour couvrir tous les domaines est telle que les répartiteurs se trompent tout de même dans 18% des cas.

En outre, la très forte activité (160 000 demandes par an) explique la nécessité d'automatiser l'envoi aux experts concernés. Mais nous disposons d'une bonne base pour ce travail : une énorme quantité de brevets a déjà été traitée par les experts. S'ils avaient été mal aiguillés par les répartiteur, les experts ont rectifié la classification en les faisant passer à d'autres experts. On peut ainsi faire un traitement par lots.

Les documents sont multiclassifiés mais il n'est pas nécessaire de retrouver toutes les classes car un expert qui reçoit une demande à examiner est capable de l'envoyer à des collègues concernés spécialistes d'autres domaines. L'expérience menée avec 2000 documents par classe a donné une précision du premier choix (Cf. 1.3.5) de 80%.

1.1.1.2 Distribution d'information

Certaines entreprises font commerce d'informations concernant un ou plusieurs domaines scientifiques, culturels ou sociétaux.

Fiscaal-up-to-date est une société néerlandaise qui collecte et distribue sur CD-ROM toutes les informations sur la fiscalité aux Pays-Bas : lois, décret et règlements mais aussi compte-rendu de débats à la chambre et d'autres informations plus difficiles à trouver dans la presse. Pour aider les clients, les documents sont classifiés en grand sujets (impôts directs, TVA, fiscalité des entreprises, ... 25 sujets) subdivisés en catégories plus précises (plus de 2700 classes). Certains documents appartiennent à plusieurs catégories.

Semblablement, des *portails internet* (Yahoo, par exemple) proposent des liens organisés en catégories (immobilier, cinéma, ...), sous-catégories (immobilier-ventes, immobilier-locations, ...), ...

Les documents collectés par Fiscaal ou les sites repérés par Yahoo doivent être classifiés avant d'être proposés aux clients.

Pour Fiscaal, il s'est avéré que l'automatisation était très difficile (moins de 60% de réussite) à cause du faible nombre de documents dans certaines catégories et à cause d'incohérences dans la classification existante dues à des changements dans l'organisation des catégories. Le travail d'automatisation a donc nécessité une redéfinition propre des catégories et un nettoyage de la classification existante. Nous préciserons plus loin comment cela peut être réalisé (Cf. **??**).

À part cela, il a été réalisé un traitement par lots multiclassifié sur 32000 documents déjà diffusés par Fiscaal.

1.1.1.3 Recherche d'information

Lorsqu'un utilisateur cherche un renseignement sur le web avec les moteurs actuels, il indique des mots dont il pense qu'ils apparaîtront dans les pages qui l'intéresse. Il n'y a donc pas de phase d'apprentissage automatique. Et l'on peut considérer que l'on a affaire à une classification binaire : les documents recherchés et les autres.

Mais on peut imaginer des moteurs plus performants qui tiendraient compte des pages acceptées et refusées par l'utilisateur pour lui en proposer de plus pertinentes. Il s'agirait alors de traitement adaptatif.

De tels systèmes ne sont pas diffusés car ils sont peu utiles : dans la grande majorité des cas, lorsque le renseignement désiré est trouvé, l'utilisateur passe à autre chose et n'utilisera plus jamais le classifieur construit. Et, si d'autres utilisateurs font la même recherche, ils n'ont pas de moyen de réutiliser ce travail.

Néanmoins, ils pourrait être utiles à des professionnels désirant se tenir au courant des nouveautés dans un domaine particulier.

1.1.2 résultat de la classification

Dans la plupart des cas, le résultat d'un classifieur est simplement booléen : tel document d appartient ou n'appartient pas à telle classe c.

Ce résultat peut s'accompagner d'une valeur qui représente la confiance qu'on lui accorde : cela peut être un score, une probabilité que le document soit dans la classe.

Dans certains cas, on présentera comme conclusion une liste ordonnée (ranking) : soit la liste des documents les plus probables dans chaque classe, soit la liste des classe les plus probables pour chaque document. Cette liste peut être de taille limitée (ex. 10 documents par classe) ou illimitée (tous les documents). La qualité de la liste résultat sera évaluée par des moyens appropriés (Cf. 1.3).

1.1.3 Notation

Nous travaillons donc avec un ensemble de classes :

$$C = \{c_1, ..., c_M\}, M = |C|$$

et un ensemble de documents :

$$D = \{d_1, ..., d_N\}, N = |D|$$

chaque document d_i contient n_i termes. Il existe N termes différents.

Nous disposons d'exemples, c'est-à-dire d'information sur l'appartenance de certains documents à certaines classes. Cela revient à disposer d'un ensemble de paires <document, classe> :

$$E = \{< d_i, c_j > | d_i \in c_j\}$$

Il est alors implicite que les paires non incluses dans E sont des contre-exemples :

$$< d_i, c_j > \notin E \Rightarrow dj \notin c_i$$

Ceci doit être gardé en mémoire lors de la constitution d'un jeu d'exemples.

Pour certaines démonstrations, on préférera expliciter les contre-exemples en définissant :

$$E' = \{< d_i, y_{ji} > | 1 \leq i \leq m, 1 \leq j \leq mn\}$$

avec[2] :

$$d_j \in c_i \Rightarrow y_{ij} = 1$$

$$d_j \notin c_i \Rightarrow y_{ji} = 0$$

Nous parlerons également d'exemple positif (pour la classe c) lorsque $y_{ji} = 1$ et d'exemple négatif dans le cas contraire. Le mot *exemple*, sans précision, se référera alors aux exemples positifs.

L'ensemble E_i des exemples positifs d'une classe ci est donc l'ensemble des documents pertinents pour cette classe :

$$E_i = \{d_j \in D | < d_j, c_i > \in E\} = \{d_j \in D \in D | y_{ji} = 1\}$$

Le résultat de la classification automatique sera noté ainsi :

- un booléen z_{ji} indique si d_j a été mis dans la classe c_i ou non ;
- ou un rang $Rang_{ji}$ indique comment le document d_j a été classé dans la classe c_i.

Un document est un ensemble de termes (voir **??**ci-dessous ce qu'est un terme) qui peuvent apparaître plusieurs fois. On représentera donc un document par un ensemble de paires

$$d_i = \{(t_k, n_{ik})\}$$

où n_{il} est la fréquence (nombre d'occurence) du terme t_l dans le document d_i. L'ensemble total des termes T est donc :

$$T = \{t_k | \exists d_i : (t_k, n_{ik}) \in d_i\}, \qquad L = |T|$$

Le nombre de documents où un terme apparaît est alors :

$$f_k = |\{d_i | (t_k, n_{ik}) \in d_i\}|$$

[2] certains auteurs utilisent -1 pour *faux* à la place de 0

1.2 Le modèle vectoriel

Les théories associées aux méthodes de classification se placent toutes dans le même modèle : un *espace vectoriel* de dimension L dont chaque axe correspond à un terme t_k, la valeur sur l'axe représentant la *force* (importance, voir les différentes manière de la calculer en **??**) du terme dans un document ou dans une classe. Chaque document d_i est alors représenté par le vecteur des forces des termes qu'il contient complété par des 0 pour les termes qu'il ne contient pas :

$$\overrightarrow{d_i} = [s_{1i}, s_{2i,}..., s_{Li}]$$

où N est le nombre total des termes qui apparaissent dans l'ensemble des documents, et $s_{kj} = 0$ si le terme t_k n'apparaît pas dans le document d_j.

En général, on suppose également que les différents termes utilisés sont indépendants (repère orthogonal), mais certaines méthodes annexes visent à repérer les dépendances avant la classification proprement dite, de façon à diminuer la dimension de l'espace vectoriel.

Les *termes* qui représentent un document peuvent être de différentes sortes mais partent tous des mots de son texte. Le plus simple est de considérer ces mots tels qu'ils apparaissent ; on parle alors de représentation par *paquets de mots*.

Ces mots peuvent être remplacés par leurs *racines*, ce qui diminuent la dimension de l'espace puisque les différentes formes d'un mot (singulier/pluriel, différentes conjugaison d'un verbe, ...) se retrouvent sur le même axe.

Mais les idées importantes n'étant pas forcément représentées dans une phrase par des mots simples, on peut chercher les *groupes de mots* significatifs : par exemple, les groupes nominaux, ce qui inclut les noms composés.

L'idéal serait de ne plus parler de mots mais d'idées. Pour cela, des tentatives ont été faites pour trouver les *notions* associées aux mots. On utilise pour cela un graphe qui organise les notions en une ontologie et donne les différents synonymes pour chaque notion.

Ces différentes méthodes linguistique seront développées en **??**. D'autre part, les différentes manière de calculer la force des termes seront détaillées en **??**.

1.2.1 Classifieurs linéaires

Étant donné donc un espace vectoriel correspondant à tous les termes qui apparaissent dans les documents à classer.

Un classifieur est linéaire si la décision de mettre ou non un document d dans une classe c se base sur une fonction linéaire des forces des termes dans le document. Étant donné $\overrightarrow{d_j} = [s_{1j}, s_{2j,}..., s_{Nj}]$, on définit le score d'un document d_j dans une classe c_i par :

$$S_{ji} = \sum_{k=1,N} s_{kj} w_{ki}$$

où w_{ki} est un poids estimé pour le terme t_k dans la classe c_i. On décidera alors que :

$$d \in c \equiv S_{ji} \geq \theta_i$$

où θ_i est un seuil fixe ou calculé pour la classe c_i.

1.2.1.1 changement d'espace +=+

1.2.2 Stratégies

Les méthodes utilisées peuvent être classées suivant la stratégie générale qu'elles utilisent. Nous distinguerons la *modélisation* , la *séparation* et le *voisinage*[3].

modélisation : cette stratégie consiste à chercher un modèle qui représente la classe, c'est-à-dire un document fictif d_M résumant les caractéristiques de la classe.

Cela pourra être fait par des méthodes statistiques (SBS) ou géométriques (Rocchio). Puis les documents d à classer seront comparés à ce document modèle d_M et d sera affecté à la classe s'il est suffisamment près de d_M.

L'appréciation de la similitude entre d_i et d_M est faite par le produit scalaire :

$$S_{d,c} = \overrightarrow{d} \cdot \overrightarrow{d_M} = \sum_{i=1,N} s_{i,d} s_{i,d_M}$$

et l'on voit que l'on obtient un classifieur linéaire où le poids d'un terme dans une classe est la force de ce terme dans le document modèle : $w_{i,c} = s_{i,d_M}$.

Comme :

$$\overrightarrow{d} \cdot \overrightarrow{d_M} = \cos(\overrightarrow{d}, \overrightarrow{d_M}) \parallel \overrightarrow{d} \parallel \cdot \parallel \overrightarrow{d_M} \parallel$$

$S_{d,c}$ est nul si les 2 vecteurs sont orthogonaux. Si toutes les forces $s_{i,d}$ sont positives ou nulles, cela ne peut se produire que si les documents d et d_M n'ont pas de termes en commun :

$$s_{i,d_M} \neq 0 \Rightarrow s_{i,d} = 0$$

$$\wedge s_{i,d} \neq 0 \Rightarrow s_{i,d_M} = 0$$

D'autre part, $\overrightarrow{d} \cdot \overrightarrow{d_M}$ est maximum lorsque les vecteurs sont colinéaires et si l'on a normalisé tous les vecteurs, $\overrightarrow{d} \cdot \overrightarrow{d_M} = \cos(\overrightarrow{d}, \overrightarrow{d_M})$ est égal à 1 dans ce cas. On obtient donc une mesure de l'appartenance d'un document à une classe dont les valeurs appartiennent à l'intervalle $[0, 1]$ mais qui n'est pas forcément l'estimation d'une probabilité.

Si l'on se contente d'un seuil fixe pour déterminer si un document est suffisamment près du modèle, on ne tient pas compte de la dispersion des documents qui peut être très différente d'une classe à une autre. Le seuil doit donc être

[3] D'autres techniques ne s'insèrent pas dans ce découpage, par exemple, le *boosting* qui consiste à combiner des classifieurs de mauvaise qualité pour obtenir un bon classifieur.

adapté à chaque classe, ou bien les poids normalisés différemment dans chaque classe pour qu'un seuil fixe puisse être utilisé. Ceci peut être réalisé en tenant compte uniquement de la répartition des exemples positifs dans l'espace, ou en prenant en compte les exemples négatifs (Rocchio).

séparation : cette stratégie consiste à chercher un hyperplan (surface dont la fonction est linéaire) qui sépare les exemples des contre-exemples.

FORMULES +=+

Les méthodes SVM et Winnow diffèrent par la manière d'y parvenir : la première cherche les documents qui sont proche de cette frontière puis la calcule tandis que la deuxième procède par approche progressive, en modifiant les coefficients du plan chaque fois que l'on trouve un exemple (positif ou négatif) encore mal classé.

voisinage : une méthode (kNN) utilise une stratégie qui lui est propre et qui consiste à chercher les documents pré-classifiés qui sont proches d'un document à classer et de choisir (à la majorité) la classe en fonction des classes de ces voisins.

On remarquera que cela ne nécessite aucune préparation, contrairement aux autres stratégies, mais que chaque nouveau document doit être comparé à tous les exemples connus.

1.2.2.1 Efficacité :

De manière générale, nous pouvons faire les constations suivantes : les méthodes par modélisation nécessitent moins de calculs pour l'apprentissage que les méthodes par séparation. En effet, SVM nécessite la résolution d'un problème d'optimisation quadratique et Winnow, travaille par apprentissage progressif, ce qui peut nécessiter de nombreuses itérations où chaque document est examiné. Néanmoins, nous verrons qu'en pratique, il faut se limiter à un petit nombre d'itérations et que le nombre des exemples peut être réduit après une première passe.

Avec ces deux stratégies, la classification d'un nouveau document se réduit au calcul d'un produit scalaire qui peut encore être accéléré si les 0 du vecteur des forces ne sont pas enregistrés. Par contre, kNN procède alors à n produits scalaires, si n est le nombre de documents pré-classifiés.

Sur le plan de la qualité des résultats, la séparation est généralement considérée comme la stratégie la meilleure. Mais, si la modélisation n'obtient pas un aussi bon résultat en général, elle peut se révéler intéressante lorsque les exemples sont peu nombreux.

1.2.3 Le modèle probabiliste

En supposant que les termes apparaissent indépendamment, il est possible d'estimer la probabilité qu'un document appartienne à une classe. Cela revient à travailler dans un espace vectoriel où les poids des termes sont remplacés par des probabilités. Ces probabilités seront estimées sur l'ensemble d'apprentissage :

– probabilité d'appartennir à une classe c_i :

$$P(c_i) = \frac{|c_i|}{\sum_{i=1,n} |c_i|}$$

Un document est alors affecté à une classe si la probabilité estimée qu'il lui appartienne est supérieure à la probabilité qu'il ne lui appartienne pas. On peut aussi se fixer un seuil et n'affecter le document à une classe si la probabilité est supérieure à ce seuil.

C'est le principe de la méthode SBC (Simple Bayesian Classifier) basée sur le théorème de Bayes.

1.2.4 Le modèle booléen

Sans calculer de poids ou estimer de probabilités, on peut baser la classification sur la présence ou l'absence de certains termes dans les documents.

Dans ce cas, la première méthode pour combiner les différents termes est de noter la présence et l'absence par 1 et 0, de considérer ces valeurs comme des poids et d'appliquer une méthode dans l'espace vectoriel correspondant.

La deuxième méthode consiste à travailler réellement en booléen (vrai, faux) et à combiner ces valeurs par les opérateurs *et*, *ou*, *non*. Il n'existe pas d'outil qui détermine les opérateurs à utiliser par apprentissage sur un ensemble de document connus. Cette façon de faire est utilisée pour la recherche d'information où l'expression booléenne est donnée par l'utilisateur : c'est ce qu'on appelle une requête.

1.3 Mesures de la qualité d'un classifieur

Lorsque différentes méthodes sont disponibles pour construire un classifieur, il faut évaluer leurs performances respectives pour choisir la meilleure. Évidemment, la notion de *meilleure méthode* est relative à l'utilisation que l'on veut faire du résultat de la classification. C'est pourquoi, différentes valeurs peuvent être utilisées, qui se basent toutes sur une certaine vision du résultat par un utilisateur. Par la suite,

Dans un premier temps, nous supposerons que nous ne travaillons qu'avec une seule classe et son complément, c'est-à-dire qu'il s'agit de séparer les documents qui appartiennent à une classe de ceux qui n'y appartiennent pas.

1.3.1 Précision, rappel

Une première manière de juger de la qualité d'un classifieur est de compter les documents qui manquent dans chaque classe (évaluation de l'incomplétude) et ceux qui y sont en trop (évaluation du bruit).

L'ensemble des documents peut être divisé en 4 sous-ensembles dans lesquels les documents entrent selon qu'ils sont on non pertinents ou sélectionnés.

	pertinents	non pertinents	total
sélectionnés	X	Y	S
non sélectionnés	Z	W	$n - S$
total	P_e	$n - P_e$	

La *précision* est la proportion de documents correctement sélectionnés dans la classe :

$$P = \frac{X}{X+Y} = \frac{X}{S}$$

Le *rappel* (recall) est la proportion de documents pertinents qui ont été sélectionnés :

$$R = \frac{X}{X+Z} = \frac{X}{P_e}$$

Ces valeurs mesurent le degré de justesse (précision) et le degré de complétude (rappel) au sens de la théorie des modèles.

On notera que le rappel ne peut-être calculé que si l'on connaît le nombre de documents pertinents, donc si la classification a déjà été faite. On l'utilisera donc pour évaluer le résultat sur un jeu de documents connus. Mais on ne pourra pas l'utiliser pour évaluer en cours de route un classifieur adaptatif (dans cas de la recherche d'information, par exemple).

Ces valeurs permettent d'évaluer la gène créée par les défauts de classification : information manquante (rappel < 1) et bruit gênant (précision < 1).

courbes P/R

On peut évidemment augmenter autant que l'on veut la valeur du rappel en sélectionnant de plus en plus de documents. Mais dans ce cas, on sélectionne de plus en plus de documents non pertinents et la précision diminue. À la limite, si l'on sélectionne tous les documents, on est sûr de sélectionner tous les documents pertinents, ce qui donne un rappel égal à 1 et une précision proche de 0 (mais non nulle).

Symétriquement, si on sélectionne de moins en moins de documents, on peut améliorer la précision. On diminue ainsi le nombre de documents pertinents sélectionnés (le rappel), mais on n'est jamais sûr d'obtenir une précision parfaite car il est possible, lorsqu'il ne reste qu'un document sélectionné, que celui-ci ne soit pas pertinent.

À part ce dernier cas particulier, la précision et le rappel évoluent en sens inverse lorsque l'on fait varier le nombre de documents sélectionnés, comme on le voit sur les 2 exemples de courbes, figure 1.1. La première correspond à un très bon classifieur, la seconde à un classifieur médiocre.

Il faut donc trouver un compromis entre un plus ou moins bon rappel et une plus ou moins bonne précision.

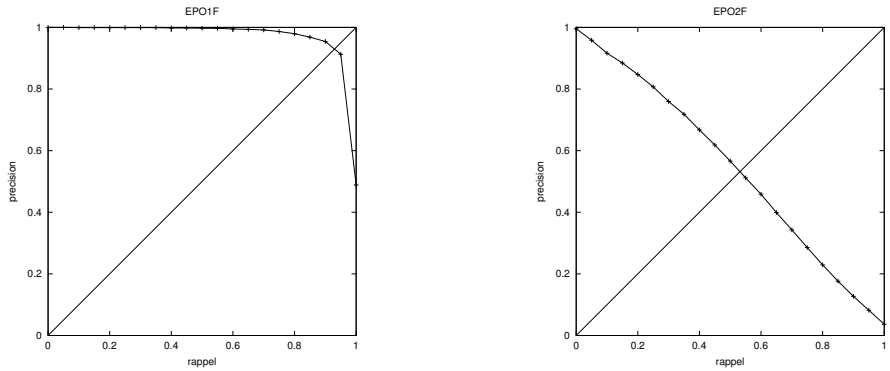

FIG. 1.1 – 2 exemples de courbes P/R

Le point d'égalité[4]

Une première solution de compromis est le point où la précision et le rappel sont égaux :

$$P = \frac{X}{X+Y} = R = \frac{X}{X+Z}$$

Il en découle : $Y = Z$ et : $S = P_e$.

Ce point est donc obtenu lorsque le nombre de documents sélectionnés est égal au nombre de documents pertinents. Sur la figure, c'est l'endroit où la courbe croise la diagonale.

1.3.2 La fonction F

Pour combiner les valeurs de la précision et du rappel en une seule valeur on utilise généralement la moyenne harmonique pondérée suivant la formule suivante :

$$F(\gamma) = \frac{(\gamma^2+1)PR}{\gamma^2 P + R}, \gamma \geq 0 \tag{1.1}$$

On remarque que :

$$\forall \gamma : R = 1 \wedge P = 1 \Leftrightarrow F(\gamma) = 1$$

$$\forall \gamma : R = 0 \vee P = 0 \Leftrightarrow F(\gamma) = 0$$

C'est pourquoi on utilise cette fonction plutôt qu'une autre plus simple, telle que la moyenne arithmétique.

Le paramètre γ sert à indiquer l'importance que l'on accorde respectivement à la précision et au rappel : plus γ est grand, plus le rappel prend de l'importance.

[4]Le terme anglais *breakeven point* désigne à l'origine, dans le domaine du commerce, le point où les dépenses et les rentrées sont égales. Dans ce cas, il se traduit par *seuil de rentabilité*.

$$F(0) = P,\ F(\infty) = R,\ F(1) = \frac{2PR}{P+R},\ 0 \leq F(\gamma) \leq 1$$

C'est pourquoi on prend γ égal à 1 lorsqu'on n'a pas de raisons de privilégier l'une ou l'autre valeur.

La valeur de γ étant choisi, on considérera la valeur $1 - F(\gamma)$ comme une estimation de l'erreur et on cherchera, pour chaque méthode, les paramètres qui donnent la plus grande valeur de $F(\gamma)$.

Au point d'égalité, on a toujours $F(\gamma) = P = R$.

La fonction F et le point d'égalité

Si l'on choisit $\gamma = 1$, on utilise donc la moyenne harmonique sans pondération :

$$F(1) = \frac{2PR}{P+R} = \frac{2\frac{X}{S} \times \frac{X}{R_e}}{\frac{X}{S} + \frac{X}{R_e}} = \frac{2\frac{X.X}{S \times R_e}}{\frac{X \times R_e + X \times S}{S \times R_e}} = \frac{2X}{R_e + S} \tag{1.2}$$

+=+ REVOIR LES NOMS DES VARIABLES X=RS ...

Au point d'égalité :

$$F(\gamma) = \frac{(\gamma^2+1)PP}{\gamma^2 P + P} = P = R$$

1.3.3 Comparaison de jeux de documents, retombées, G_γ

Si ces valeurs permettent de comparer deux classifieurs sur le même jeux de documents, il est plus délicat de comparer des classification de différents jeux de documents. En effet, la précision est influencée par le nombre de contre-exemples dans le jeux d'essai.

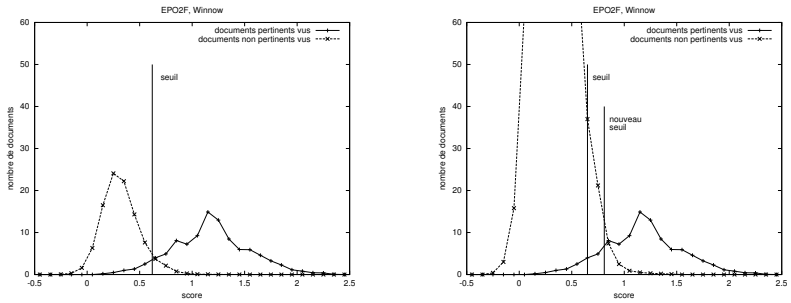

FIG. 1.2 – influence du nombre de contre-exemples

La figure 1.2 montre la répartition des scores des documents pertinents et non pertinents dans 2 cas de figure :

– les 2 ensembles contiennent le même nombre de documents ;
– il y a 10 fois plus de documents non pertinents que de pertinents.

La surface sous chaque courbe montre le nombre de documents. Un seuil étant choisi, la surface sous la courbe des pertinents à gauche du seuil correspond aux documents pertinents non sélectionnés, donc représente l'erreur de rappel. La surface à droite du seuil, sous la courbe des non pertinents représente l'erreur de précision (documents sélectionnés à tort). $F(1)$ est maximum lorsque ces deux erreurs sont égales

Le premier graphe suggère de choisir un seuil de 0,7 et donc de sélectionner tous les documents dont le score est supérieur à cette valeur. Les surfaces sous les courbes représentant le nombre de documents, on voit qu'avec ce seuil, $\overline{P}S \approx P\overline{S}$ et donc que $P \approx S \approx F(1)$.

Si l'on choisit le même seuil dans le deuxième cas, la surface sous la courbe des non pertinents, à droite de 0,7, est beaucoup plus grande et donc $\overline{P}S \gg P\overline{S}$ et la précision est bien moins bonne qu'elle n'était dans le premier cas tandis que le rappel n'a pas changé. Pourtant le calcul des scores est le même ainsi que le seuil choisi, donc le classifieur est le même.

Il faut donc bien garder à l'esprit que la précision dépend du nombre de contre-exemples et éviter d'utiliser cette précision pour comparer 2 classifieurs travaillant sur 2 jeux de données différents.

Si l'on doit quand même travailler avec la précision, l'ensemble de documents utilisé pour l'apprentissage (ou simplement pour l'estimation du seuil) devra respecter la proportion de documents que l'on trouve dans l'ensemble total des documents.

Par contre si l'on définit les *retombées* (fall-out) par la proportion de documents non-pertinents qui ont été sélectionné par erreur :

$$T = \frac{Y}{Y+W} = \frac{Y}{n-P}$$

ces retombées restent inchangées lorsque seule la proportion de pertinents change. Cette valeur est donc caractéristique du classifieur.

On pourra alors, en imitant la fonction F, définir une fonction G qui combine les retombées et le rappel, en remplaçant la précision P par $1-T$, la proportion de documents non pertinents non sélectionnés :

$$G(\gamma) = \frac{(\gamma^2+1)(1-T)R}{\gamma^2(1-T)+R}, \gamma \geq 0$$

Comme pour la fonction F , nous avons les propriétés :

$$G(0) = 1-T,\ G(\infty) = R,\ G(1) = \frac{2(1-T)R}{1-T+R},\ 0 \leq G(\gamma) \leq 1$$

1.3.4 Compléments sur la fonction F(1)

La valeur de $F(\gamma)$ varie avec P et R donc le nombre de documents sélectionnés (pertinents ou non) donc avec le seuil θ d'acceptation d'un classifieur linéaire. La valeur optimum de θ est celle qui donne la valeur maximum de $F(\gamma)$.

Dans le cas où $\gamma = 1$, ce point semble être proche du point d'égalité, comme on le voit sur la figure 1.3.

Est-il possible que $F' = 0$ quand $P = R$? D'après la définition 1.1 :

$$F' = 2\frac{(P'R + PR')(P + R) - (P' + R')PR}{(P + R)^2}$$

$$F' = \frac{P'PR + P'R^2 + P^2R' + PRR' - P'PR - R'PR}{(P + R)^2}$$

$$F' = 2\frac{P'R^2 + 2PR'}{(P + R)^2}$$

Donc :

$$F' = 0 \Leftrightarrow P'R^2 + P^2R' = 0$$

$$F' = 0 \wedge P = R \Rightarrow P^2(P' + R') = 0$$

$$F' = 0 \wedge P = R \Rightarrow P' = -R'$$

Donc $P = R \wedge P' = -R' \Rightarrow F' = 0$, ce qui semble être vérifié sur les deux graphes de la figure **??**.

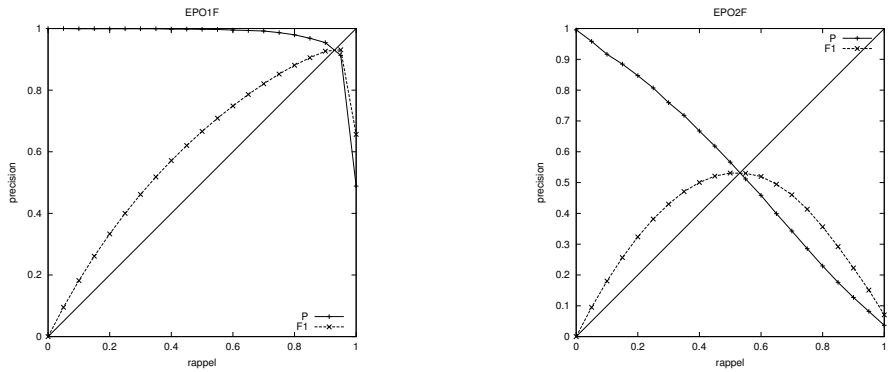

FIG. 1.3 – 2 les courbes P/R avec F(1)

En partant de l'équation 1.2 :

$$F' = 2\frac{X'(R_e + S) - XS'}{(Re + S)^2}$$

comme $S = X + Y$:

$$F' = 2\frac{R_eX' + XX' + YX' - XX' - XY'}{(Re + S)^2}$$

$$F' = 2\frac{(R_e + Y)X' - XY'}{(Re + S)^2}$$

Le dénominateur étant toujours strictement positif :

$$F' = 0 \Leftrightarrow (R_e + Y)X' - XY' = 0$$

$$F' = 0 \Leftrightarrow (R_e + Y)X' = XY'$$

$$F' = 0 \Leftrightarrow \frac{Y'}{X'} = \frac{(R_e + Y)}{X} = \frac{X + Z + Y}{X} = \frac{Z + Y}{X} + 1$$

X, Y, Z étant positifs :

$$F' = 0 \Leftrightarrow Y' > X'$$

Le point de croisement :

Les courbes de distribution des documents pertinents ou non en fonction du score (figure 1.2) représentent également les variations X' et Y' des nombres de documents sélectionnés (pertinents ou non) en fonction du seuil d'acceptation θ. Elles peuvent être assimilées à des lois normales de paramètres respectifs m_{P_e}, σ_{P_e}, (pertinents) et $m_{\overline{P_e}}, \sigma_{\overline{P_e}}$ (non pertinents). Dans l'exemple, ces paramêtres ont les valeurs estimées suivantes :

$m_{P_e} = 1.1764 \quad \sigma_{P_e} = 0.3624 \quad m_{\overline{P_e}} = 0.3218 \quad \sigma_{\overline{P_e}} = 0.1780$

Le point de croisement des deux courbes est tel que :

$$n(\frac{b - m_{P_e}}{\sigma_{P_e}}) = rn(\frac{b - m_{\overline{P_e}}}{\sigma_{\overline{P_e}}})$$

où n n est la loi de probabilité normale centrée et réduite et r est le ratio entre les nombres $\overline{P}_e$ and P_e de documents non pertinents et pertinents :$r = \frac{\overline{P}_e}{P_e}$. Par symétrie :

$$n(\frac{b - m_{\overline{P_e}}}{\sigma_{\overline{P_e}}}) = rn(\frac{m_{\overline{P_e}} - b}{\sigma_{\overline{P_e}}}) \tag{1.3}$$

Quand $r = 1$, nous avons 2 possibilités

$$\frac{b - m_{P_e}}{\sigma_{P_e}} = \frac{b - m_{\overline{P_e}}}{\sigma_{\overline{P_e}}} \quad ou \quad \frac{b - m_{P_e}}{\sigma_{P_e}} = \frac{m_{\overline{P_e}} - b}{\sigma_{\overline{P_e}}}$$

Comme on le voit sur la figure, avec de tels moyennes et écarts-types, seule la seconde possibiliyé donne une solution :

$$\sigma_{P_e} m_{\overline{P_e}} - \sigma_R b = \sigma_{\overline{P_e}} b - \sigma_{\overline{P_e}} m_{P_e}$$

$$\sigma_{P_e} m_{\overline{P_e}} + \sigma_{\overline{P_e}} m_{P_e} = \sigma_{\overline{P_e}} b + \sigma_{P_e} b$$

$$b = \frac{\sigma_{P_e} m_{\overline{P_e}} + \sigma_{\overline{P_e}} m_{P_e}}{\sigma_{\overline{P_e}} + \sigma_{P_e}}$$

Numériquement nous obtenons $b = 0.603$

Quand $r \neq 1$, la solution de l'équation 1.3 doit être trouvée en utilisation les tables de distribution de la loi normale.

F(1) et le point de croisement :

Puisque$F' = 0 \Leftrightarrow Y' > X'$, le point ou F(1) est maximum est à gauche du point de croisement.

Le point d'égalité et le point de croisement :

Au point d'égalité ($P = R$), nous avons vu que$Y = Z$. C'est-à-dire que les surfaces des "*triangles*" (sous les courbes de nombres de documents et limités par la verticale en θ) sont égales.

$$P_R(x < b) = rP_{\overline{R}}(x > b)$$

$$P_R(x < b) = rP_{\overline{R}}(x < b')$$

avec b' symétrique de b par rapport à la moyenne :

$$m_{\overline{R}} - b' = b - m_R \Rightarrow b' = 2m_R - b$$

Comme pour le point de croisement :

$$N(\frac{b - m_R}{\sigma_R}) = rN(\frac{b' - m_{\overline{R}}}{\sigma_{\overline{R}}}) = rN(\frac{m_{\overline{R}} - b}{\sigma_{\overline{R}}})$$

Où N est la probabilité cumulée de la loi normale centrée et réduite.
Quand $r = 1$, on retrouve :

$$\frac{b - m_{P_e}}{\sigma_{P_e}} = \frac{m_{\overline{P_e}} - b}{\sigma_{\overline{P_e}}}$$

et le point d'égalité est le point de croisement.
Quand $r \neq 1$, il faut chercher la solution dans les tables de la probabilité cumulée de la loi normale.

F(1) et le point d'égalité :

Le maximum de F(1) se trouve strictement à gauche (θinférieur) du point de croisement. Quand le nombre de documents pertinents est égal au nombre de documents non pertinents, le point d'égalité est sur le point de croisement.

Dans ce cas au moins, le maximum de F(1) est donc *différent* du point d'égalité.

1.3.5 Précision du premier choix

Voyons maintenant le cas du routage : les documents arrivant doivent être aiguillés vers 1 ou plusieurs destinataires. Si l'on cherche absolument à atteindre tous les destinataires concernés, l'efficacité de la méthode sera mesurée par les valeurs vues précédemment : précision et rappel.

Mais, dans certains cas (voir l'exemple des demandes de brevets **??**), on peut supposer qu'un destinataire averti sera capable de transmettre le document aux autres personnes concernées ; on envoie donc le document à un seul destinataire. Le rappel est alors forcément mauvais puisque délibérément on n'a pas cherché à trouver toutes les classes (les destinataires) et seul la précision sera une mesure utile de la qualité : c'est la *précision du premier choix*

Remarquons que dans cette situation, les exemples sont multiclassifiés, tandis que le résultat est volontairement monoclassifié.

+=+ précision moyennée en 1 point

1.3.6 Fonctions d'utilité

Ces fonctions veulent refléter l'utilité pour un usager de l'ensemble des documents sélectionnés par le classifieur. Reprenant le tableau précédent, on affecte une valeur à chaque cas possible.

	pertinents	non pertinents
sélectionnés	U_X	U_Y
non sélectionnés	U_Z	U_W

Et la fonction d'utilité est le polynôme :

$$U = X \times U_X + Y \times U_Y + Z \times U_Z + W \times U_W$$

Cette valeur sera bien sur positive ou nulle pour les documents bien classés et négative (gène) ou nulle pour les documents mal classés. En effet on peut considérer que certaines erreurs n'apportent évidemment rien mais qu'elles ne gênent pas non plus. Le choix des coefficients de U est laissée à l'appréciation de chacun en fonction de sa perception du problème.

Mais en général U_W est pris égal à 0 car les documents non-pertinents rejetés ne gênent en rien mais n'apportent aucune valeur ajoutée.

Si, en plus, U_Z est également nul, les documents non sélectionnés n'interviennent pas dans le calcul. Cela a l'avantage de permettre l'évaluation de U en regardant uniquement les documents sélectionnés, donc sans connaître une pré-classification de tous les documents. Par exemple, la méthode peut être évaluée en regardant les documents sélectionnés sur le web, situation où il serait impossible de pré-classifier tous les documents. Ce cas est donc bien adapté à la comparaison de méthode de recherche d'information.

exemples : on trouve dans la littérature diverses applications avec des coefficients différents.

nom	U_X	U_Z	U_Y	U_T
erreur	0	-1	-1	0
	3	0	-2	0
	3	-1	-1	0

+=+

1.3.7 Exactitude, erreur

Ces mesures donnent le même rôle à la classe considérée et son complément. Ce sont l'exactitude E, proportions de documents correctement mis dans la classe ou correctement rejetés :

$$E = \frac{X+W}{n} = \frac{X+W}{X+Y+Z+W}$$

et l'erreur $\overline{E}$, proportion des document incorrectement mis dans la classe ou incorrectement rejetés :

$$\overline{E} = \frac{Y+Z}{n} = 1 - E$$

Ces mesures sont rarement utilisées parce que n est généralement très grand +=+ voir Yang 99 et Cohen 95 depuis seb02

1.3.8 précision interpolée (pour les listes ordonnées)

Les courbes P/R présentées **??** ont été établies à partir d'une liste ordonnée des documents accompagnés d'un score pour une classe donnée. En faisant varier le seuil θ, on a obtenu les points de la courbe. Comme on peut le voir sur le graphe, les valeurs de θ ont été choisies de manière à donner des valeurs du rappel égales à 0, 0,05, ..., 1.

Le cas extrême $R = 0$ peut on non être inclus dans le calcul. En effet, deux cas se présentent selon que le document avec le plus grand score est pertinent ou non.

- S'il est pertinent, faire tendre R vers 0 revient à relever le seuil d'acceptation pour enlever tous les documents sélectionnés jusqu'au dernier. Alors $S = X + Y = 0$et $P_0 = \frac{0}{0}$qui est indéfini.
- S'il ne l'est pas, lorsque tous les documents pertinents ont été enlevés de la sélection par un seuil suffisamment haut, il reste au moins un document sélectionné et $S \neq 0$mais $X = 0$ donc $P_0 = 0$.

1.3.9 précision moyenne

Les 21 valeurs correspondantes de la précision interpolée P_0, $P_{0,05}$, ..., P_1 peuvent être moyennées pour donner une estimation de l'efficacité de la méthode de classification, indépendamment du seuil.

Il est également possible de faire une moyenne de la manière suivante :

- partant d'un seuil très élevé, on le fait descendre petit à petit ;
- chaque fois qu'un nouveau document d pertinent est sélectionné, on calcule la précision (rapport du nombre de documents pertinents mieux classés que d au nombre de documents mieux classés que d) ;
- on fait ensuite la moyenne de toutes les valeurs trouvées.

$$M = \frac{1}{|E_c|} \sum_{d \in E_c} \frac{|\{d' \in E_c | Rang_{d'} \leq Rang_d\}|}{Rang_d}$$

1.3.10 Réponse à une question

1.3.10.1 inverse du rang

Lorsque le problème est de retrouver un seul document pertinent (par exemple un qui contient la réponse cherchée sur le web à une question donnée) dans un grand nombre de document, le classifieur sera d'autant meilleur que le document réponse apparaît plus tôt dans la liste ordonnée des réponses.

L'inverse du rang du document réponse est une mesure d'autant plus grande que le classifieur est bon. En fait, c'est la même valeur que la précision moyenne M ci-dessus.

Sil y a plusieurs documents pertinents, on fera la moyenne des valeurs trouvées, mais cette moyenne n'est, dans ce cas, pas égale à M.

1.3.10.2 top 10

Un moteur de recherche donne généralement une liste de résultat divisée en pages de 10 réponses. Il est agréable de trouver la réponse cherchée dans la première page de résultat.

On peut donc mesurer la qualité d'un classifieur par un booléen qui indique si la réponse (ou l'une des réponses) apparaît dans les 10 premiers résultats. On peut cependant préférer se limiter à 5 résultats, ce qui correspond à peu près au contenu d' un écran.

1.3.11 précision et rappels relatifs

Ces mesures sont destinées à évaluer la qualité de l'indice de confiance attaché à chaque classification.

voir Hatzivassiloglou 00

1.3.12 autres mesures +=+

1.4 Expérimentation

La classification supervisée nécessite la connaissance des classes de documents préclassifiés. On peut profiter de cette information pour faire des expériences, dans le but de trouver les meilleurs paramètres de la methode utilisée (voir **??**) ou simplement de se faire une idée de la qualité des résultats que l'on obtiendra lors de l'exploitation du classifieur.

1.4.1 dispositif expérimental

1.4.2 erreur d'apprentissage, erreur de généralisation

La qualité d'un classifieur peut être mesurée sur l'ensemble d'apprentissage. Cela nous dit à quel point la méthode utilisée permet de reproduire une clas-

sification connue. Si, idéalement, l'ensemble d'apprentissage est strictement représentatif de l'ensemble de tous les documents existants, cette mesure suffit.

Mais cette situation est pratiquement impossible à atteindre vu la taille énorme du nombre des termes éventuellement importants (vocabulaire) vocabulaire et son évolution fréquente dans le temps.

On mesure donc la qualité sur un autre ensemble, dit de test, pour voir à quel point le résultat obtenu peut se généraliser à d'autres documents. On constate pratiquement tout le temps que l'erreur de généralisation reste supérieure à l'erreur d'apprentissage.

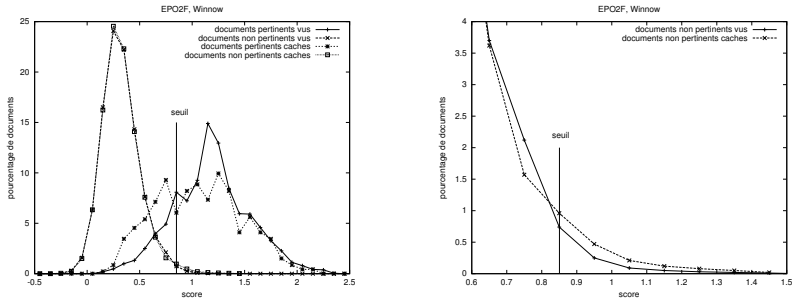

FIG. 1.4 – glissement des scores entre l'apprentissage et le test

La figure 1.4 met en évidence un glissement des scores qui explique la différence de qualité mesurée entre l'apprentissage et le test. Les courbes sont données en pourcentage pour améliorer la lisibilité mais en réalité, il y a 43 fois plus de documents non pertinents. C'est pourquoi, les surfaces dessinées sous les courbes, à droite et à gauche du seuil, ne sont pas égales (voir 1.3.3).

La courbe de gauche montre bien que, par rapport à la courbe des documents vus, la courbe des documents pertinents cachés s'écrase vers la gauche : un nombre plus important de documents pertinents reçoivent un score plus faible que le seuil (que ce qu'il faudrait pour que les courbes soient identiques). Cela s'explique par la présence dans les documents cachés de termes importants qui n'apparaissaient pas dans les documents vus. Ces termes (peur-être des synonymes d'autres termes) n'ont pas reçu le poids qu'ils mériteraient et les documents qui les contiennent sont pénalisés.

Le glissement des courbes des documents non pertinents se voit lorsque l'on agrandi la courbe correspondante, ce qui la rapproche de sa taille réelle proportionnelle au nombre de ces documents. Un nombre plus important de documents non pertinents reçoivent un score plus fort que le seuil : des termes ont reçu un poids trop fort ou, symétriquement au cas des pertinents, des termes négatifs importants (indicateurs de la non-appartenance à la classe) étaient absents des documents vus.

1.4.2.1 Surentraînement :

Un phénomène bien connu en apprentissage automatique est le *surentraînement* : si l'on apprend trop sur l'ensemble d'apprentissage, le classifieur obtenu est très bon sur cet ensemble mais, trop spécialisé, il donne de mauvais résultat sur un autre ensemble de documents. Donc, seules les méthodes qui font un apprentissage répétitif sont sujet à ce phénomène.

Winnow est de ce type. En itérant à volonté, on peut en général converger vers un classifieur qui reproduit parfaitement l'ensemble d'apprentissage, mais on voir sur la figure 1.5 qu'avec les itérations, les résultats deviennent moins bon sur l'ensemble de test.

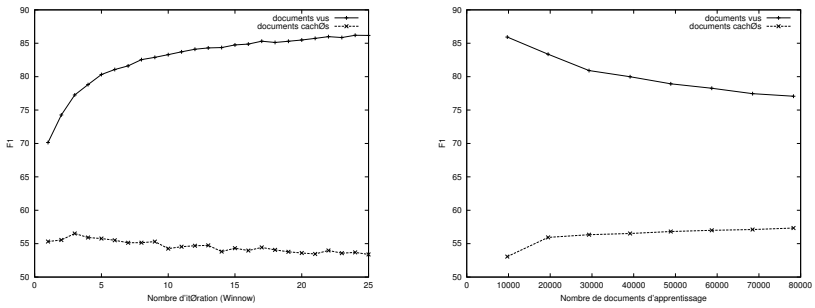

FIG. 1.5 – surentraînement de Winnow et courbe d'apprentissage progressif

Le phénomène est différent, mais le résultat est semblable, lorsque l'on apprend avec de plus en plus de documents. L'ensemble d'apprentissage étant modifié, on ne peut parler de surentraînement sur un ensemble.

La courbe à droite de la figure montre que l'erreur de test peut diminuer, et c'est souvent le cas. En effet, un des principaux problèmes que nous rencontrons est le nombre énorme de termes pouvant être utiles à la classification. L'augmentation du nombre de documents traités augmente la taille du vocabulaire connu et donc améliore la classification de nouveaux documents.

Par contre l'erreur d'apprentissage augmente avec le nombre de documents, car il est plus difficile de classifier un ensemble plus grand.

1.4.3 micro et macro moyennes

Lorsque l'on travaille avec plusieurs classes, il est nécessaire de résumer les mesures faites sur chaque classe en une seule valeur moyenne. Nous disposons en effet pour chaque classe c_j d'une précision P_j, d'un rappel R_j, d'une valeur $F_j(\gamma)$, etc. Il y a 2 façon de moyenner les valeurs vues dans les paragraphes précédents.

macro-moyenne : pour calculer cette valeur, on considère chaque classe comme un élément indivisible et on fait la moyenne des mesures sur chaque classe.

$$P = \frac{\sum_{j=1,m} P_j}{m}$$

$$R = \frac{\sum_{j=1,m} R_j}{m}$$

Pour $F(\gamma)$, on peut procéder de 2 manières :

- moyenner les $F_j(\gamma)$;
- calculer la valeur finale à partir des P et R moyennés :

$$F(\gamma) = \frac{(\gamma^2 + 1)PR}{\gamma^2 P + R}$$

micro-moyenne : on considère chaque exemple séparément pour calculer les valeurs X, Y, Z et W puis on en déduit les valeurs qui nous intéressent (P, R, F, ...).

En séparant les documents, la micro-moyenne donne plus d'importance aux classes les plus nombreuses, mais cela correspond bien à l'impression que le classifieur donne à un utilisateur car cela reflète le nombre de document mal classifiés dans un sens ou dans un autre.

On remarquera cependant que :

- si toutes les classes contiennent le même nombre de documents, la micro-moyenne d'une valeur est égale à sa macro-moyenne ;
- en monoclassification, comme chaque document appartient à une classe et une seule, s'il manque dans sa classe, il est en trop dans une autre classe. donc, lorsque l'on totalise les erreurs, $Z = Y$, ce qui implique $P = R$.