Evaluation de la pertinence dans un système de recommandation sémantique de nouvelles économiques

Résumé. De nos jours dans les secteurs commerciaux et financiers la veille électronique d'articles économiques est cruciale. Maintenir une veille efficace implique de cibler les articles à consulter, car la charge d'information est importante. Pour répondre à cette problématique, nous proposons un système novateur de recommandation d'articles, car il s'appuie sur l'intégration d'une description sémantique des items et des profils basés sur une modélisation ontologique des connaissances. Nous appuyons notre système de recommandation sur un modèle vectoriel intrinsèquement efficace que nous avons perfectionné pour pallier la confusion native de ces modèles entre les notions de similarité et de pertinence qui ne permet pas de prendre en compte les effets de la différence dans la précision des descriptions sémantiques des profils et articles, sur la perception de la pertinence pour l'utilisateur. Nous présentons donc dans cet article une nouvelle méthode d'évaluation de la pertinence adaptée au modèle vectoriel.

1 Introduction

Afin de rester en phase avec les tendances actuelles du marché, le processus de prise de décision dans le domaine économique nécessite la centralisation et l'apport de grandes quantités d'informations. Pour cela, les hommes d'affaires, les entrepreneurs et les vendeurs doivent parfaitement connaître leur environnement. Cela signifie qu'il faut maintenir une veille économique constante facilitant l'identification des perspectives d'affaires, permettant de décrocher de nouveaux contrats. Cette veille incontournable est cependant complexe à assurer, c'est pourquoi nous proposons un outil efficace de recommandation d'articles régionaux d'actualités économiques utilisant des représentations sémantiques communes des connaissances afin de décrire les besoins de l'utilisateur et les informations permettant d'y répondre.

Notre approche s'établit sur l'adéquation de la recommandation aux besoins des utilisateurs, ainsi avons-nous mené une enquête auprès des clients (lecteurs) afin de définir les critères qui pourraient permettre la personnalisation du contenu de la revue. Les résultats de l'enquête ainsi que la connaissance des experts du domaine ont permis de mettre en avant les trois critères principaux suivants : les *Thèmes* (principaux évènements économiques traités dans l'article), les *Secteurs économiques* dont traitent l'article et les *Localisations*. Cette étude nous permet de baser de manière adéquate notre système de recommandation sur le contenu des articles

économiques. Pour cela des méthodes de Traitement Automatisé du Langage Naturel, ainsi qu'un travail manuel d'indexation sont mis en œuvre afin d'affecter au mieux les trois vocabulaires contrôlés décrivant ces critères principaux d'indexation. Ces critères sont communs à la description des profils des lecteurs ainsi que des articles, nous permettant de rapprocher ces deux indexations selon des mesures de *similarité* et de *pertinence*.

Ces deux indicateurs de similarité et pertinence sont au cœur de cet article, car comme nous le montrons dans l'état de l'art en section 2, un amalgame est fait entre ces deux notions. Nous les distinguons donc et les définissons en section 4. Par ailleurs, le second point important de notre démarche est l'*expansion* des profils des lecteurs et des articles par notre base de connaissances ontologique que nous présentons en section 3. Enfin, nous évaluons ces deux axes d'expansion ontologique de nos vecteurs d'une part et d'intégration d'une mesure de pertinence appelée (*Relevancy measure*) d'autre part avant de conclure et de présenter les travaux qui développent cette approche.

2 Etat de l'art

Deux principaux systèmes de recommandation sont distingués : les systèmes dits de *filtrage collaboratif* et les systèmes *basés sur le contenu*. La synthèse de (Rao et Talwar, 2008) propose une comparaison générale des principaux avantages et inconvénients de ces deux systèmes de recommandation. Notre besoin de recommandation rapide d'articles économiques, chaque jour nouveaux, exclut la solution de systèmes à base de filtrage collaboratif. En effet, ces systèmes nécessitent qu'un nombre suffisant d'utilisateurs aient lu chacun des articles avant d'être capables de les recommander. Ces systèmes peinent donc à recommander de nouveaux items. En outre, nous devons être en mesure de recommander des profils d'utilisateurs très particuliers, car certains besoins clients peuvent être uniques. Cela n'est pas possible avec un système de filtrage collaboratif car la recommandation des items se fait en fonction des appétences de personnes ayant un profil similaire. Nous nous sommes donc tournés vers les systèmes basés sur le contenu, plus adaptés à nos besoins. Les avantages des systèmes de recommandation basés sur le contenu dans le cadre de la recommandation d'articles d'actualité sont également développés dans (Liu et al., 2010).

Il existe de nombreux systèmes de recommandation qui fonctionnent sans utilisation de connaissances supplémentaires (Liu et al., 2010), (Billsus et Pazzani, 1999) et (Resnick et al., 1994) mais (IJntema et al., 2010) ont montré que l'utilisation de connaissances extérieures peut améliorer la recommandation. Ils parlent alors de systèmes de recommandation basés sur la *sémantique* pour qualifier des systèmes basés sur le contenu utilisant des connaissances externes. Les systèmes de recommandation basés sur la sémantique utilisent des connaissances lexicales (Getahun et al., 2009), comme WordNet (Fellbaum, 1998), ou alors des connaissances de domaine (Middleton et al., 2004), voire une combinaison des deux (IJntema et al., 2010) dans l'objectif d'améliorer les performances du système. Les ontologies utilisées par ces systèmes existent déjà ou sont créées à la main, et maintenues. Contrairement à ces systèmes, notre base de connaissances est utilisée comme index, les articles et les profils y sont définis sémantiquement.

Le modèle vectoriel (Salton, 1971) consiste en la représentation des items à recommander (dans notre cas des articles), ainsi que parfois du besoin (requêtes, profils) sous la forme de vecteurs. Cette présentation permet l'utilisation de différentes métriques afin de les comparer.

Dans cet article nous utiliserons les similarités cosinus et Jaccard ainsi que la distance Euclidienne. Beaucoup de systèmes de recommandation basés sur le contenu l'utilisent lors de la réalisation des tâches de comparaison que ce soit entre items, ou entre item et profil((IJntema et al., 2010) (Middleton et al., 2004) (Getahun et al., 2009) (Billsus et Pazzani, 1999) (Ahn et al., 2007)). Cette méthode d'algèbre linaire présente deux principaux avantages : non seulement elle fournit un résultat non binaire, permettant donc d'ordonner les résultats des systèmes de recommandation, mais elle permet également des calculs rapides et une bonne résistance à la montée en charge.

Par ailleurs, des méthodes de recherche d'information peuvent être utilisées afin de prendre en compte cette connaissance tout en utilisant une modélisation vectorielle. L'approche proposée par (Voorhees, 1994) utilise la base de connaissance lexicale WordNet (Fellbaum, 1998) afin d'améliorer la gestion de l'hétérogénéité du langage naturel et donc d'améliorer la compréhension des besoins de l'utilisateur. L'idée est d'ajouter de l'information aux requêtes des utilisateurs (*expansion de requêtes*). Cette méthode permet d'augmenter le rappel dans l'objectif d'améliorer les performances globales du système.

Nous avons transposé cette méthode aux systèmes de recommandation. (Middleton et al., 2004) utilise cette méthode sans la nommer. (IJntema et al., 2010) y a également recours, mais contrairement à Middleton, il utilise d'autres relations ontologiques que " is_a " pour étendre le profil de l'utilisateur.

Nous avons constaté que les notions de similarité et de pertinence sont généralement confondues dans les systèmes utilisant une modélisation vectorielle. Définir la pertinence comme une similarité ne permet pas de prendre en compte les différents degrés de spécificité dans la description du besoin des utilisateurs. Cette description est pourtant rendue possible par l'utilisation de connaissances externes au système. Nous proposons donc une mesure d'évaluation de la pertinence, *Relevancy measure*, utilisant les notions de similarité, mais prenant en compte la perception de la pertinence par l'utilisateur.

3 Vectorisation

Notre objectif concerne la recommandation d'articles économiques produits par une société auprès de ses lecteurs abonnés. Afin de répondre à cette problématique, nous proposons une approche en deux phases. La première consiste à créer une représentation des besoins des utilisateurs et du contenu des articles. La seconde étape s'attache à la recommandation de ces articles auprès de ces utilisateurs. Dans cette partie nous abordons la première étape de notre démarche : la création des vecteurs utilisateurs et articles, puis leur expansion sémantique par une ontologie.

3.1 Génération des vecteurs

Afin de définir le contenu des articles ainsi que les profils des lecteurs, un système d'indexation a été développé. Il permet l'utilisation d'un référentiel commun d'indexation pour les articles et profils, facilitant leur comparaison et donc la recommandation.

3.1.1 Indexation d'articles

L'indexation est l'étape où le lien est fait dans la base de connaissances entre l'article et les connaissances qui lui sont associées, cela permet la création d'une représentation compréhensible par la machine du contenu de chaque article. Les articles sont indexés de façon semi-automatique, après leur rédaction de façon supervisée par leurs auteurs. La plate-forme GATE (Cunningham, 2002) est utilisée pour l'analyse des articles et l'extraction des informations. Les informations non structurées contenues dans les articles sont analysées. Deux types d'information peuvent être distingués : les informations *explicites* (par exemple les lieux, les personnes, les organisations, etc) et les informations *implicites* (par exemple, le thème de chaque article ou les secteurs économiques concernés). Les résultats de cette analyse sont ensuite vérifiés, corrigés et validés manuellement par leur rédacteur. Les vecteurs de description de chaque article utilisés lors de la comparaison avec les profils contiennent les instances des critères avec lesquels ils sont en relation directe dans la base de connaissances.

3.1.2 Indexation des profils des lecteurs

L'indexation des profils de lecteurs s'opère lors de leur inscription. Les vendeurs de la société en charge de la compréhension des besoins de chaque client, proposent une période d'essai gratuite qui permet à un expert de créer un premier profil à chacun des clients. Les profils sont indexés en fonction des mêmes critères que les articles. Ils sont décrits dans la même base de connaissances que les articles afin de faciliter leur rapprochement. Une interface permet a l'expert la création manuelle des profils, cela permet d'éviter le problème du démarrage à froid, commun aux systèmes de recommandation basés sur le contenu (Rao et Talwar, 2008). De façon similaire aux articles, les vecteurs de description de chaque profil utilisés lors de la comparaison avec les articles contiennent les instances des critères avec lesquels ils sont en relation directe dans la base de connaissances.

3.2 Expansion de vecteurs

Définition d'Ontologie. Nous caractérisons une ontologie suivant la définition de (Ehrig et al., 2004) :

$$O = (C, T, \leq_C, \leq_T, R, A, \sigma_A, \sigma_R, \leq_R, \leq_A)$$

avec C, T, R, A des ensembles disjoints de concepts, types de données, relations et attributs, $\leq_C, \leq_T, \leq_R, \leq_A$ les hiérarchies de classes, type de données, relation et attributs, et σ_A, σ_R des fonctions qui produisent une signature pour chaque $\sigma_A : A \rightarrow C \times T$ attribut et $\sigma_R : R \rightarrow C \times C$ relation.

Définition de Base de connaissances. Nous définissons une base de connaissances par :

$$K = (C, T, R, A, I, V, i_C, i_T, i_R, i_A)$$

avec C, T, R, A, I, V des ensembles disjoints de concepts, type de données, relations, attributs, instance et valeurs de données. i_C la fonction d'instanciation des classes $i_C : C \rightarrow 2^I$. i_T est la

fonction d'instanciation des types de données $i_T : T \rightarrow 2^V$. i_R est la fonction d'instanciation des relations $i_R : R \rightarrow 2^{I \times I}$. i_A est la fonction d'instanciation des attributs $i_A : A \rightarrow 2^{I \times V}$.

Conception de la base de connaissances. Une fois les vecteurs d'articles et de profils constitués, nous les enrichissons grâce à notre base de connaissances composée de plusieurs ontologies selon les principes des ontologies modulaires (d'Aquin et al., 2009) :

- une ontologie de *domaine* instanciant les secteurs d'activité et événements économiques,
- une ontologie *générale* ayant pour charge dans un premier temps des paramètres de géolocalisation et la temporalité,
- une ontologie du *système* instanciant les profils et les articles.

L'ontologie du système est peuplée par les instances de profils et d'articles ainsi que les relations « isAbout » et «isInterestedIn » qui permettent d'associer respectivement les profils et les articles aux instances des critères définis dans la base de connaissance. L'objectif étant la création d'une représentation sémantique, compréhensible par la machine, des besoins et intérêts de chaque utilisateur ainsi que du contenu de chaque article.

Expansion de vecteurs. Notons, que lors de la vectorisation, nous ne tenons pas compte des connaissances externes. Ainsi si un article concerne Dijon et qu'un profil s'intéresse à la Bourgogne, le système, ne sachant pas que Dijon est en Bourgogne, ne considérera pas l'article. La modélisation vectorielle ne permet en pratique pas de prendre en compte la relation qui existe entre Dijon et Bourgogne dans notre exemple. En effet, (Voorhees, 1994) montre que toutes les dimensions sont orthogonales dans le modèle vectoriel et qu'ainsi, tous les éléments de chaque vecteur sont considérés comme indépendants. Nous nous intéressons donc à prendre en compte cette connaissance externe par une expansion des vecteurs.

Dans les travaux sur les systèmes de recherche d'informations de (Voorhees, 1994) les requêtes des utilisateurs sont représentées sous forme de vecteurs. Ces vecteurs sont étendus par l'ajout de synonymes et de méronymes [1]. Plus récemment cette méthode a été adaptée aux systèmes de recommandations par (IJntema et al., 2010). Elle prend la forme d'expansion de vecteurs de profil. Dans ces travaux seuls les vecteurs décrivant le besoin d'information sont étendus, cela dans l'objectif d'augmenter les performances des systèmes en augmentant les mesures de rappel. Dans ces systèmes, les informations ajoutées aux vecteurs sont des informations en relation directe dans les bases de connaissances externes utilisées avec les informations déjà présentes dans les vecteurs. L'ajout d'instances en relation directe avec les instances déjà présentes dans le vecteur profil ne nous permettrait pas à partir de Bourgogne d'ajouter Dijon, mais seulement Côte d'Or et les autres départements de la région. La pertinence de l'article pour le profil serait donc toujours nulle. L'ajout des instances en relation via transitivité avec les instances déjà présentes dans le vecteur profil, permet alors d'ajouter Dijon au vecteur et ainsi de prendre en compte que l'article traite bien d'un contenu en relation avec le contenu souhaité par l'utilisateur. Seulement, l'ajout par transitivité ajoute non seulement Dijon, mais aussi les quatre mille autres communes de la région. La pertinence de l'article pour le profil ne serait donc pas nulle, mais tout de même très faible, alors qu'elle devrait être relativement forte. Afin de pallier à ce problème, nous étendons les vecteurs articles en plus des vecteurs profils. De plus, afin de limiter la taille du vecteur, nous ajoutons les ancêtres de l'instance

1. Désigne une sous partie, par exemple *toit* est un méronyme de *maison*

sélectionnée et non les descendants. Notre méthode se rapporte aux méthodes de recherche de la profondeur de l'ancêtre commun dans un graphe pour l'évaluer la distance sémantique entre deux nœuds.

4 Similarité versus pertinence

La tâche de recommandation est basée sur la comparaison entre profils et articles et s'appuie donc sur leur index commun défini dans la base de connaissances. Les méthodes classiques de recherche d'information ou de recommandation utilisant une modélisation vectorielle déduisent directement la pertinence de la mesure de similarité entre le vecteur représentant le profil et celui représentant l'article. La base théorique, est que le profil peut être considéré comme un article idéal, donc plus un article est similaire à cet article idéal (profil) plus il est pertinent, plus il correspond aux intérêts et besoins de l'utilisateur.

Cependant, nous introduisons ici notre notion de précision entre un profil et un article. En effet, afin de décrire le contenu d'un article, ou les besoins d'un profil, les descripteurs utilisés pour chaque critère peuvent être plus ou moins précis.

Le fait d'avoir pour un critère donné un article proposant un contenu plus précis que celui recherché par un profil n'a pas les mêmes répercussions sur la perception de la pertinence que le fait d'avoir un profil intéressé pour un critère donné par un contenu plus précis que celui proposé par un article. Cela doit donc être pris en compte lors de l'évaluation de la pertinence.

Pour exposer la distinction que nous opérons entre similarité et précision, nous rappelons tout d'abord une définition de similarité, puis nous introduirons notre définition de pertinence.

4.1 Similarité

Puisque nous travaillons avec des modèles vectoriels, nous pouvons utiliser la définition suivante de mesure de similarité entre articles et profils :

$$SimilariteF(\overrightarrow{a}, \overrightarrow{p}) = \frac{\sum \omega_c Similarite_c(\overrightarrow{a_c}, \overrightarrow{p_c})}{\sum \omega_c}$$

$Similarite_c(\overrightarrow{a}, \overrightarrow{p})$ étant la mesure de similarité entre le profil $\overrightarrow{p}$ et l'article $\overrightarrow{a}$ pour le critère spécifique c, tel que $c \in \{Themes, Secteur, Location\}$. Nous utiliserons lors de l'évaluation alternativement les mesures, Cosinus, Jaccard et Euclide en tant que mesure de similarité $Similarite_c(\overrightarrow{a}, \overrightarrow{p})$. ω_c est le coefficient de pondération défini pour le critère c. $\forall i_{x,c} \in I'_c,\ \overrightarrow{p_c} =< i_{1,c}, i_{2,c}, ..., i_{n,c} >$ et $\overrightarrow{a_c} =< i_{1,c}, i_{2,c}, ..., i_{m,c} >$ avec n et $m \in [0; |I'_c|]$. Un ou plusieurs concepts peuvent être définis pour chaque critère. Pour le critère localisation par exemple, nous avons les concepts, pays région, département et ville. Ainsi donc dans la liste des instances disponibles pour la description des articles et profils sur chaque critère il est possible d'utiliser toutes les instances de tous ces concepts.

4.2 Pertinence d'un critère

Ici, nous apportons la distinction entre *pertinence* et *similarité*. Nous nous concentrons notamment sur la gestion de la différence de précision entre la définition des profils et des

articles et son influence sur la mesure pertinence. Par exemple, Dijon étant pour le critère localisation, plus précis que Bourgogne, si un article traite de Dijon et un profil montre un intérêt pour la Bourgogne, la pertinence doit être plus élevée que dans le cas inverse. Afin d'intégrer ce paramètre, nous utilisons un vecteur intermédiaire pour chaque critère. Le sous-vecteur $\overrightarrow{s_c}$ est composé des instances communes entre le vecteur de l'article $\overrightarrow{a_c}$ et celui du profil $\overrightarrow{p_c}$.

Dans la hiérarchie des concepts, les concepts les plus généraux englobent d'autres concepts plus spécifiques, il en va de même dans la hiérarchie d'instances. Les instances hautes dans la hiérarchie sont moins précises que les instances basses.

Si un article traite d'instances de bas niveau et qu'un profil s'intéresse à des instances de haut niveau (de la même branche) la pertinence de l'article pour le profil doit être plus élevée que dans le cas contraire. Car si l'article est plus général que le profil alors il y a une perte de précision par rapport au besoin de l'utilisateur et qui doit être répercutée par une perte de pertinence. Ainsi, nous définissons la pertinence pour un critère c par :

$$Pertinence_c(\overrightarrow{a_c}, \overrightarrow{p_c}) = \frac{\omega'_{1,c} \times Similarite_c(\overrightarrow{a_c}, \overrightarrow{s_c}) + \omega'_{2,c} \times Similarite_c(\overrightarrow{p_c}, \overrightarrow{s_c})}{\omega'_{1,c} + \omega'_{2,c}}$$

Avec S_c le sous-ensemble commun d'éléments de l'ensemble d'instances en relation à la fois avec le profil $I'_{p,c}$ et l'article $I'_{a,c}$; $S_c = I'_{p,c} \cap I'_{a,c}$. $\forall i_{x,c} \in S_c$ le vecteur $\overrightarrow{s_c}$ est composé des éléments de l'ensemble S_c ; $\overrightarrow{s_c} =< i_{1,c}, i_{2,c}, ..., i_{t,c} >$.

Avec cette méthode, il est possible de pondérer de plusieurs façons la différence de précision entre profils et articles, afin de l'adapter aux besoins. Dans notre cas nous utilisons $\omega'_{1,c} = 1$ et $\omega'_{2,c} = 4$ car nous considérons que la perte de précision du profil par rapport à l'article ne doit pas influencer plus de 20% du résultat. Par contre, la perte de précision de l'article par rapport au profil doit influencer fortement le résultat, ici 80%. Il est toutefois possible de modifier ces valeurs, et il est aussi possible de les gérer de façon distincte selon le critère considéré.

4.3 Pertinence globale : *Relevancy measure*

La pertinence globale $Relevancy(\overrightarrow{a}, \overrightarrow{p})$ est la somme des mesures de pertinence pour chacun des critères, éventuellement pondérées. Cette mesure est utilisée dans notre prototype pour trier les résultats (articles) proposés à l'utilisateur en fonction de son profil :

$$Relevancy(\overrightarrow{a}, \overrightarrow{p}) = \frac{\sum \omega_c * Pertinence_c(\overrightarrow{a_c}, \overrightarrow{p_c})}{\sum \omega_c}$$

Pour illustrer notre propos, considérons la figure 1. Grâce à notre approche, la valeur de pertinence pour le cas A.2 est plus élevée que dans le cas A.1, car dans le cas A.2, a et p ont des ancêtres communs plus généraux que dans le cas A.2. En outre, les cas B1 et B2 illustrent le problème de la précision. Avec notre méthode asymétrique, la valeur de similarité entre a et p dans le cas B.1 est plus élevée que dans le cas B.2. Les besoins des utilisateurs sont plus spécifiques que les informations de l'article (sur ce critère), donc l'article est moins pertinent pour l'utilisateur.

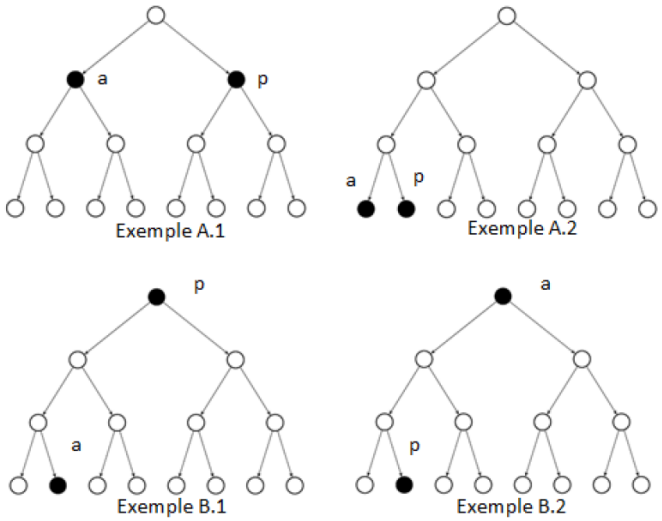

FIG. 1 – *Exemples de profils p et d'articles a pour un critère donné*

5 Expérimentations

Nous avons défini une méthode couplant une expansion des vecteurs de profils et d'articles et une prise en compte de la différence de précision entre les descriptions fournies par ces vecteurs. Nous évaluons donc ici ces deux supports fondamentaux de notre approche.
Pour cela nous avons élaboré un jeu de test comportant 10 profils de lecteurs et 70 articles (ce qui correspond à la production quotidienne d'articles pour cette société). Ce jeu de données est suffisamment conséquent pour répondre aux besoins de l'évaluation, mais de taille raisonnable pour permettre à un expert d'établir une recommandation manuelle de référence. Les mesures de pertinence expérimentée ici sont appliquées dans un espace vectoriel permettant l'utilisation de nombreuses méthodes d'évaluation de la similarité. Nous avons donc établi notre benchmark sur trois des plus classiques : Similarité Cosinus, Similarité Jaccard et distance euclidienne.

5.1 Méthodes d'évaluation

Evaluation Binaire Pour évaluer la recommandation de ces algorithmes nous nous basons sur les mesures classiques de précision [2] et de rappel [3].
Cette évaluation étant binaire, nous avons besoin de résultats binaires à partir d'algorithmes de recommandation. Or ils fournissent des articles de façon triée à l'aide d'une valeur de pertinence (entre 0 et 1). Nous avons donc défini un seuil au-delà duquel un item est recommandé et en dessous duquel il ne l'est pas. Le seuil de 0,5 a été choisi pour l'évaluation de la pertinence binaire.

2. Nombre d'articles pertinents retrouvés par rapport au nombre d'articles total proposé en réponse d'une requête.
3. Nombre d'articles pertinents retrouvés au regard du nombre d'articles pertinents que possède la base de données.

Afin de considérer à la fois la précision [4] et le rappel dont l'optimisation est un centre d'intérêt important dans le cadre des systèmes de recommandation, (Lewis et Gale, 1994) propose une mesure simple : la F-mesure. La F-mesure est une combinaison pondérée de précision et de rappel qui produit des scores allant de 0 à 1.

Evaluation de rang. Pour évaluer l'ordre des articles recommandés par les algorithmes, nous utilisons les deux mesures de corrélation linéaire de rang les plus populaires : le rho de Spearman et le tau de Kendall. Ces deux métriques produisent des scores allant de -1 à 1. 0 étant l'absence de similitude, 1 la similitude complète et -1 l'inverse.

5.2 Evaluation de l'expansion des vecteurs

Dans cette partie nous évaluons l'intérêt l'expansion des vecteurs profils et articles. Pour cela nous restons dans un contexte où la pertinence d'un article pour un profil est déduite directement de leur similarité. Nous confrontons deux algorithmes : celui utilisant le modèle vectoriel classique sans expansion (méthode *C*), et avec expansion des deux vecteurs articles et profils par l'ajout d'instances (méthode *B*).

Algorithmes	Précision	Rappel	F1-mesure	Tau de Kendall	Rho de Spearman
COSINUS B	**0.916**	0.453	**0.607**	**0.830**	**0.894**
COSINUS C	0.883	0.181	0.301	0.713	0.694
JACCARD B	0.883	0.150	0.256	0.819	0.886
JACCARD C	0.883	0.150	0.256	0.712	0.693
EUCLIDE B	0.396	**0.985**	0.565	0.649	0.734
EUCLIDE C	0.549	0.495	0.521	0.549	0.615

TAB. 1 – *Comparaison vecteurs étendus et non étendus par des mesures d'évaluation binaires et de corrélation de rang.*

Les résultats de l'évaluation de la recommandation en utilisant comme mesure de pertinence la similarité directe entre les vecteurs classiques et les vecteurs étendus sont présentés dans la table 1. Pour chaque algorithme d'évaluation de la pertinence par mesure de similarité (Jaccard, Cosinus, Euclide),lLa F1-mesure montre que lorsque les vecteurs sont étendus afin de prendre en compte les connaissances de la base de connaissances (méthode *B*), les résultats sont au moins aussi bons qu'avec les vecteurs classiques (méthode *C*). L'évaluation de l'ordre des articles rangés par les différents algorithmes montre les mêmes résultats. Nous pouvons observer une perte de précision avec la distance euclidienne. Ce problème de perte de précision a déjà été expliqué par (Voorhees, 1994) avec sa propre méthode d'expansion du vecteur. En effet, l'expansion des vecteurs vise l'amélioration du rappel et comme le montrent les résultats, cela peut avoir un coût en précision. Nous confirmons ici les résultats de (Middleton et al., 2004) et (IJntema et al., 2010) quant à l'intérêt de l'expansion de vecteurs, et montrons que notre approche d'expansion ontologique s'inscrit dans ce constat.

4. Dans cette section, le mot *précision* fait uniquement référence à la mesure de la précision de la recommandation, dans le sens précision / rappel.

5.3 Evaluation de la *Relevancy measure*

Nous nous intéressons ici à l'évaluation de l'apport fourni par la prise en compte de la différence de précision entre la description des profils et des articles lors de la mesure de la pertinence. La métrique *Relevancy measure*, permet de prendre en compte lors de l'évaluation de la pertinence d'un article son adéquation avec le degré de spécificité par rapport à celui souhaité par le lecteur. Ainsi nous comparons dans cette section les résultats fournis lors de l'utilisation de vecteur étendu par une mesure de pertinence directement déduite de la similarité des vecteurs (méthode *B*) et par notre métrique *Relevancy measure* (méthode *A*).

Algorithmes	Précision	Rappel	F1-mesure	Tau de Kendall	Rho de Spearman
COSINUS A	0.856	0.971	**0.910**	**0.836**	**0.898**
COSINUS B	0.916	0.453	0.607	0.830	0.894
JACCARD A	**0.928**	0.588	0.720	0.836	0.896
JACCARD B	0.883	0.150	0.256	0.819	0.886
EUCLIDE A	0.566	0.971	0.715	0.728	0.817
EUCLIDE B	0.396	**0.985**	0.565	0.649	0.734

TAB. 2 – *Comparaison des mesures de pertinence avec et sans prise en compte des différences de précisions des descriptions par évaluation binaires et de corrélation de rang.*

Les résultats de l'évaluation de la recommandation permettant de distinguer d'une part la méthode de mesure de la pertinence basée sur la similarité directe et notre méthode *Relevancy measure* d'atre part, sont présentées dans la table 2. Elles utilisent toutes les deux des vecteurs étendus.

Les deux méthodes d'évaluation (Tau de Kendall ou Rho de Spearman), indiquent que la méthode A fournit un meilleur classement d'articles. En ce qui concerne la F1-mesure, elle indique aussi que la méthode A fournit les meilleurs résultats, c'est à dire proposant le meilleur rapport entre précision et rappel.

En conclusion, ces évaluations témoignent de la pertinence de notre approche et mettent en avant que les paramètres les plus efficaces pour la recommandation d'articles sont d'effectuer une expansion des vecteurs profils et articles, et de prendre en compte les différences de précision entre l'expression du besoin et la description du contenu des articles comme le permet notre méthode.

6 Conclusion et travaux futurs

Dans cet article, nous avons présenté l'adaptation d'un système basé sur le modèle vectoriel de recommandation à notre méthode spécifique d'indexation qui définit sémantiquement les articles et les profils dans une base de connaissances par l'intermédiaire des relations avec les connaissances du domaine prédéfinies dans celle-ci. Nous avons présenté notre approche qui répond aux manques de l'état de l'art sur les points de gestion du degré de precision du besoin et de leur comparaison avec l'offre.

Nous avons présenté notre approche qui intègre une *expansion ontologique* des vecteurs d'articles et de profils d'utilisateurs. Nous avons distingué et défini les notions de *similarité* et de *pertinence*. Enfin, nous avons évalué nos algorithmes en utilisant à la fois une méthode d'évaluation binaire et de corrélation de rang afin de montrer les apports de notre approche. Cette

évaluation montre que notre approche fournit le meilleur classement d'articles, notamment quand elle est utilisée avec une mesure de comparaison de vecteurs *cosinus*.
Nous projetons d'évaluer notre méthode en intégrant les comportements de l'utilisateur lors de l'utilisation de l'outil de recommandation. De plus, nous enrichissons la base de connaissances afin d'améliorer encore la pertinence.

Notre démarche étant expérimentalement validée, le prolongement de ces travaux s'intéresse à son passage à l'échelle. Cependant une autre méthode que la méthode vectorielle devra être appliquée car selon nos premiers travaux, elle génère des effets de bord lorsque le volume de données devient conséquent.
Par ailleurs, nous souhaitons améliorer la recommandation en nous appuyant sur des algorithmes provenant de l'étude des graphes afin de calculer la pertinence entre les profils et les articles. L'information que nous utilisons est déjà structurée dans une ontologie (plus efficace dans la transmission sémantique que le modèle vectoriel), il semble donc inutile de restructurer l'information sous forme de vecteurs pour effectuer des comparaisons entre les articles et les profils à moins que cela n'apporte un réel gain en temps de calcul. Certaines approches permettant de comparer des instances dans une base de connaissances existent déjà (Albertoni et Martino, 2006), (Ehrig et al., 2004) et sur lesquelles nous nous appuierons.

Références

Ahn, J.-w., P. Brusilovsky, J. Grady, D. He, et S. Y. Syn (2007). Open user profiles for adaptive news systems : help or harm ? pp. 11. ACM Press.

Albertoni, R. et M. D. Martino (2006). Semantic similarity of ontology instances tailored on the application context. In *Lecture Notes in Computer Science Volume 4275*, pp. 1020–1038. Springer.

Billsus, D. et M. J. Pazzani (1999). A personal news agent that talks, learns and explains. pp. 268–275. ACM Press.

Cunningham, H. (2002). GATE, a general architecture for text engineering. *Computers and the Humanities 36*(2), 223–254.

d'Aquin, M., A. Schlicht, H. Stuckenschmidt, et M. Sabou (2009). Criteria and evaluation for ontology modularization techniques. In *Modular Ontologies*, Volume 5445, pp. 67–89. Springer Berlin Heidelberg.

Ehrig, M., P. Haase, M. Hefke, et N. Stojanovic (2004). Similarity for ontologies - a comprehensive framework. In *In Workshop Enterprise Modelling and Ontology : Ingredients for Interoperability, at PAKM 2004*.

Fellbaum, C. (1998). *WordNet : an electronic lexical database*. Cambridge, Mass : MIT Press.

Getahun, F., J. Tekli, R. Chbeir, M. Viviani, et K. Yetongnon (2009). Relating RSS News/Items. In *Web Engineering*, Number 5648 in Lecture Notes in Computer Science, pp. 442–452. Springer Berlin Heidelberg.

IJntema, W., F. Goossen, F. Frasincar, et F. Hogenboom (2010). Ontology-based news recommendation. pp. 1. ACM Press.

Lewis, D. D. et W. A. Gale (1994). A sequential algorithm for training text classifiers. In *SIGIR '94*, pp. 3–12. Springer London.

Liu, J., P. Dolan, et E. R. Pedersen (2010). Personalized news recommendation based on click behavior. pp. 31. ACM Press.

Middleton, S. E., N. R. Shadbolt, et D. C. De Roure (2004). Ontological user profiling in recommender systems. *ACM Transactions on Information Systems 22*(1), 54–88.

Rao, K. et V. Talwar (2008). Application domain and functional classification of recommender Systems—A survey. *DESIDOC Journal of Library and Information Technology 28*(3), 17–35.

Resnick, P., N. Iacovou, M. Suchak, P. Bergstrom, et J. Riedl (1994). GroupLens : an open architecture for collaborative filtering of netnews. pp. 175–186. ACM Press.

Salton, G. (1971). The SMART retrieval system - experiments in automatic document processing.

Voorhees, E. M. (1994). Query expansion using lexical-semantic relations. In *SIGIR '94*, pp. 61–69. Springer London.

Summary

Today in the commercial and financial sectors, staying informed about economic news is crucial and involves targeting good articles to read, because the huge amount of information. To address this problem, we propose an innovative article recommendation system, based on the integration of a semantic description of articles and on a knowledge ontological model. We support our recommendation system on an intrinsically efficient vector model that we have perfected to overcome the confusion existing in models between the concepts of similarity and relevancy that does not take into account the effects of the difference in the accuracy of the semantic descriptions precision between profiles and articles, on the perceived relevancy to the user. We present in this paper a new evaluation of the relevancy adapted to vector model.