De la classification d'opinions à la recommandation : l'apport des textes communautaires

Damien Poirier[*,**] — **Françoise Fessant**[*] — **Isabelle Tellier**[**]

[*] *Orange Labs*
2 avenue Pierre Marzin, 22300 Lannion, FRANCE
prénom.nom@orange-ftgroup.com

[**] *Laboratoire d'Informatique Fondamentale d'Orléans*
Rue Léonard de Vinci, 45000 Orléans, FRANCE
prénom.nom@univ-orleans.fr

RÉSUMÉ. *Cet article s'intéresse à la classification d'opinions de textes communautaires par apprentissage supervisé, en vue de les utiliser pour un système de recommandation. Nous comparons différents prétraitements, représentations et techniques d'apprentissage sur des données réelles parlant de films et présentant diverses particularités (textes très courts en anglais, contenant beaucoup de codes type sms, d'abréviations, de fautes d'orthographe, etc.). Nous étudions en détails les résultats de différents classifieurs ainsi que l'apport des prétraitements sur ce type de données. Pour finir, nous évaluons les résultats du meilleur classifieur à l'aide d'un moteur de recommandation de type filtrage collaboratif.*

ABSTRACT. *This paper is about opinion classification of posts from a social networks by supervised machine learning, in order to use them in a recommender system. We compare different pre-processings, representations and machine learning tools on real data about movies having specificities (very short texts in English, containing a lot of sms-like codes, abbreviations, misspelling...). We study in detail the results of different classifiers and the contribution of the pre-processings on this kind of data. Finally, we evaluate the best classifier with a recommender system based on collaborative filtering.*

MOTS-CLÉS : *classification d'opinions, apprentissage supervisé, textes communautaires, cyberlangue, recommandation automatique, filtrage collaboratif, cold-start.*

KEYWORDS: *opinion classification, supervised learning, texts from social networks, cyberlangage, recommendation, collaborative filtering, cold-start.*

1. Introduction

Les systèmes de recommandation automatique sont devenus, à l'instar des moteurs de recherche, un outil incontournable pour tout site Web focalisé sur un certain type d'articles disponibles dans un catalogue riche, que ces articles soient des objets, des produits culturels (livres, films, morceaux de musique, etc.), des éléments d'information (news) ou encore simplement des pages (liens hypertextes). L'objectif de ces systèmes est de sélectionner, dans leur catalogue, les articles (ou items) les plus susceptibles d'intéresser un utilisateur particulier. Nageswara Rao et Talwar (2008) ont répertorié un vaste ensemble de systèmes de recommandation pour différents domaines applicatifs, dans des contextes académiques et industriels. Le travail présenté ici vise à faciliter la mise en place d'un système de ce type pour le service VOD (*Video On Demand*) du portail Orange. Différentes approches peuvent être mises en œuvre pour cela, mais toutes ont la particularité de requérir un minimum de données de départ sur lesquelles les algorithmes vont pouvoir s'appuyer. Or, ces données ne sont pas toujours disponibles, ou pas en assez grande quantité, surtout quand le service de recommandation vient d'être mis en place.

Parallèlement, les données présentes sur l'Internet ne cessent de s'enrichir depuis l'apparition du Web 2.0, grâce au contenu produit par les utilisateurs (*User Generated Content*). Ces données, majoritairement textuelles, sont encore très peu utilisées en recommandation. Pourtant le contenu généré par les utilisateurs contient énormément de commentaires subjectifs sur des articles de catalogues de toutes sortes. Le site communautaire Flixster[1] est un exemple de site participatif où se retrouvent chaque jour des dizaines de millions de fans de cinéma pour partager leurs impressions et sentiments sur des films. Le site Twitter[2] contient également des avis, uniquement sous forme de commentaires textuels, sur des sujets beaucoup plus variés. Beaucoup de forums également présentent ce type d'informations. La fouille de données d'opinions (Pang et Lee, 2008) est un domaine qui s'est beaucoup développé ces dernières années dans le but d'extraire des informations utiles de ce genre de textes.

Nous avons donc d'un côté des systèmes de recommandation qui peuvent manquer de données, et d'un autre des textes riches d'informations non encore exploitées. Notre objectif dans ce travail est d'utiliser la fouille d'opinions pour alimenter un système de recommandation. C'est une idée simple et naturelle, mais elle a donné lieu à encore très peu de publications, sans doute parce que les deux domaines relèvent de communautés de recherche différentes et que les expérimentations requises par chacun d'eux sont longues à mener. Les seuls travaux répertoriés sur le sujet exploitent un moteur de recommandation thématique, dans lequel les articles à recommander sont décrits par un ensemble d'attributs. Prendre comme attributs les mots de textes qui parlent de ces articles est alors effectivement une solution possible. Mais les moteurs de recommandation les plus efficaces procèdent plutôt par filtrage collaboratif, une méthode qui se fonde non pas sur des attributs mais sur des notes attribuées par des utilisateurs

1. www.flixster.com
2. http ://twitter.com/

à des articles. Enchaîner un système d'affectation de notes par fouille d'opinions et un système de recommandation par filtrage collaboratif sur des données réelles est notre principale contribution dans cet article.

Il est organisé de la manière suivante : tout d'abord nous introduisons le domaine de la recommandation et les différentes approches possibles pour la mener à bien. Nous présentons aussi la mesure classique employée pour évaluer la qualité d'un système de recommandation, ainsi que le moteur utilisé pour nos expériences. Celui-ci est capable de s'appuyer aussi bien sur des attributs que sur des notes, ce qui nous donnera l'occasion de pouvoir comparer les différentes méthodes. Cette partie se clôt avec le schéma de la chaîne de traitements mise en œuvre dans nos expériences. La partie suivante dresse un bref panorama de la classification automatique de textes selon l'opinion qu'ils expriment, en détaillant tous les choix possibles que nous avons pris en compte. Puis, nous exposons les conditions de nos expériences, en détaillant les particularités du corpus qui a servi à les mener. Les textes récupérés sont très spécifiques. Ils sont en général très courts (une dizaine de mots) et le style dans lequel ils sont rédigés se rapproche de celui utilisé dans les forums ou dans les systèmes de messagerie instantanée. Différents traitements et outils couramment utilisés dans la littérature sont testés et évalués sur ces données particulières. Nos expériences visent à identifier quels sont les prétraitements pertinents pour ce type de textes et leur impact sur le classifieur. Enfin, la dernière partie est celle où le meilleur de nos classifieurs appris est mis à contribution pour alimenter un système de recommandation par filtrage collaboratif. Nous montrons que les résultats qu'il fournit permettent de produire des recommandations de très bonne qualité, meilleures que celles qui se contentent d'attributs pourtant riches et variés. Cette partie montre également que l'apport de données extérieures pour la mise en route d'un moteur de recommandation est tout à fait pertinent.

2. Les systèmes de recommandation

Le but des systèmes de recommandation est de prédire l'affinité entre un utilisateur et un article, en se fondant sur un ensemble d'informations déjà acquises sur cet utilisateur et sur d'autres, ainsi que sur cet article et sur d'autres. Nous présentons tout d'abord les différentes approches possibles du domaine, avant d'évoquer les travaux ayant déjà essayé d'exploiter des textes pour faire de la recommandation.

2.1. *Les différents systèmes de recommandation*

Il existe plusieurs familles de systèmes de recommandation (Nageswara Rao et Talwar, 2008), en fonction de la manière dont la recommandation est effectuée et de la nature des données :

– **le filtrage basé sur le contenu**, ou filtrage thématique, s'appuie sur le contenu des articles par le biais de descripteurs, pour les comparer à un profil utilisateur lui-

même constitué des mêmes descripteurs (Pazzani et Billsus, 2007). Lors de l'arrivée d'un nouvel article, le système compare sa représentation avec le profil de l'utilisateur, afin de prédire l'opinion qu'il aura de lui. Un article est recommandé en fonction de sa proximité avec le profil de l'utilisateur ;

– **le filtrage collaboratif** s'appuie sur les appréciations données par un ensemble d'utilisateurs sur un ensemble d'articles. Ces appréciations, traduites en valeurs numériques, peuvent être des notes, des comptes d'achats effectués, des nombres de visites, etc. On distingue deux grandes approches de filtrage collaboratif. L'approche se référant aux utilisateurs (Resnick *et al.*, 1994) consiste à comparer les utilisateurs entre eux et à retrouver ceux ayant des goûts en commun, les notes d'un utilisateur étant ensuite prédites selon son voisinage. L'approche se référant aux articles (Sarwar *et al.*, 2001) consiste à rapprocher les articles appréciés par les mêmes personnes et à prédire les notes des utilisateurs en fonction des articles les plus proches de ceux qu'ils ont déjà notés ;

– **le filtrage hybride** exploite aussi bien les évaluations numériques que les descripteurs de contenu. Les systèmes hybrides peuvent également faire appel à des sources d'informations complémentaires telles que des données démographiques ou sociales (Pazzani, 1999). Différentes méthodes d'hybridation peuvent être envisagées afin de combiner les sources ou les modèles. On peut par exemple appliquer séparément le filtrage collaboratif et d'autres techniques de filtrage pour générer des recommandations candidates, et combiner ces ensembles de recommandations par pondération, cascade, bascule, etc. afin de produire les recommandations finales pour les utilisateurs (Burke, 2007). La pondération consiste à combiner les prédictions obtenues par des sources de données différentes ou des modèles différents et d'utiliser un système de vote pour la prédiction finale des recommandations. L'hybridation par bascule ne se propose pas de combiner différents résultats de systèmes de recommandation, mais d'en sélectionner un dynamiquement, au moment de la recommandation, selon un critère donné. Le mode d'hybridation en cascade est un mode strictement hiérarchique où chaque recommandeur en aval ne fait que raffiner la recommandation obtenue par le recommandeur précédent.

Les différents types de filtrage ont leurs forces et leurs faiblesses et il est nécessaire pour le concepteur du système de choisir la stratégie la plus adaptée à son domaine applicatif et à ses données. L'avantage du filtrage basé sur le contenu est que chaque utilisateur est absolument indépendant des autres. Ainsi, un utilisateur pourra recevoir des recommandations même s'il est le seul inscrit, pour peu qu'il ait décrit son profil en donnant un ensemble de thèmes qui l'intéressent. En revanche, cette technique de filtrage est soumise à un effet « entonnoir », car le profil évolue naturellement par restrictions progressives des thèmes recherchés. Ainsi, l'utilisateur ne reçoit que les recommandations relatives aux thèmes déjà présents dans son profil, une fois celui-ci devenu stable. Par conséquent, il risque de ne pas découvrir de nouveaux domaines potentiellement intéressants pour lui.

À l'opposé, tous les utilisateurs d'un système reposant sur le filtrage collaboratif peuvent tirer profit des évaluations des autres, sans que le système dispose pour au-

tant d'une représentation du contenu des articles. Chacun bénéficie en quelque sorte de l'avis des autres, et ceci d'autant plus que les goûts sont partagés. Grâce à son indépendance vis-à-vis de la représentation des données, cette technique peut s'appliquer dans les contextes où le contenu est soit indisponible, soit difficile à analyser. Elle peut ainsi s'utiliser pour tous les types de données : texte, image, audio et vidéo. De plus, l'utilisateur est susceptible grâce à elle de découvrir divers domaines intéressants, car le filtrage collaboratif n'est pas soumis à l'effet « entonnoir ». Un autre avantage du filtrage collaboratif est que les jugements de valeur des utilisateurs intègrent non seulement la dimension thématique mais aussi d'autres facteurs relatifs à la qualité des articles tels que la diversité, la nouveauté, l'adéquation au public visé, etc. Le filtrage collaboratif est ainsi le type de filtrage le plus fréquemment utilisé.

Mais ce type de filtrage est aussi plus contraignant que le filtrage basé sur le contenu car il implique plus fortement les utilisateurs, qui doivent attribuer des notes aux articles. De plus, le bon fonctionnement d'un tel système et sa performance reposent sur une grande quantité d'information. Les utilisateurs doivent évaluer suffisamment d'articles pour dépasser le problème du *démarrage à froid* (Schein *et al.*, 2002). En effet, les préférences d'un utilisateur doivent être déjà suffisamment riches pour pouvoir se comparer aux autres. Les utilisateurs doivent par ailleurs être en suffisamment grand nombre pour atteindre une certaine masse critique au-delà de laquelle les calculs de prédiction deviennent pertinents. Les évaluations doivent concerner des ensembles d'articles et d'utilisateurs qui se recoupent, afin de permettre au système de comparer les préférences. Par exemple, on ne peut pas conclure que deux personnes sont proches si elles n'ont qu'une seule évaluation en commun.

Dans (Meyer, 2011), l'auteur compare les performances de systèmes de recommandation fondés sur des sources de données collaboratives, thématiques et hybrides pour différentes situations : cas d'un système déjà monté en charge, cas d'un système débutant avec des profils utilisateurs peu renseignés. Il conclut que les systèmes collaboratifs semblent indépassables en termes de performances lorsque les utilisateurs ont des profils longs, mais sont moins compétitifs si les profils des utilisateurs sont peu renseignés. Des stratégies d'hybridation de différentes méthodes ou de différentes sources (par l'apport de sources de données complémentaires par exemple) sont pertinentes pour le problème du démarrage à froid. L'approche proposée dans cet article s'inscrit dans cette optique en démontrant que des données textuelles provenant du Web 2.0 peuvent permettre d'éviter le passage forcé au filtrage basé sur le contenu lors d'un démarrage à froid et ainsi réduire la baisse de performance qui en découle.

2.2. *Description du moteur et indicateur d'évaluation utilisé*

Le système de recommandation utilisé dans le cadre de nos expériences a été développé à France Telecom (Candillier *et al.*, 2008). Il permet de faire à la fois du filtrage collaboratif par une approche se référant aux articles et du filtrage thématique. Dans tous les cas, il s'appuie sur une matrice de distances entre articles. Les modèles de recommandation de ce type sont actuellement ceux qui procurent le maxi-

mum d'avantages en termes de couverture fonctionnelle et de qualité prédictive, mais aussi de tenue de charge, de transparence et de réactivité aux changements de profils (Koren, 2010). Ils sont dans le domaine industriel très avantageux par rapport aux modèles à factorisation de matrice et aux modèles d'apprentissage.

Dans le cas du filtrage collaboratif fondé sur les articles, la matrice de distances entre articles est construite à l'aide de la fonction de similarité de Pearson dont nous rappelons la formule :

$$Pear(i,j) = \frac{\sum_{\{u \in S_i \cap S_j\}} (r_{iu} - \overline{r_i}) \times (r_{ju} - \overline{r_j})}{\sqrt{\sum_{\{u \in S_i \cap S_j\}} (r_{iu} - \overline{r_i})^2 \sum_{\{u \in S_i \cap S_j\}} (r_{ju} - \overline{r_j})^2}} \quad [1]$$

où S_i (respectivement S_j) est l'ensemble des notes obtenues par l'article i (respectivement j), r_{iu} (respectivement r_{ju}) la note donnée par l'utilisateur u sur l'article i (respectivement j), et $\overline{r_i}$ (respectivement $\overline{r_j}$) la moyenne des notes obtenues par i (respectivement j).

Dans le cas du filtrage thématique, c'est la distance de Jaccard qui a été préférée :

$$Jacc(i,j) = \frac{|\{S_i \cap S_j\}|}{|\{S_i \cup S_j\}|} \quad [2]$$

où S_i (respectivement S_j) est l'ensemble des descripteurs de l'article i (respectivement j).

Une fois la matrice de similarité articles $\times$ articles construite, les recommandations sont établies à l'aide des notes connues des utilisateurs, ces notes constituant leurs profils. La fonction de prédiction de la note de l'article i pour l'utilisateur u est la suivante suivante :

$$p_{ui} = \overline{r_i} + \frac{\sum_{\{j \in S_u\}} sim(i,j) \times (r_{uj} - \overline{r_j})}{\sum_{\{j \in S_u\}} sim(i,j)} \quad [3]$$

où S_u est l'ensemble des notes connues données par l'utilisateur u et la fonction $sim(i,j)$ est soit égale à $Pear(i,j)$ soit égale à $Jacc(i,j)$ suivant le type de filtrage considéré. Le prédicteur tient ainsi compte des moyennes des notes déjà attribuées par les autres utilisateurs aux différents articles.

La RMSE (*Root Mean Squared Error*) est la mesure utilisée afin de vérifier la qualité d'un système de recommandation. Elle mesure l'erreur faite entre la note prédite et la vraie note donnée par l'utilisateur. Elle se calcule de la façon suivante :

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\overline{X_i} - X_i)^2}{n}} \quad [4]$$

où X_i représente la note réelle et $\overline{X_i}$ la note prédite par le recommandeur.

La RMSE étant une mesure d'erreur, on attend d'elle qu'elle soit la plus basse possible. Afin d'avoir une idée de ce que représente une RMSE en recommandation automatique, précisons que pour gagner le million de dollars du concours Netflix, il fallait obtenir une RMSE de 0,85 (le moteur utilisé sur le site avait alors une RMSE

d'environ 0,95). Atteindre cette performance a demandé presque trois années de travail aux équipes de recherche et a entraîné la construction de systèmes très complexes et très gourmands en puissance de calcul (et donc inexploitables dans un contexte industriel). Il est évident que nous n'espérons pas nous mesurer à ces performances mais leur connaissance permettra tout de même de situer nos résultats.

2.3. *Exploiter des textes d'opinions en recommandation*

La problématique de notre travail est d'exploiter des textes porteurs d'opinions afin d'alimenter un système de recommandation. À notre connaissance, tous les systèmes de recommandation existants fondés sur des textes non structurés s'appuient sur des méthodes liées au filtrage thématique (Pazzani et Billsus, 2007). Or, la brève revue que nous venons de faire de ces systèmes suggère immédiatement deux approches possibles : on peut soit extraire des textes des descripteurs sur lesquels s'appuiera un filtrage thématique, soit traduire en une évaluation numérique l'opinion qu'ils véhiculent, afin de mettre en œuvre un filtrage collaboratif. Nous savons également que les systèmes exploitant le filtrage collaboratif sont bien plus performants que ceux exploitant le filtrage thématique. Nous nous concentrerons donc dans cet article sur la deuxième approche, qui nécessite d'enchaîner une tâche de classification d'opinions avec une tâche de filtrage collaboratif. Cette idée a été évoquée précédemment (Cane *et al.*, 2006 ; Dziczkowski et Wegrzyn-Wolska, 2007) mais jamais testée explicitement. Nous sommes donc, à notre connaissance, les premiers à mener de telles expériences de bout en bout. Concernant l'unique tâche de classification d'opinions, l'originalité de nos travaux porte sur les données étudiées. Elles sont en effet particulières, étant rédigées dans un style très proche du discours électronique médié (Panckhurst, 2006) (appelé également Cyberlangue). En cela, les textes contiennent une quantité non négligeable d'erreurs orthographiques, typographiques ou grammaticales, des abréviations, des néographies, des didascalies électroniques, etc.

La démarche suivie pour ces expérimentations se décompose donc en deux étapes (voir figure 1). Une première étape, la construction des données d'usage, consiste à transformer des données textuelles en notes. Cela correspond à une tâche de classification d'opinions. Les données textuelles utilisées sont toutes reliées à un auteur et à un article, en l'occurrence ici un film. Le résultat obtenu en sortie de cette première étape est une matrice d'usage contenant des triplets utilisateur-film-note. Ces données permettent, dans un second temps, d'établir des recommandations à l'aide d'une méthode de filtrage collaboratif s'appuyant sur les distances entre articles. Les triplets sont en effet utilisés afin de construire une matrice de similarité films $\times$ films contenant des mesures de distance entre chaque article. Des notes sont ensuite prédites pour chaque couple (utilisateur, film) pour lequel la note n'est pas encore renseignée.

La section suivante présente un état de l'art de la classification d'opinions, qui sera la première tâche à accomplir avec nos données.

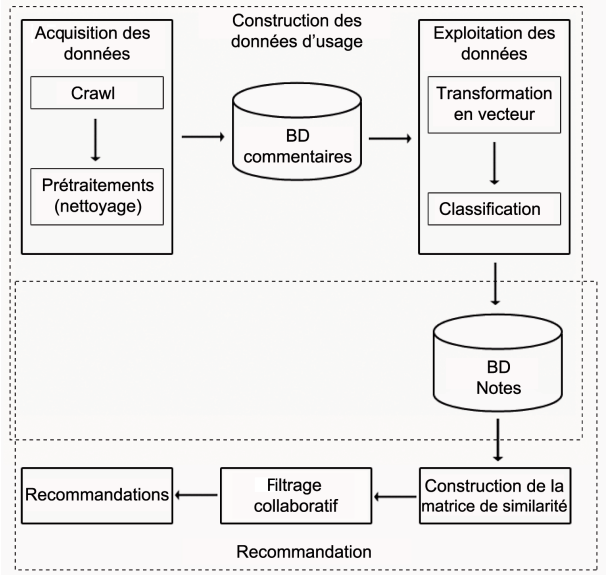

Figure 1. *Chaîne de traitements*

3. La classification de textes d'opinions

Nous introduisons d'abord les différents éléments qui interviennent dans une tâche de classification de textes d'opinions par apprentissage automatique, puis nous détaillons les choix effectués pour ces différents éléments dans notre contexte.

3.1. *Introduction au domaine*

Rappelons tout d'abord que la fouille d'opinions se compose de plusieurs tâches (Pang et Lee, 2008) : l'extraction d'opinions, le résumé d'opinions et la classification d'opinions :

– l'extraction d'opinions consiste à identifier dans un corpus les textes porteurs d'opinions, ou encore à localiser les passages porteurs d'opinions dans un texte. Plus précisément, on se préoccupe ici de classer les textes ou les parties de textes selon qu'ils sont objectifs ou subjectifs ;

– le résumé d'opinions consiste à rendre l'information rapidement et facilement accessible en mettant en avant les opinions exprimées et les cibles de ces opinions présentes dans un texte. Ce résumé peut être textuel (extraction des phrases ou expressions contenant les opinions), chiffré (pourcentage, note), graphique (histogramme) ou encore imagé (thermomètre, étoiles, pouce levé ou baissé...) ;

– la classification d'opinions a pour objectif d'attribuer une étiquette à un texte selon l'opinion qu'il exprime.

Puisque nous avons besoin d'associer des notes à des textes, nous nous intéressons ici uniquement à la classification d'opinions. Deux grands types de méthodes sont utilisés pour cette tâche. Il y a tout d'abord les approches plutôt *linguistiques* qui consistent à répertorier le vocabulaire porteur d'opinions, puis à établir des règles de classification selon la présence ou l'absence des mots appartenant à ce vocabulaire. Il existe également les approches mettant en œuvre des outils issus du domaine de l'*apprentissage automatique*. Nous avons déjà dans un travail précédent comparé ces deux approches (Poirier *et al.*, 2009) ; nous nous intéressons ici uniquement à celles de la deuxième famille, qui sur nos données se sont avérées nettement plus efficaces. Les méthodes utilisées dans ce cadre sont issues de la classification dite supervisée (ou apprentissage supervisé), où un classifieur est appris à l'aide d'exemples de données (ici de textes) dont on connaît déjà la classe (ici la note). Les mots des textes sont alors généralement considérés comme des données indépendantes et équivalentes les unes aux autres, leur sémantique n'étant pas explicitement prise en compte. On peut donner une définition un peu plus formelle du problème de la classification supervisée.

Définition : *soit X un ensemble de données, Y un ensemble d'étiquettes (ou classes) et D un ensemble des représentations des données. Soit $d : X \longrightarrow D$, une fonction connue qui associe à chaque donnée $x \in X$ une représentation $d(x) \in D$ et $S \subset D \times Y$ un ensemble de données étiquetées $(d(x), y)$. La classification supervisée consiste à construire en s'appuyant sur l'ensemble S un* classifieur, *c'est-à-dire une fonction de $D \longrightarrow Y$ qui permette de prédire la classe de toute nouvelle donnée $x \in X$, représentée par $d(x) \in D$.*

D'après cette définition, quatre éléments distincts entrent en jeu dans la classification supervisée :

– la représentation des données (l'ensemble D) ;

– les étiquettes ou classes de prédiction (l'ensemble Y) ;

– les exemples de données étiquetées, qui constituent le corpus d'apprentissage (l'ensemble S) ;

– le classifieur ou prédicteur.

De plus, des prétraitements peuvent être appliqués sur les données avant la tâche de classification dans le but d'améliorer ses performances, que ce soit en termes de résultats ou de temps de calcul. Ils sont en général intégrés dans la définition de la représentation des données.

3.2. *Les différents choix possibles étudiés*

Nous détaillons maintenant chacun des éléments identifiés dans la section précédente, en retardant légèrement celui de la représentation des données, dont la définition dépend beaucoup des autres.

3.2.1. *Le corpus d'apprentissage*

La classification supervisée nécessite des exemples (données étiquetées) afin de construire le « corpus d'apprentissage ». Ce corpus ayant un impact direct sur l'apprentissage des règles, et par conséquent sur la classification, il est nécessaire que les exemples soient représentatifs de l'ensemble des données. Cette hypothèse est généralement difficile à vérifier. En classification d'opinions, ou plus généralement en classification de textes, les corpus étudiés sont assez restreints car l'étiquetage des exemples est souvent effectué à la main. Ceci entraîne un coût élevé et ne permet donc pas l'obtention d'un gros corpus d'apprentissage. Cependant, les données présentes sur les sites Web 2.0, qui peuvent être étiquetées par les utilisateurs, permettent aujourd'hui de traiter des corpus beaucoup plus conséquents.

3.2.2. *Les classes de prédiction*

Concernant le choix des classes, il est généralement imposé par le corpus d'apprentissage. La classification est binaire lorsque le nombre de classes $|Y|$ est égal à 2. Il peut naturellement être supérieur, mais l'augmentation du nombre de classes augmente par conséquent le taux d'erreurs. En classification d'opinions, le nombre de classes de prédiction choisi est généralement de 2 (aime, n'aime pas) ou 3 (aime, avis neutre, n'aime pas) (Seki *et al.*, 2007 ; Seki *et al.*, 2008 ; Ounis *et al.*, 2006 ; Macdonald *et al.*, 2007 ; Ounis *et al.*, 2008 ; Grouin *et al.*, 2007).

3.2.3. *Les prétraitements*

Les prétraitements les plus couramment utilisés ont deux objectifs. Dans un premier temps, ils servent à réduire la taille du vocabulaire, et par conséquent de l'espace de représentation. Dans un deuxième temps, les prétraitements peuvent permettre d'améliorer les performances du système de prédiction des opinions. On fait généralement en sorte que le premier objectif, la réduction de l'espace de représentation, n'empiète pas sur les performances de la classification. Il s'avère que la réduction de l'espace peut également être une manière d'améliorer les performances, notamment en regroupant certaines variables du vocabulaire pour ainsi accentuer leur influence. C'est par exemple l'objectif principal de la lemmatisation, qui consiste à remplacer chaque mot du texte par sa forme canonique conventionnelle (lemme). La suppression de certains mots ou caractères ne véhiculant pas ou peu d'information peut également permettre d'atteindre les deux objectifs. Il n'est pas rare que la ponctuation, les chiffres, les symboles ou encore les articles indéfinis soit retirés du corpus. On peut également utiliser des antidictionnaires (*stop-list* en anglais). Ces antidictionnaires sont des lexiques contenant des mots non nécessaires pour la tâche étudiée, parce qu'ils ne sont généralement pas porteurs de sémantique pour le critère de classification. On peut également réduire les dimensions par le biais de méthodes statistiques comme TF-IDF, χ^2 ou encore LSA (*Latent Semantic Analysis*).

D'autres prétraitements, comme l'étiquetage des mots par leur catégorie grammaticale, sont utilisés afin d'apporter de l'information supplémentaire. Ce type de prétraitements enrichit la représentation au lieu de la simplifier.

3.2.4. *La représentation*

Un document textuel est généralement représenté sous la forme dite « sac de mots ». Cette représentation suppose de sélectionner des éléments de vocabulaire représentant les dimensions d'un espace vectoriel, et de représenter chaque document par un vecteur dans cet espace. Ceci implique deux choix principaux :

– un document est tout d'abord une séquence de caractères. La première étape consiste donc à segmenter cette séquence afin d'obtenir un ensemble d'éléments qui constitueront le vocabulaire de la représentation. Différents choix sont possibles suivant le délimiteur choisi. Les mots sont les éléments de vocabulaire les plus utilisées en classification d'opinions mais d'autres choix existent. On peut par exemple limiter la taille des éléments à un nombre n de caractères, on obtiendra alors des n-grammes de lettres. Des n-grammes de mots peuvent aussi être sélectionnés, afin de préserver un peu de sémantique. Par exemple, dans la phrase « je n'aime pas ce film », le bigrammes « aime pas » est très informatif. Mais les bigrammes de mots donnent lieu à des espaces de trop grandes dimensions et sont rarement utilisés en classification d'opinions ;

– une fois le vocabulaire (et donc l'espace vectoriel) sélectionné, représenter un document par un vecteur dans cet espace peut se faire de diverses manières. Les représentations les plus courantes sont :

- la représentation binaire (Crestan *et al.*, 2007 ; Nigam et Hurst, 2006 ; Pang et Lee, 2004) est la moins coûteuse en temps de calcul. Elle se contente d'enregistrer, pour un document, quels éléments du vocabulaire sont présents (valeur égale à 1) ou absents (valeur égale à 0),

- la représentation fréquentielle (Plantié *et al.*, 2008 ; Pang *et al.*, 2002) est une extension de la représentation binaire qui prend en compte le nombre d'occurrences des éléments du vocabulaire dans chaque document. Un texte est donc représenté par un vecteur dont chaque composante correspond au nombre de fois où un de ces éléments est présent dans le texte,

- la représentation fréquentielle normalisée est une représentation qui normalise les vecteurs de représentation des textes par leur longueur. Les nombres d'occurrences obtenus avec la représentation fréquentielle sont donc remplacés par des mesures de proportion des éléments du vocabulaire dans chaque document,

- la représentation TF-IDF (Généreux et Santini, 2007 ; Trinh, 2007), enfin, qui établit un compromis entre la fréquence d'un élément du vocabulaire dans le document considéré et sa présence dans tous les autres documents du corpus.

3.2.5. *Le classifieur*

Beaucoup de méthodes de classification supervisée existent et beaucoup d'entre elles ont été testées pour la classification d'opinions. On peut citer les arbres de décision, les réseaux de neurones, la régression logistique, les règles de décision ainsi que des méthodes combinant différents classifieurs comme les systèmes de votes ou les

algorithmes de Boosting. Toutefois, les méthodes les plus présentes dans la littérature, et qui semblent également être les plus performantes sur les textes, sont les machines à vecteurs supports (Pang et Lee, 2004 ; Kaiser *et al.*, 2010 ; Wilson *et al.*, 2004 ; Nigam et Hurst, 2006 ; Généreux et Santini, 2007 ; Trinh, 2007 ; Crestan *et al.*, 2007 ; Plantié *et al.*, 2008) et les classifieurs bayésiens naïfs (NB) (Plantié *et al.*, 2008 ; Maurel *et al.*, 2007 ; Pang et Lee, 2004 ; Yu et Hatzivassiloglou, 2003). Ce sont ces deux techniques que nous emploierons dans nos expériences. Les machines à vecteurs supports, appelés encore séparateurs à vaste marge (SVM), sont des classifieurs binaires. Pour les adapter au contexte d'une classification multiclasse, plusieurs stratégies sont applicables. Nous avons adopté ici le principe du *un-contre-tous* qui consiste à apprendre un modèle à l'aide de la première classe face à toutes les autres, puis de la deuxième classe face à toutes les autres, ainsi de suite, et à fusionner les modèles obtenus. Les NB peuvent, eux, directement opérer des classifications sur un nombre quelconque de classes.

4. Les données d'apprentissage et leurs particularités

4.1. *Particularités des données*

Les données utilisées pour nos expérimentations sont extraites du site communautaire *Flixster*. Ce site, fréquenté par des millions d'internautes, est dédié aux amateurs de cinéma. Il permet à ses utilisateurs de créer et d'alimenter des pages personnelles (blogs) où ils peuvent partager leurs opinions via des boîtes à réactions, des commentaires, des forums, des listes de favoris, etc. Les utilisateurs peuvent également se déclarer « amis » avec d'autres utilisateurs et se constituer ainsi un réseau social virtuel. Ils ont également la possibilité de commenter et de noter chaque film indépendamment, et ceci une fois seulement par film. Ainsi, chaque utilisateur actif présent sur le site est relié à un certain nombre de films, chaque couple (utilisateur, film) correspondant à au plus un unique commentaire. Ce sont ces commentaires qui constituent la partie textuelle de notre corpus. Ils sont généralement associés à une note postée par l'utilisateur lors de sa rédaction. Cette note, comprise entre 0,5 et 5, est censée résumer l'opinion portée sur le film en question. Nous pouvons donc logiquement considérer que ces commentaires, tout du moins la grande majorité d'entre eux, sont des textes subjectifs porteurs d'une opinion sur un film. Pour chaque utilisateur considéré nous avons systématiquement collecté, en plus des commentaires, les notes qui y sont associées.

La répartition de ces notes sur le site est très inégale. Les utilisateurs ont tendance à rédiger beaucoup plus de commentaires positifs que de commentaires négatifs. La figure 2 présente la répartition des classes d'un ensemble de commentaires pris au hasard. Afin de ne pas introduire de biais dans la comparaison des résultats des différentes expérimentations, nous avons fait le choix d'équilibrer la représentation des notes dans les corpus. Sinon, le classifieur consistant à choisir systématiquement la classe majoritaire aurait de bons résultats, ce que nous ne souhaitons pas. Pour cela, nous avons réparti les dix notes possibles dans cinq classes (1 à 5) puis nous avons

équilibré ces cinq classes en sélectionnant au hasard un nombre défini de commentaires pour chaque classe. Nous avons donc extrait deux corpus. Le premier, dédié à la tâche d'apprentissage, contient 175 000 commentaires et le deuxième, réservé aux tests, en contient 50 000. Une validation croisée n'est pas nécessaire sur ces données, parce que les ensembles d'apprentissage et de test sont de taille largement suffisante.

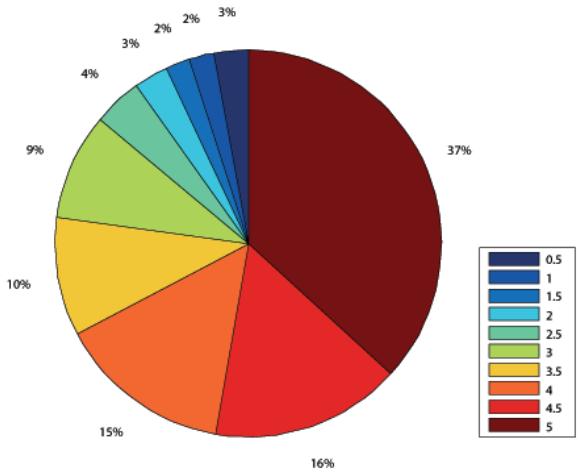

Figure 2. *Répartition des commentaires selon la note attribuée par l'auteur*

4.2. *Particularités des textes communautaires*

Les textes étudiés présentent certaines caractéristiques qui les différencient grandement des critiques écrites par des professionnels, souvent étudiées en classification d'opinions. Ils sont en général extrêmement courts avec 14 mots en moyenne par commentaire. La figure 3, faite à partir du corpus d'apprentissage, nous montre que les commentaires peuvent avoir des tailles assez variables, allant de 1 à 10 000 mots, mais que les commentaires longs sont très rares. La quantité de commentaires contenant plus de 100 mots est inférieure à 3 %. Elle nous montre également que la très grande majorité des textes (90 % d'entre eux) contiennent moins de 30 mots, et que la moitié (50 % d'entre eux) en contiennent moins de 6. Les mots considérés pour mesurer la taille des commentaires sont les chaînes de caractères alphanumériques. Les séparateurs sont le caractère espace ainsi que toutes les ponctuations, y compris l'apostrophe.

Mise à part la variabilité de taille, la variabilité des styles d'écriture est également une particularité de ces textes communautaires. Il y a des textes bien écrits et facilement compréhensibles pour un être humain, comme on peut le voir dans le tableau 1

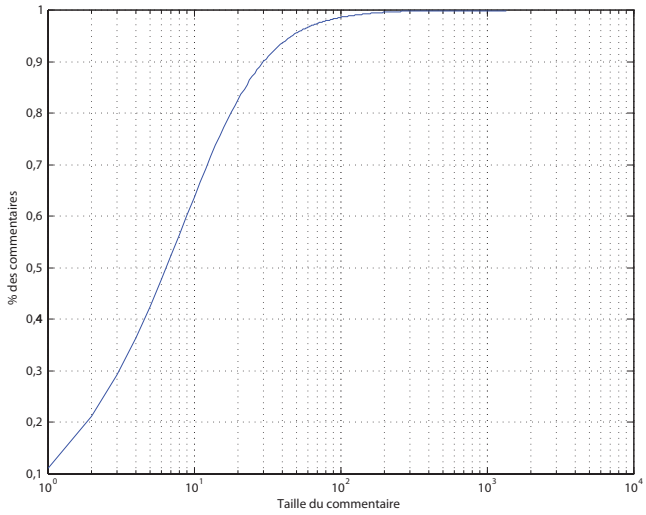

Figure 3. *Nombre de mots par commentaire*

mais le style employé est souvent plus proche du langage oral que du langage écrit et parfois difficilement déchiffrable. La grammaire s'écarte de la norme en vigueur et les fautes d'orthographe, volontaires ou non, peuvent venir compliquer la lecture. On peut également noter l'utilisation récurrente d'onomatopées, d'abréviations, d'étirements de mots et d'écritures de type SMS. Le tableau 2 présente quelques exemples assez représentatifs de commentaires que l'on peut trouver dans le corpus.

Note	Commentaire
1	Don't let the looks fool you, this is not Sin City. It's not even remotely in the same league as Sin City. It is literally laughable... especially the dialogue...
2	Not a very impressive movie.. the acting out of Cena was pretty good, but the dialogue left MUCH to be desired..
3	t feels like the big explosive climatic conclusion of a FPS videogame. And that's a good thing.
3	It had a couple of pretty funny moments, but didn't think it was anything special.
5	This film was really funny and nearly had me in tears

Tableau 1. *Exemples de commentaires bien écrits*

Note	Commentaire
0,5	PLEASE ! ! !
0,5	EWWWWWWWWWWWWWWWWW ! ! ! ! ! ! ! ! ! !
1	sux !
1	ZZZzzzzz...is it over yet ?
2	I m@ke it r@1n
2	IT WUZ TOUCHIN..IT JUSS AINT TOUCH ME
2,5	Not 100% what I expected
3,5	Very funny. Typical and quite predictable story but pretty hilarious. I had some good laughs. :)
4	LOL funny... steve martin jean reno rule =) so as beyonce.... just the ending.... bad
5	hhhhheeeellllllllooooootom cruise is in it
5	i love this film its soo funny

Tableau 2. *Exemples de commentaires spécifiques aux réseaux sociaux*

5. Mise en œuvre de la classification d'opinions

Comme évoqué en section 3.2, la tâche de classification supervisée entraîne différents choix qui concernent les classes de prédiction, les prétraitements, la représentation et le classifieur. Les délimiteurs choisis pour la construction du vocabulaire sont l'espace et toutes les ponctuations, y compris l'apostrophe. La ponctuation fait ainsi également partie du vocabulaire car elle semble jouer un rôle très important dans les textes communautaires, ne serait-ce que par sa présence dans les « smileys ». Pour les autres choix, nous avons sélectionné les méthodes les plus régulièrement utilisées dans le domaine.

5.1. *Choix effectués*

Les solutions évaluées dans nos expériences sont les suivantes :

– prétraitements :

- minimaux : ils consistent uniquement à mettre tous les caractères en minuscules et à supprimer les éléments du vocabulaire les moins fréquents, c'est-à-dire ceux ayant un nombre d'occurrences inférieur à trois dans tout le corpus d'apprentissage,

- *stop-list* (ou antidictionnaire) : nous en avons testé plusieurs et nous nous sommes aperçus que plus elles contenaient de mots, moins les résultats de classification étaient bons. Nous présenterons donc ici les résultats concernant la liste la plus courte. Elle contient 319 mots et est disponible sur le Web[3],

3. http ://armandbrahaj.blog.al/2009/04/14/list-of-english-stop-words/

- linguistiques : ils consistent en une correction orthographique et en une lemmatisation faites à l'aide d'un analyseur syntaxique (Guimier de Neef *et al.*, 2002) ;

– représentations :

- fréquentielle,

- fréquentielle normalisée,

- TF-IDF ;

– classifieurs :

- SVM (*Support Vector Machine*) : nous avons utilisé l'outil SVM^{light}[4]. Il s'agit d'une implémentation des machines à vecteurs supports de Vapnik (Vapnik, 1995). Les algorithmes d'optimisation utilisés sont décrits dans Joachims (2002) et Joachims (1999),

- SNB (*Selective Naive Bayes*) : nous avons utilisé l'outil Khiops [5] qui est un classifieur naïf bayésien avec sélection de variables (Boullé, 2005 ; Boullé, 2006 ; Boullé, 2007). Dans une première phase de préparation des données, les variables explicatives sont évaluées individuellement au moyen d'une méthode de discrétisation optimale dans le cas numérique, et de groupement de valeurs optimal dans le cas catégoriel. Dans la phase de modélisation, un modèle de classification est construit, en moyennant efficacement un grand nombre de modèles fondés sur des sélections de variables,

- NB (*Naive Bayes*) classique : nous avons également utilisé Khiops sans le mode de sélection des variables.

Les expérimentations ont été réalisées comme suit : nous avons tout d'abord évalué chaque classifieur sur une classification binaire, avec les prétraitements minimaux et avec toutes les représentations sélectionnées. Nous avons ensuite conservé le classifieur donnant le meilleur résultat sur notre corpus de test puis nous l'avons évalué avec les autres prétraitements. Afin de comparer toutes les expérimentations, nous calculons le F_{score} qui est l'évaluation la plus couramment utilisée en classification d'opinions. Pour finir nous avons conservé le meilleur modèle et nous avons effectué les classifications sur trois classes et sur cinq classes. On notera que la segmentation des textes en trigrammes de lettres a été réalisée et comparée à la segmentation en mots, pour les différentes représentations vectorielles citées ci-dessus et les différents types de classifieurs. Les performances de classification ont été, dans tous les cas, inférieures à celles obtenues avec une segmentation des textes en mots. Nous ne présentons donc pas, dans cet article, les résultats concernant ces expérimentations.

4. Outil téléchargeable en freeware sur http ://svmlight.joachims.org/

5. Outil téléchargable en shareware sur http ://perso.rd.francetelecom.fr/boulle/

5.2. *Évaluation des classifieurs*

Les trois classifieurs sélectionnés ont tous été évalués avec les trois représentations envisagées : la fréquentielle, la fréquentielle normalisée et le TF-IDF. Les prétraitements minimaux ont été appliqués aux corpus afin de réduire le nombre de dimensions de l'espace de représentation. Les attributs composant ce vecteur sont donc tous les mots du corpus d'apprentissage mis en minuscules moins ceux apparaissant moins de trois fois dans le corpus d'apprentissage. La minusculisation permet de diviser par 2,5 la taille du vocabulaire (on passe de 151 490 à 58 474 mots). La suppression des mots très peu fréquents permet une nouvelle réduction de la taille du vocabulaire d'un facteur 3, le nombre de mots restants étant de 19 255 (nous avons vérifié que cette réduction n'avait pas d'impact sur les performances de classification). Les résultats des différentes classifications sont présentés dans le tableau 3. Sur nos données, c'est la méthode SVM associée à la représentation fréquentielle qui donne le meilleur résultat.

Représentations / Classifieurs	**Fréquentielle**	**Fréq. normalisée**	**TF-IDF**
SVM	**0,741**	0,724	0,726
SNB	0,726	0,729	0,728
NB	0,695	0,708	0,708

Tableau 3. *F_{score} obtenu pour chaque représentation avec les trois classifieurs dans le cas d'une classification binaire*

Le classifieur bayésien naïf, lui, semble plus efficace quand il est associé à la sélection de variables. Il est d'ailleurs intéressant d'étudier ces variables jugées les plus informatives. Leur nombre varie de 635 à 741 selon la représentation choisie mais les mieux classées sont les mêmes dans tous les cas. Le tableau 4 présente les quarante variables les plus informatives par ordre décroissant de degré d'information. La plus informative est le point d'exclamation, suivi par le mot « love », puis « loved » et ainsi de suite. On remarque que certaines variables semblent *a priori* ne pas contenir une grande information sémantique (« is », « was », « this », etc.) mais ces résultats montrent que, statistiquement, elles sont discriminantes pour la tâche visée. Cela est notamment dû aux styles des auteurs qui varient selon l'opinion émise dans le texte. Par exemple, les commentaires négatifs sont plus courts que les positifs et l'emploi du passé y est plus présent. L'analyse de cette liste donne des informations intéressantes sur le contenu des commentaires et on peut notamment en extraire des renseignements sur l'impact que pourraient avoir certains prétraitements couramment utilisés dans le domaine.

On remarque notamment que les caractères de ponctuation sont très représentés, le point d'exclamation étant même en tête de liste. Il semble être en effet un indicateur fort des commentaires positifs. Lorsqu'il apparaît une fois dans un commentaire, il y a 52 % de chance que ce commentaire soit positif, et lorsqu'il apparaît plus de trois fois, ce score passe à 67 %. Concernant le point, 66 % des commentaires qui en contiennent

!	best	sucked	lame
love	but	excellent	horrible
loved	movie	was	okay
not	stupid	alright	too
ok	t	waste	luv
boring	worst	awsome	nothing
awesome	.	didn	fantastic
great	crap	terrible	is
amazing	brilliant	’	this
bad	hilarious	wasn	no

Tableau 4. *Variables les plus informatives trouvées par la méthode SNB*

deux ou plus sont des commentaires négatifs. Enfin, 74 % des commentaires contenant un point d’interrogation sont des commentaires négatifs. On peut déduire de ces chiffres que la suppression de la ponctuation n’est pas un prétraitement recommandé pour ce type de textes. L’intérêt de descripteurs caractéristiques de la ponctuation (en général représentés sous la forme du nombre d’occurrences de différentes caractères de ponctuation !, ?, etc.) a été montré par (Davidiv *et al.*, 2010) pour la classification de sentiments sur des données provenant de Twitter.

Le cas de la négation est souvent discuté dans la littérature et sa prise en compte est une des plus grosses problématiques de la classification d’opinions (Pang et Lee, 2008). Les négations jouent en effet un grand rôle dans l’expression d’une opinion mais, en observant leur répartition dans les classes de notre corpus, on s’aperçoit qu’elles n’ont pas nécessairement besoin de faire l’objet d’un traitement spécifique. Par exemple, lorsque le mot « not » apparaît dans un texte, il s’agit, dans 82 % des cas, d’un commentaire négatif. À titre de comparaison, ce chiffre est de 86 % pour le mot « bad ». Les commentaires contenant les mots « no », ou « didn’t » ou encore « wasn’t » sont également des commentaires plus souvent négatifs que positifs : 77 % pour « no », 80 % pour « didn’t » et 83 % pour « wasn’t ». On peut donc en déduire que, dans ce type de textes, un traitement particulier pour la négation n’est pas forcément nécessaire car les négations ont un comportement proche des mots à polarité négative. On trouvera une discussion plus complète sur le traitement de la négation dans ces textes dans notre travail précédent (Poirier *et al.*, 2009). Différentes manières de prendre en compte la négation (s’appuyant sur des lexiques dédiés) y sont explorées et leur impact sur la classification d’opinions évalué. Mais en classification d’opinions par apprentissage supervisé, il est plus difficile d’appliquer un traitement spécifique de la négation.

On remarque également la présence de mots ayant la même racine, que ce soit dû à leur conjugaison ou à des fautes d’orthographe, volontaires ou involontaires. C’est par exemple le cas avec les mots « love », « loved », « luv » ou « awesome », « awsome » ou « great », « gr8 », ou encore « was », « is », « been », etc. Les exemples de ce type sont nombreux. L’union de ces différents mots en un seul pourrait apporter à la fois une

réduction de l'espace de représentation, et peut-être également un gain d'information pour la classification.

5.3. *Évaluation des prétraitements*

L'impact des prétraitements a été testé à l'aide des SVM et des trois représentations. Les résultats obtenus, ainsi qu'un rappel des résultats précédents avec les SVM, sont présentés dans le tableau 5.

Représentations / Prétraitements	**Fréquentielle**	**Fréq. normalisée**	**TF-IDF**
Minimaux	**0,741**	0,724	0,726
Antidictionnaire	0,720	0,700	0,703
Linguistiques	0,738	0,722	0,724

Tableau 5. *F_{score} obtenus pour tous les prétraitements et toutes les représentations avec le classifieur SVM*

5.3.1. *Antidictionnaire (*stop-list*)*

Le meilleur résultat obtenu avec un antidictionnaire est une nouvelle fois dû à la représentation fréquentielle qui a un F_{score} de 0,720 contre 0,700 pour la représentation fréquentielle normalisée et 0,703 pour la représentation TF-IDF. Ce meilleur F_{score} est toutefois inférieur à celui obtenu sans la suppression des mots de l'antidictionnaire. On peut supposer que cette baisse de performance est due à la suppression de certains mots informatifs. En effet, cent mots présents dans l'antidictionnaire font partie des variables informatives extraites lors de l'évaluation des classifieurs. Il s'agit de mots comme « but », « was », « is », « so », « at », « to », « for », qui n'ont pas de sens sémantique très fort mais dont la présence semble apporter une information non négligeable pour la détermination de l'opinion. Nous n'avons pas poussé plus avant l'analyse des mots de la *stop-list* responsables de la dégradation des performances.

5.3.2. *Prétraitements linguistiques*

Rappelons tout d'abord que les prétraitements linguistiques appliqués sont une correction orthographique suivie d'une lemmatisation. Ces deux types de traitements n'ont pas été évalués séparément. Ils permettent une réduction du nombre de dimensions d'environ 25 %. Le nombre d'éléments du vocabulaire est alors de 14 224, ce qui a un impact sur les temps de calcul qui n'est pas négligeable. De plus, les F_{score} atteints, bien qu'étant un peu moins bons, sont très proches de ceux des premières expérimentations.

La très légère baisse de performance pourrait être expliquée par le fait que la lemmatisation annihile le degré d'information de certains mots. Par exemple, 66 % des commentaires contenant le mot« was » sont négatifs alors que c'est le cas pour seulement 51 % des commentaires contenant « is ». La fusion de ces deux mots en un

seul entraîne donc un nivellement de la quantité d'information contenue dans chacun d'entre eux.

5.4. *Impact du nombre de classes*

Les commentaires du corpus étant originellement répartis sur dix classes, nous avons jugé qu'il était nécessaire de réduire ce nombre en regroupant certaines d'entre elles. Nous avons donc testé plusieurs regroupements aboutissant à différents nombres de classes finales. Nous avons évalué des classifications sur deux, trois et cinq classes à l'aide du modèle qui a obtenu les meilleurs résultats jusqu'à présent, à savoir le classifieur SVM avec des prétraitements minimaux et une représentation fréquentielle. La comparaison des résultats de chaque classification est difficile à faire. En effet, au-delà de deux classes le calcul du F_{score} n'est plus possible et les erreurs faites sur cinq classes n'ont pas nécessairement le même poids que celles faites sur deux ou trois classes. Nous avons donc décidé de vérifier la qualité de ces classifications à l'aide de la tâche de recommandation (partie 6.2). Les tableaux 6 et 7 sont les matrices de confusion obtenues pour les classifications sur trois et cinq classes.

		Vraies classes		
		NEG	**NEUTRE**	**POS**
Classes prédites	NEG	**71,7**	34,5	11,0
	NEUTRE	15,1	**34,5**	17,4
	POS	13,2	31,0	**71,6**

Tableau 6. *Résultats de la classification d'opinions sur trois classes (en pourcentage suivant les vraies classes)*

		Vraies classes				
		1	**2**	**3**	**4**	**5**
Classes prédites	1	**60,3**	29,2	14,9	7,3	5,4
	2	16,8	**30,7**	22,4	7,2	3,5
	3	7,1	15,2	**23,2**	14,6	6,8
	4	6,0	10,7	18,8	**24,9**	15,0
	5	9,8	14,2	20,7	46,0	**69,3**

Tableau 7. *Résultats de la classification d'opinions sur cinq classes (en pourcentage suivant les vraies classes)*

5.5. *Discussion*

Le meilleur résultat obtenu, concernant les classifications binaires, est donc dû au classifieur SVM associé aux prétraitements minimaux (c'est-à-dire sans prétraite-

ments linguistiques ni utilisation d'un antidictionnaire) et à une représentation fréquentielle. Nous avons cherché à observer les commentaires mal classés par cette méthode. Pour la grande majorité d'entre eux, il n'est pas difficile de comprendre pourquoi le modèle s'est trompé. Nous citons ici les cas d'erreurs les plus fréquents, sans prétendre à l'exhaustivité. Dans certains cas, une ou plusieurs fautes d'orthographe empêchent le classifieur de repérer un mot important, comme dans l'exemple suivant qui a été classé comme commentaire positif : « it wasent a very good movie ». D'autres textes peuvent être mal classés car ils ne contiennent que des informations subjectives comme le commentaire : « so sad » qui a été classé comme négatif alors que l'auteur a attribué une note de 4,5 au film ou encore « it was very crazy ! » qui a été classé positif alors que l'auteur n'a pas du tout apprécié le film. Enfin nous pouvons également citer les nombreux cas où l'auteur du commentaire est en cause car il s'est trompé en attribuant la note, ou plus précisément, il n'a pas modifié la note mise par défaut par le site (0,5). Il n'y a donc alors aucune cohérence entre la note et le commentaire. De nouveaux prétraitements pourraient être évalués afin de réduire le taux de mauvaises classifications comme par exemple une radicalisation des mots (algorithme de Porter) ou encore en construisant un lexique de smileys afin de conserver ces suites de caractères dans les textes.

6. Application à la recommandation personnalisée

6.1. *Nouvelles données utilisées*

L'évaluation de la classification d'opinions à l'aide de la recommandation automatique nécessite des données nouvelles, mais qui doivent porter sur des films pour lesquels nous disposons aussi de commentaires. Nous avons pour cela utilisé les données du challenge Netflix (Bennett et Lanning, 2007) lancé en 2006 dont l'objectif était d'améliorer de 10 % les résultats du moteur de recommandation initial du site Netflix. Rappelons que ce site collecte les avis (sous la forme de notes) de loueurs de DVD. Ces nouvelles données, que nous nommerons « Données Netflix », contiennent 5 200 000 notes postées par 25 000 loueurs différents de DVD et portent sur 17 770 films. Elles ne constituent qu'une partie des données proposées par le challenge. Nous avons séparé ces notes disponibles en deux ensembles : 4 700 000 servent de données d'apprentissage pour la recommandation, les 500 000 notes restantes servent de test pour évaluer la qualité des notes prédites. La construction de ces deux ensembles s'est faite utilisateur par utilisateur : 90 % des notes de chaque utilisateur, sélectionnées aléatoirement, ont été placées dans les données d'apprentissage, les 10 % restantes servant pour la validation.

Pour comparer les vraies notes de l'ensemble de test avec celles prédites grâce à des informations collectées ailleurs, nous avons sélectionné 3 300 000 nouveaux commentaires provenant, comme précédemment, du site communautaire Flixster, portant sur 10 220 films communs avec les Données Netflix. Ces commentaires ne faisaient pas partie de l'ensemble d'apprentissage des classifieurs mais ils possèdent bien sûr

les mêmes caractéristiques que ceux présentés dans les sections précédentes, c'est-à-dire que ce sont des textes très courts et relativement mal écrits.

Nos expérimentations vont porter sur la façon de calculer la similarité entre films, qui sert dans la formule [3] à prédire la note d'un utilisateur donné sur un film donné. Dans le cas du filtrage collaboratif, nous rappelons que cette similarité est donnée par la formule [1]. Nous allons en particulier comparer trois méthodes différentes :

– en utilisant les vraies notes des visiteurs du site Netflix (des loueurs de DVD) ;

– en utilisant les vraies notes des utilisateurs du site Flixster (des amateurs de cinéma mais qui n'ont a priori pas loué de DVD *via* Netflix) ;

– en utilisant les notes prédites à partir des commentaires de Flixster par le classifieur appris automatiquement.

6.2. *Résultats des évaluations en recommandation*

Nous avons tout d'abord étalonné notre système de recommandation sur les vraies notes du challenge Netflix. Nous avons donc affaire dans ce cas à de véritables données, du type de celles utilisées dans l'industrie. La RMSE obtenue par filtrage collaboratif avec cette méthode est de 0,862, ce qui est très satisfaisant. Le tableau 8 récapitule les résultats obtenus lors de cet étalonnage.

Un de nos objectifs est de prouver que l'apport de données textuelles non structurées provenant de sites communautaires peut être une alternative avantageuse au filtrage thématique. Nous avons donc aussi mesuré les performances de l'approche thématique sur les films du challenge Netflix. Dans ce cas, la similarité entre films est évaluée par la formule [2], dans laquelle les ensembles S_i et S_j sont constitués de descripteurs qui représentent les films i et j. Pour définir ces descripteurs, nous avons utilisé les données du site de référence sur le cinéma : *IMDB*[6]. Ce site fournit des informations très complètes pour chaque film : liste de ses acteurs, réalisateur(s), producteur(s), genre(s), nationalité(s), année de production, scénariste(s), compagnie(s), langue(s) et des dizaines de termes clés saisis à la main et censés être représentatifs de son intrigue (par exemple « monolith », « computer », « computer chess », « moon », « evolution », « year 2001 », « alien », et plus d'une centaine d'autres pour le film *2001, l'Odyssée de l'espace*). Il s'agit donc de données très riches, qui ne pourraient pas nécessairement être disponibles pour des articles autres que des films. Les valeurs de tous ces champs constituent l'ensemble des descripteurs qui représentent chaque film. La RMSE obtenue par filtrage thématique avec cette méthode est de 0,968.

Les données utilisées pour ces expérimentations figurant parmi les plus riches connues, nous considérons ces deux valeurs comme les meilleurs résultats possibles que notre système de recommandation peut atteindre sur l'ensemble de test des Données Netflix, respectivement par filtrage collaboratif et par filtrage thématique. Elles

6. www.imdb.com

Filtrage collaboratif (Données Netflix)	Filtrage thématique (Données IMDB)
0,862	0,968

Tableau 8. *Évaluation du système de recommandation sur deux ensembles de données connus*

illustrent parfaitement que le filtrage thématique est généralement moins performant que le filtrage collaboratif, quand les deux sont applicables.

6.3. *Recommandations utilisant les données du site communautaire*

Nous nous mettons désormais dans la position d'un système de recommandation qui utilise les données extérieures provenant du site Flixster pour évaluer la similarité entre ses articles, en utilisant le filtrage collaboratif.

Nous avons tout d'abord évalué notre système de recommandation avec les notes fournies explicitement par les auteurs des commentaires de Flixster. Ces notes évaluent des films. Elles n'ont donc pas exactement le même sens que celles fournies par les loueurs de DVD de Netflix qui, eux, peuvent prendre en compte dans leurs évaluations d'autres propriétés de l'objet DVD (qualité de la copie, des bonus, etc.). La RMSE atteinte dans ce cas est de 0,897. Cela conforte l'idée que l'exploitation de données externes au site initial mais traitant du même domaine est une démarche qui peut s'avérer efficace – plus efficace en tout cas que l'utilisation de descripteurs et d'un filtrage thématique.

Nous avons ensuite utilisé les notes prédites par notre meilleur classifieur, celui qui exploite un SVM sur une classification binaire, avec des prétraitements minimaux et une représentation fréquentielle. Le résultat en RMSE est alors 0,898. Les notes prédites sont donc quasiment équivalentes en termes de recommandation aux notes fournies par les auteurs des textes. Enfin, nous avons également entraîné le système en remplaçant les notes de Flixster connues ou prédites par des notes aléatoires. La RMSE obtenue est 0,989. Ce résultat n'est pas mauvais, ce qui peut paraître surprenant. Cela s'explique sans doute par le fait que les relations entre les utilisateurs et les films restent valables bien que la note soit modifiée. Ce résultat montre que ces liens entre utilisateurs et films semblent contenir une information non négligeable. Savoir qui a évalué (et donc vu) tel film, sans connaître l'opinion, possède un intérêt pour calculer des similarités entre films et en déduire une recommandation pertinente. Le tableau 9 récapitule ces trois résultats.

Pour finir, nous avons également comparé le résultat en recommandation obtenu en calculant les similarités avec les notes issues de la classification d'opinions binaire et ceux obtenus avec des classifications sur trois et cinq classes, afin de déterminer si

Filtrage collaboratif		
Vraies notes	Notes prédites	Notes aléatoires
0,897	0,898	0,989

Tableau 9. *Résultats obtenus avec les données communautaires*

une classification plus fine pouvait avoir une influence sur la qualité des recommandations. La RMSE obtenue à partir des résultats de la classification sur trois classes est de 0,907 et de 0,913 pour la classification sur cinq classes. Il semble donc que le gain en précision obtenu avec les classifications multi-classes ne compense pas l'augmentation naturelle du nombre d'erreurs. Le tableau 10 récapitule ces derniers résultats.

Filtrage collaboratif		
Avec la classification d'opinions sur 2 classes	Avec la classification d'opinions sur 3 classes	Avec la classification d'opinions sur 5 classes
0,898	0,907	0,913

Tableau 10. *Résultats obtenus avec les données communautaires et les différentes classifications d'opinions*

7. Conclusion

Dans cet article nous nous sommes intéressés à l'exploitation de textes produits par des internautes dans le contexte de la recommandation de contenu. Les textes communautaires véhiculent l'opinion de leurs auteurs et nous avons cherché à transformer ces textes non structurés en données d'usage pour contribuer à alimenter un moteur de recommandation. Nous avons proposé une chaîne de traitements qui associe une étape de classification d'opinions, pour transformer les commentaires textuels des internautes en notes, et une étape de recommandation par filtrage collaboratif pour exploiter ces données d'usage. Les textes d'opinions proviennent de blogs et n'ont aucun lien avec le service recevant les recommandations, en dehors de leur sujet (en l'occurrence des films). Il s'agissait donc de vérifier que l'apport de données extérieures pouvait avoir un intérêt pour le domaine de la recommandation de contenus, notamment dans le but de pallier le problème du *démarrage à froid*. Nos expériences semblent montrer que c'est le cas.

Pour la partie classification d'opinions, nous avons comparé différentes stratégies pour apprendre à associer une note à un texte à partir d'exemples : un classifieur bayésien naïf, un classifieur bayésien naïf avec sélection de variables et une machine à vecteurs supports. Nous avons également évalué différents prétraitements et représentations. Les meilleures performances de classification ont été obtenues avec

un SVM et ce, quelle que soit la représentation des textes choisie. On retrouve ici les résultats de Joachims (1999) et de nombreux autres auteurs en classification d'opinions, les SVM étant très bien adaptés aux données décrites en grande dimension et aux données creuses. Le classifieur bayesien naïf avec sélection de variables est légèrement moins performant que le SVM en classification mais il nous a apporté des informations intéressantes sur le caractère informatif du vocabulaire. Il a ainsi notamment fait émerger des mots importants pour la classification d'opinions que l'on élimine avec les méthodes de prétraitements classiques comme l'utilisation d'antidictionnaires ou la suppression de la ponctuation. Il s'avère que sur les données utilisées, les meilleurs résultats de classification ont été obtenus avec les données les moins prétraitées (minusculisation, suppression des mots peu fréquents) et que l'utilisation d'un antidictionnaire ou de prétraitements linguistiques (correction orthographique et lemmatisation) n'ont pas permis d'améliorer les résultats. La complexification de la représentation n'a également pas apporté d'amélioration, la représentation fréquentielle ayant obtenu des meilleurs scores que les représentations fréquentielles normalisées et TF-IDF.

Concernant la partie recommandation, nous avons montré que l'apport de données extérieures pour le calcul des matrices de similarités entre articles dans le moteur de recommandation est tout à fait pertinent. Nous avons confronté nos résultats à ceux obtenus sur le corpus de référence Netflix ainsi qu'à une méthode de filtrage thématique portant sur des descripteurs très riches provenant de l'Internet Movie Data Base. Nos résultats avec des données extérieures sont moins bons qu'avec les notes initiales de Netflix mais bien meilleurs qu'avec les informations provenant d'IMDB, ce qui montre qu'il peut être plus pertinent d'alimenter un moteur en données d'usage extraites de sites communautaires que de faire de la recommandation thématique.

Les travaux concernant la classification d'opinions restent toutefois assez préliminaires (nous cherchons uniquement à prédire la note) mais les derniers résultats montrent que l'information que nous extrayons est déjà très précieuse pour un domaine d'application qui devient de plus en plus important, la recommandation automatique. De plus, toutes les expérimentations ont été menées en vue d'une application industrielle. Nous avons également pu identifier certaines caractéristiques de l'expression des opinions sur les sites communautaires : la négation est utilisée en grande partie pour exprimer une opinion négative, les auteurs sont plus prolixes lorsqu'ils émettent un jugement positif, la conjugaison des verbes a une importance (le passé est plus présent dans les commentaires négatifs alors que le présent est plus utilisé pour les textes positifs), la ponctuation joue également un grand rôle, etc.

Dans nos travaux futurs, nous voudrions être capables de mieux évaluer les quantités respectives de notes initiales et de données externes nécessaires pour établir de bonnes recommandations. Dans le cas où c'est un système de classification automatique qui est utilisé sur des textes, nous souhaiterions également pouvoir mesurer plus

précisément la corrélation entre la qualité d'une classification et celle de la recommandation à laquelle elle contribue. Nous envisageons aussi d'étudier comment enrichir un moteur de recommandation en prenant en compte de nouveaux types d'informations. La connectivité du réseau des « amis » déclarés dans les sites communautaires pourrait ainsi constituer une source de données nouvelles et précieuses à intégrer dans les modèles.

8. Bibliographie

Bennett J., Lanning S., « The Netflix Price », ACM, San Jose, California, USA, 2007.

Boullé M., « A Bayes optimal approach for partitioning the values of categorical attributes », *Journal of Machine Learning Research*, vol. 6, p. 1431-1452, 2005.

Boullé M., « MODL : a Bayes optimal discretization method for continuous attributes », *Machine Learning*, vol. 65, n° 1, p. 131-165, 2006.

Boullé M., « Compression-Based Averaging of Selective Naive Bayes Classifiers », *Journal of Machine Learning Research*, vol. 8, p. 1659-1685, 2007.

Burke R., « Hybrid Web Recommender Systems », *in* P. Brusilovsky, A. Kobsa, W. Nejdl (eds), *The Adaptive Web*, vol. 4321 of *Lecture Notes in Computer Science*, Springer, chapter 12, p. 377-408, 2007.

Candillier L., Meyer F., Fessant F., « Designing Specific Weighted Similarity Measures to Improve Collaborative Filtering Systems », *in* P. Perner (ed.), *8th Industrial Conference on Data Mining (ICDM'2008)*, LNCS, Springer Verlag, Leipzig, Germany, july, 2008.

Cane, Stephen, Chung F. L., « Integrating Collaborative Filtering and Sentiment Analysis : A Rating Inference Approach », *Proceedings of The ECAI 2006 Workshop on Recommender Systems*, Riva del Garda, I, p. 62-66, 2006.

Crestan E., Gigandet S., Vinot R., « Approches naïves à l'analyse d'opinion », *Actes de l'atelier de clôture du 3e DÉfi Fouille de Textes*, AFIA, Grenoble, France, p. 45-56, 2007.

Davidiv D., Tsur O., Rappoport A., « Enhanced Sentiment Learning Using Twitter Hashtags and Smileys », *23rd International Conference on Computational Linguistics (COLING'2010)*, Beijing, China, august, 2010.

Dziczkowski G., Wegrzyn-Wolska K., « RRSS - Rating Reviews Support System Purpose Built for Movies Recommendation », *AWIC*, p. 87-93, 2007.

Généreux M., Santini M., « Défi : Classification de textes français subjectifs », *Actes de l'atelier de clôture du 3e DÉfi Fouille de Textes*, AFIA, Grenoble, France, p. 83-93, 2007.

Grouin C., Berthelin J.-B., El Ayari S., Heitz T., Hurault-Plantet M., Jardino M., Khalis Z., Lastes M., « Présentation de DEFT'07 », *Actes de l'atelier de clôture du 3e DÉfi Fouille de Textes*, AFIA, Grenoble, France, p. 1-8, 2007.

Guimier de Neef E., Boualem M., Chardenon C., Filoche P., Vinesse J., « Natural language processing software tools and linguistic data developped by France Télécom RD », 2002.

Joachims T., « Making large-scale support vector machine learning practical », *Advances in kernel methods : support vector learning*p. 169-184, 1999.

Joachims T., *Learning to Classify Text Using Support Vector Machines : Methods, Theory and Algorithms*, Kluwer Academic Publishers, Norwell, MA, USA, 2002.

Kaiser C., Krockel J., Bodendorf F., « Swarm Intelligence for Analyzing Opinions in Online Communities », *Hawaii International Conference on System Sciences*, vol. 0, p. 1-9, 2010.

Koren Y., « Factors in the neighbors : Scalable and accurate collaborative filtering », *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 4, n° 1, p. 1-24, 2010.

Macdonald C., Ounis I., Soboroff I., « Overview of the TREC-2007 Blog Track », *Proceedings of The sixteenth Text REtrieval Conference (TREC 2007)*, NIST, p. 17-30, 2007.

Maurel S., Curtoni P., Dini L., « Classification d'opinions par méthodes symbolique, statistique et hybride », *Actes de l'atelier de clôture du 3e DÉfi Fouille de Textes*, AFIA, Grenoble, France, p. 109-116, 2007.

Meyer F., « Apport des données thématiques dans les systèmes de recommandation : hybridation et démarrage à froid », *Extraction et gestion des connaissances (EGC'2011)*, Brest, France, 2011.

Nageswara Rao K., Talwar V., « Application domain and functional classification of recommender systems a survey », *Desidoc journal of library and information technology*, vol. 28, ACM Press, p. 17-36, 2008.

Nigam K., Hurst M., « Towards a Robust Metric of Polarity », *Computing Attitude and Affect in Text : Theory and Applications*, Springer, Dordrecht, The Netherlands, 2006.

Ounis I., de Rijke M., Macdonald C., Mishne G., Soboroff I., « Overview of the TREC-2006 Blog Track », *Proceedings of The Fifteenth Text REtrieval Conference (TREC 2006)*, NIST, p. 17-31, 2006.

Ounis I., Macdonald C., Soboroff I., « Overview of the TREC-2008 Blog Track », *Proceedings of The seventeenth Text REtrieval Conference (TREC 2008)*, NIST, 2008.

Panckhurst R., « Le discours électronique médié : bilan et perspectives. », 2006.

Pang B., Lee L., « A sentimental education : sentiment analysis using subjectivity summarization based on minimum cuts », *ACL '04 : Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, Morristown, NJ, USA, p. 271, 2004.

Pang B., Lee L., « Opinion Mining and Sentiment Analysis », *Found. Trends Inf. Retr.*, vol. 2, n° 1-2, p. 1-135, 2008.

Pang B., Lee L., Vaithyanathan S., « Thumbs up ? : sentiment classification using machine learning techniques », *EMNLP '02 : Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, Association for Computational Linguistics, Morristown, NJ, USA, p. 79-86, 2002.

Pazzani M., Billsus D., « Content-Based Recommendation Systems », p. 325-341, 2007.

Pazzani M. J., « A Framework for Collaborative, Content-Based and Demographic Filtering », *Artif. Intell. Rev.*, vol. 13, n° 5-6, p. 393-408, 1999.

Plantié M., Roche M., Dray G., Poncelet P., « Is a Voting Approach Accurate for Opinion Mining ? », *DaWaK '08 : Proceedings of the 10th international conference on Data Warehousing and Knowledge Discovery*, Springer-Verlag, Berlin, Heidelberg, p. 413-422, 2008.

Poirier D., Fessant F., Bothorel C., Guimier de Neef E., Boullé M., « Approches Statistique et Linguistique Pour la Classification de Textes d'Opinion Portant sur les Films », *RNTI*p. 147-169, 2009.

Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J., « GroupLens : an open architecture for collaborative filtering of netnews », *CSCW '94 : Proceedings of the 1994 ACM confe-*

rence on Computer supported cooperative work, ACM, New York, NY, USA, p. 175-186, 1994.

Sarwar B., Karypis G., Konstan J., Reidl J., « Item-based collaborative filtering recommendation algorithms », *WWW '01 : Proceedings of the 10th international conference on World Wide Web*, ACM, New York, NY, USA, p. 285-295, 2001.

Schein A. I., Popescul A., Ungar L. H., Pennock D. M., « Methods and metrics for cold-start recommendations », *SIGIR '02 : Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, p. 253-260, 2002.

Seki Y., Kirk Evans D., Ku L.-W., Chen H.-H., Kando N., Lin C.-Y., « Overview of Opinion Analysis Pilot Task at NTCIR-6 », *Proceedings of NTCIR-6 Workshop Meeting*, Tokyo, Japan, p. 265-278, 2007.

Seki Y., Kirk Evans D., Ku L.-W., Sun L., Chen H.-H., Kando N., « Overview of Multilingual Opinion Analysis Task at NTCIR-7 », *Proceedings of NTCIR-7 Workshop Meeting*, Tokyo, Japan, p. 185-203, 2008.

Trinh A.-P., « Classification de texte et estimation probabiliste par Machine à Vecteurs de Support », *Actes de l'atelier de clôture du 3e DÉfi Fouille de Textes*, AFIA, Grenoble, France, p. 69-82, 2007.

Vapnik V. N., *The nature of statistical learning theory*, Springer-Verlag New York, Inc., New York, NY, USA, 1995.

Wilson T., Wiebe J., Hwa R., « Just how mad are you ? finding strong and weak opinion clauses », *AAAI'04 : Proceedings of the 19th national conference on Artifical intelligence*, AAAI Press, p. 761-767, 2004.

Yu H., Hatzivassiloglou V., « Towards answering opinion questions : Separating facts from opinions and identifying the polarity of opinion sentences », *In Proceedings of EMNLP-03*, p. 129-136, 2003.