

Filtrage qualitatif d'informations linguistiques

Qualitative pattern matching with linguistic information

M. Boughanem

Y. Loiseau

H. Prade

IRIT, Univ Paul Sabatier, 31062 Toulouse, {boughane,loiseau,prade}@irit.fr

Résumé :

Cet article propose la contrepartie symbolique d'une méthode d'évaluation de requêtes flexibles face à une base de données floues, pour permettre l'interrogation, en termes de mots (assortis de priorités), d'une base de données où l'information est aussi exprimée par des mots. L'approche se base sur une ontologie qualitative qui distingue entre synonymie approchée et spécialisation. Ces idées sont aussi appliquées à la recherche documentaire multilingue.

Mots-Clés :

Filtrage flou, recherche d'information, ontologie

Abstract:

This article presents the symbolic counterpart of a flexible fuzzy database-flexible request evaluation method in order to query a database in terms of words (with priority), the information being also expressed by words. The approach is based on a qualitative ontology, distinguishing between approximate synonymy and specialisation. These ideas are also applied to multilingual information retrieval.

Keywords:

Fuzzy pattern matching, information retrieval, ontology

1 Introduction

La prise en compte des préférences de l'utilisateur dans les systèmes d'information nécessite d'autoriser des requêtes flexibles [1], ce qui permet au système d'ordonner les résultats. Des approches basées sur les ensembles flous, développées dans cette perspective, peuvent être appliquées aux bases de données classiques aussi bien que floues. Un outil appelé 'filtrage flou' (*fuzzy pattern matching*, FPM) [7], proposé depuis vingt ans dans le cadre de la théorie des possibilités, calcule dans quelle mesure il est possible, et dans quelle mesure il est certain, qu'une information (éventuellement imprécise ou floue) satisfasse à une requête flexible exprimée au moyen d'ensembles flous représentant les préférences de l'utilisateur. En cas de requêtes formées de conjonctions (resp. disjonc-

tions) de contraintes élémentaires, les mesures de possibilité et de certitude sont combinées par l'opération minimum (resp. maximum) en préservant leurs sémantiques. Cette idée a aussi été étendue à la recherche d'information où des documents sont décrits par des mots clés dont la pertinence est plus ou moins certaine, ou seulement possible. De même, la requête peut utiliser des mots clés plus ou moins obligatoires, ou optionnels [10].

Dans cet article, on rappelle tout d'abord une approche récente [9] inspirée par ces idées, qui a pour but de traiter des requêtes définies en terme d'étiquettes linguistiques qui peuvent être pondérées afin d'exprimer les préférences de l'utilisateur et mieux décrire son besoin d'information. L'évaluation de la mise en correspondance d'une étiquette de la requête et de données est basée sur des degrés de possibilité et de certitude de similarité sémantique calculés au moyen d'un réseau pondéré (ontologie) associé à chaque domaine d'attribut. Cette mise en correspondance utilise donc des relations entre les termes des étiquettes, ce qui permet d'étendre la description de l'information en se basant sur des dépendances sémantiques. Les étiquettes ne sont plus explicitement associées à des ensembles flous comme dans le FPM, mais leurs relations sémantiques sont toujours supposées être évaluées en terme i) de possibilité que deux étiquettes réfèrent à une même chose, et ii) de spécialisation de sens (la certitude correspondant à un degré d'inclusion); le processus d'évaluation demeure qualitatif. L'article applique ensuite ces idées à la recherche d'informations multilingues, où des documents sont représentés par des termes extraits et les

requêtes s'expriment par des conjonctions et/ou disjonctions pondérées de mots clés.

2 Correspondance par ontologie

Les mesures de similarité sémantique entre mots ont été abondamment étudiées dans la littérature sur la recherche d'information, en utilisant par exemple des distances entre noeuds dans une taxinomie, ou basées sur une probabilité d'information commune (par ex. [11]). Dans le même esprit, une stratégie habituelle quand une requête échoue est de la remplacer par des requêtes similaires (par ex. [2]), générées en utilisant des ontologies. En restant inspirés par le FPM, nous présentons tout d'abord une approche basée sur les réseaux sémantiques pondérés par des degrés de possibilité et de nécessité [9] évitant la réécriture des requêtes.

Considérons une base de données dont les éléments sont décrits par un ensemble d'attributs identifiés $i=1, n$. Les valeurs de ces attributs sont des termes $t_i^j \in T_i'$, où T_i' est le vocabulaire relatif à l'attribut i . Le cas plus général de données imprécises, c.à.d. représentées par des disjonctions pondérées, n'est pas considéré ici (cf. [9] pour ce cas). Comme chaque attribut contribue à l'information, une donnée est une conjonction de telles valeurs symboliques : $T' = \bigwedge_{i \in [1;n]} T_i'$ et $\exists j, T_i' = \{t_i^j\}$. La même notation est utilisée (sans prime) pour les requêtes. Cependant, on autorise des disjonctions pondérées de termes au niveau de chaque attribut dans les requêtes. En effet, l'utilisation de valeurs composées pour les attributs permet de définir de nouveaux concepts. La valeur T_i de l'attribut i pour une requête s'exprime à l'aide d'une collection de termes pondérés. La requête est alors : $R = \bigwedge_{i \in A(R)} T_i$, où $T_i = \bigvee_{j \in R(T_i)} (\lambda_i^j, t_i^j)$, $A(R)$ étant l'ensemble des attributs pris en compte, $R(T_i)$ l'ensemble des termes de R pour l'attribut i . Les termes, et plus généralement les expressions symboliques, sont mises en relations au travers d'« ontologies possibilistes » O_i pour chaque attribut i . Les relations dans O_i sont modélisées

par des degrés de possibilité et de nécessité : $\Pi(t_i^j, t_i^k) = \Pi(t_i^k, t_i^j)$ représente dans quelle mesure t_i^j et t_i^k peuvent représenter la même chose. $N(t_i^j, t_i^k)$ estime à quel point il est certain que t_i^k est une spécialisation de t_i^j . En particulier, $N(t_i^j, t_i^k) = N(t_i^k, t_i^j) = 1$ représente la synonymie parfaite. Ces mesures doivent satisfaire les propriétés suivantes :

1. $\Pi(t_i^j, t_i^j) = N(t_i^j, t_i^j) = 1$
2. $\Pi(t_i^j, t_i^k) = \Pi(t_i^k, t_i^j)$,
3. $N(t_i^j, t_i^k) > 0 \Rightarrow \Pi(t_i^j, t_i^k) = 1$ (si c'est certain, cela doit être totalement possible).

Les informations de degré figurant dans l'ontologie peuvent être complétées en utilisant ces propriétés, ainsi que les deux formes de transitivité suivantes :

$$N(t_i^j, t_i^h) \geq \min(N(t_i^j, t_i^k), N(t_i^k, t_i^h)), \quad (1)$$

$$\Pi(t_i^j, t_i^h) \geq N(t_i^j, t_i^k) * \Pi(t_i^k, t_i^h). \quad (2)$$

avec $a * b = b$ si $b > 1 - a$ et $a * b = 0$ sinon. (1) représente la transitivité de la spécialisation [12]. La « transitivité hybride » (2) spécifie que si t_i^k spécialise t_i^j et si t_i^k et t_i^h peuvent représenter la même chose, alors la signification de t_i^j et t_i^h se recoupent aussi ; voir [6] pour une preuve de (1)-(2). Le degré de certitude de la synonymie de t_i^j et t_i^h peut se calculer comme $\min(N(t_i^j, t_i^h), N(t_i^h, t_i^j))$. En utilisant (1), on peut vérifier que ce degré est maximum transitif.

De plus, on peut introduire des niveaux d'importance dans les requêtes. Soit ω_i l'importance de ce qui est requis quant à la valeur de l'attribut i . L'évaluation d'une requête consiste alors à retrouver toutes les données T' telles que $\Pi(R, T')$ ou $N(R, T')$ soient non nulles, en calculant les conjonctions pondérées $\Pi(R, T') = \min_{i \in A(R)} \max(1 - \omega_i, \pi_i)$ (3a) et $N(R, T') = \min_{i \in A(R)} \max(1 - \omega_i, \nu_i)$, (3b) où $\pi_i = \max_{j \in R(T_i)} \min(\lambda_i^j, \Pi(t_i^j, t_i^k))$ et $\nu_i = \max_{j \in R(T_i)} \min(\lambda_i^j, N(t_i^j, t_i^k))$ sont des disjonctions pondérées reflétant les disjonctions figurant dans la requête au niveau

de chaque attribut. Les résultats sont classés d'abord en fonction des valeurs décroissantes de $N(R, T')$, puis en fonction des valeurs décroissantes de $\Pi(R, T')$ pour les T' ayant les mêmes valeurs de $N(R, T')$.

Exemple (sans pondération) avec deux ontologies simplifiées pour des types de logements de vacances (fig. 1) et des lieux (fig. 2). Notons

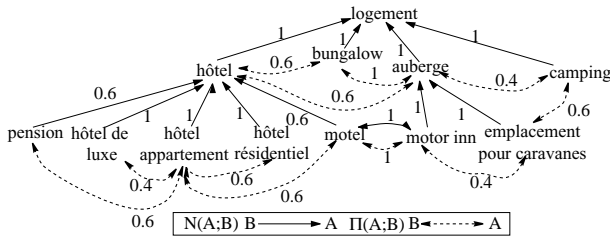


Figure 1 – Ontologie des logements

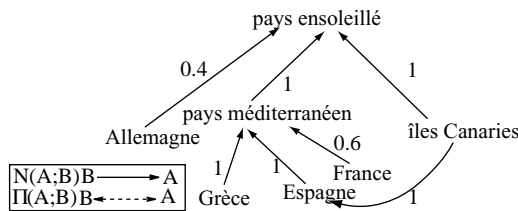


Figure 2 – Ontologie des lieux

que des mots comme *bungalow* et *auberge* ne sont considérés que comme des synonymes possibles (ou au moins peuvent offrir les mêmes services), et qu'il peut donc exister des *bungalows* qui ne sont pas des *auberges*. Les valeurs des degrés sont qualitatives. Seul l'ordre induit entre elles est important. Soit la requête : $R = (\text{hôtel} \vee \text{auberge}) \wedge (\text{pays ensoleillé})$ avec la base de données décrite dans la table 1.

Tableau 1 – Exemple de base de données

	logement	lieu	prix
A	hôtel	Angleterre	[65,70]
B	pension	Espagne	25
C	bungalow	Grèce	bon marché
D	motel	France	modéré

Nous avons pour la première donnée :

$$\pi_{\text{logement}} = \max(\Pi(\text{hôtel}, \text{hôtel}), \Pi(\text{auberge},$$

hôtel)) et $\pi_{\text{lieu}} = \Pi(\text{pays ensoleillé}, \text{Angleterre})$. De plus, $\Pi(\text{hôtel}, \text{hôtel}) = 1$ et l'ontologie donne $\Pi(\text{auberge}, \text{hôtel}) = 0.6$, donc $\pi_{\text{logement}} = 1$. Comme *Angleterre* n'est pas lié à *pays ensoleillé*, nous avons $\pi_{\text{lieu}} = 0$. $\Pi(R, T'_A) = \min(\pi_{\text{logement}}, \pi_{\text{lieu}}) = 0$. Pour B, $N(R, T'_B) = \min(\max(N(\text{hôtel}, \text{pension}), N(\text{auberge}, \text{pension})), N(\text{pays ensoleillé}, \text{Espagne})) = \min(\max(0.6, 0), 1) = 0.6$. Les autres données sont évaluées de la même manière, ce qui donne les scores (Π, N) : A : (0,0) ; B : (1,0.6) ; C : (1,0) ; D : (1,0.6). Nous obtenons donc la classification : B puis D et enfin C. Pour classer D après B, il faut raffiner la procédure de base en remarquant que D a une nécessité de 0.6 sur les deux critères, alors que B a un meilleur résultat sur un des critères de la requête (le lieu), et 0.6 sur l'autre.

Considérons à présent une requête sur les prix : $R = (\text{modéré})$. Pour faire correspondre des valeurs numériques à des termes linguistiques, nous devons définir, pour un vocabulaire de prix donné, une représentation de ces termes, tels que « modéré », par rapport à des valeurs de prix, c'est-à-dire définir une distribution de possibilité de prix pour chaque terme de ce vocabulaire. Ceci peut être réalisé comme sur la figure 3. Remarquez que les prix sont arbitraires, et peuvent (ou doivent) être définis par l'utilisateur, puisque le sens de « bon marché » par exemple dépend du contexte et de l'utilisateur.

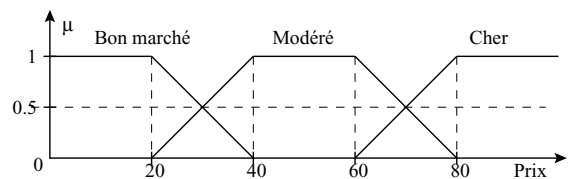


Figure 3 – Distributions des prix

Évaluons la requête avec le FPM classique [5] : $\Pi(T; T') = \sup_{u \in U} \min(\mu_T(u), \pi_{T'}(u))$, $N(T; T') = \inf_{u \in U} \max(\mu_T(u), 1 - \pi_{T'}(u))$. où μ_T est la fonction d'appartenance associée au terme T et $\pi_{T'}$ la distribution de possibilité attachée à T' . On obtient alors :

Remarquons que la nécessité $N(\text{modéré}, \text{mo-}$

Ligne	Π	N	Rang
A	0.75	0.5	2
B	0.25	0.25	3
C	0.5	0	4
D	1	0.5	1

déré) est de 0.5 car le terme « modéré » est fbu et qu'on ne peut être totalement sûr qu'un prix déclaré comme tel le soit effectivement au degré 1. La propriété $N(t_i^j, t_i^j) = 1$, requise plus haut, suppose que le terme t_i^j est considéré comme ayant un sens non fbu. Si ce n'est pas le cas, il convient de fixer une valeur entre 1 et 1/2 pour $N(t_i^j, t_i^j)$ dans l'ontologie. Il est intéressant d'examiner une évaluation de la première donnée de la table pour « modéré » et « cher ». En effet, on a $\Pi(\text{modéré}, [65,70]) = \sup_{x \in [65,70]} \mu_{\text{modere}}(x) = 0.75$ et $N(\text{modéré}, [65,70]) = \inf_{x \in [65,70]} (\mu_{\text{modere}}(x)) = 0.5$. De même, $\Pi(\text{cher}, [65,70]) = 0.5$ et $N(\text{cher}, [65,70]) = 0.25$. Nous n'avons plus les contraintes définies précédemment (notamment la propriété 3), puisque les données sont imprécises, mais seulement les contraintes de FPM classique, c.à.d. $\Pi(T; T') \geq N(T; T')$. Comme $[65, 70]$ est plus proche de « modéré » que de « cher », les degrés sont plus élevés. Ce type d'évaluation peut être combiné avec le précédent par l'opérateur *min*, ce qui permet de traiter des requêtes prenant en compte des données hétérogènes, exprimées de différentes manières (termes, valeurs, intervalles).

La combinaison des degrés obtenus par filtrage qualitatif et de ceux calculés par FPM classique, pose le problème de la commensurabilité des échelles. En effet, le FPM appliqué à une fonction d'appartenance continue peut retourner tout réel de $[0,1]$, alors que le filtrage qualitatif utilise une échelle discrète, avec un nombre restreint de niveaux, représentés numériquement pour des raisons pratiques. En supposant que le filtrage qualitatif utilise une échelle finie et homogène, comme $\{0,0.2,0.4,0.6,0.8,1\}$, les rangs obtenus par le filtrage fbu peuvent être approchés par la plus proche valeur dans cette échelle. Puisque les

deux procédures sont basées sur des degrés de possibilité et de nécessité, ceci permet de combiner les évaluations élémentaires calculées pour chaque attribut.

3 Recherche d'information

En recherche d'information, la pertinence d'un document est évaluée par rapport à une requête. Typiquement, celle-ci est une liste de mots clés pouvant être pondérés et combinés par des *et* et *ou*, et les documents sont représentés par une liste pondérée de leurs mots significatifs. Le poids d'un terme t_i dans un document est estimé en combinant la fréquence tf_{ij} de t_i dans D_j , et la fréquence inverse des termes : $idf_i = \log(d/df_i)$, où df_i est le nombre de documents contenant t_i et d est le nombre total de documents. Le document D_j est alors représenté par : $D_j = \{(\rho_{ij}, t_i), i = 1, n\}$ où n est le nombre total de termes dans l'ontologie (ou le vocabulaire concerné) et ρ_{ij} est le poids du terme t_i dans le document D_j , calculé à partir de tf_{ij} et idf_i , souvent par leur produit. Dans le cas multilingue, les documents sont dans des langues différentes et quelle que soit la langue de la requête, le système doit retourner les documents pertinents. Nous nous proposons ici d'utiliser les notions de la section 2 pour évaluer la correspondance entre requête et documents, plutôt que les calculs de distances habituels en recherche d'information. Cette approche permet d'exploiter la connaissance sémantique modélisée dans l'ontologie directement dans l'évaluation, et évite ainsi l'étape de reformulation de requête couramment utilisée.

3.1 Ontologie multilingue

Une ontologie multilingue est définie comme suit, en s'inspirant d'EuroWordNet. Un « synset » est un ensemble de termes synonymes tels que : $S_n = \{t_{hi} \in \mathcal{T}\}$ avec $\forall(i, j), t_{hi} \neq t_{hj}$, $\Pi(t_{hi}, t_{hj}) = 1$ et $N(t_{hi}, t_{hj}) = N(t_{hi}, t_{hj}) = 1$. Les termes d'un synset sont donc considérés comme parfaitement synonymes. Chaque terme appartient à un et un seul synset et est un sy-

nonyme de tous les autres termes du synset, conformément à (1) et (2), en considérant qu'un terme t_i^j est caractérisé par son sens et pas seulement son libellé (dans le cas de termes polysèmes ou homonymes). Les relations de possibilité et de nécessité pour définir des synonymies et spécialisations approchées sont définies entre synsets, comme ils l'étaient entre les noeuds à la section 2. Une ontologie multilingue est composée d'un ensemble d'ontologies dans différentes langues. Les synsets des différentes ontologies sont mis en relation les uns avec les autres par des degrés de possibilité et nécessité de 1 pour modéliser les équivalences de termes entre langues. Comme dans l'ontologie mono-

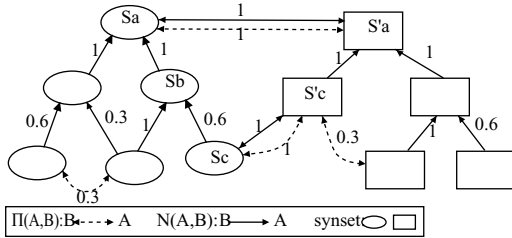


Figure 4 – Structure de l'ontologie multilingue

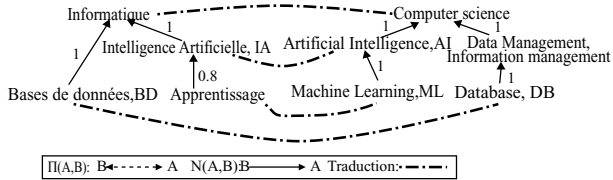


Figure 5 – Exemple d'ontologie multilingue

lingue, les relations inter-langues peuvent être étendues en utilisant (1) et (2). Puisque ces propriétés sont indépendantes de la langue, la correspondance entre une requête dans une langue et un document dans une autre langue peut être déduite. Cependant, deux ontologies dans deux langues distinctes peuvent avoir une architecture différente, comme sur la figure 4. Ici, le synset S_a se traduit par S'_a , et S_c par S'_c . Mais la traduction de S_b n'est pas définie dans l'ontologie. Néanmoins, les relations de possibilité et nécessité entre S_b et S'_a peuvent être évaluées en utilisant respectivement (2) et (1).

3.2 Indexation possibiliste

La pertinence d'un terme de la requête vis-à-vis du document doit être évaluée en utilisant les degrés de nécessité et de possibilité. Pour être homogène à la représentation de l'ontologie, la correspondance entre document et synsets doit être estimée avec les mêmes degrés, en tenant compte du poids des termes du document. Considérons chaque document comme un ensemble fbu (par ex. [3, 8]) de ses mots significatifs. Un poids ρ_{hij} d'un terme t_{hi} dans le synset S_h relativement au document D_j est donc un degré de pertinence de D_j par rapport à t_{hi} : $\rho_{hij} = \mu_{D_j}(t_{hi})$. Étant donné un synset $S_h = \{t_{hi}, i = 1, p\}$, nous voulons estimer dans quelle mesure il décrit le document D_j , c.a.d. $\Pi(S_h, D_j)$ et $N(S_h, D_j)$. Comme les termes dans le synset sont synonymes, chacun d'eux est supposé pouvoir décrire identiquement le document. Notons que dans les systèmes classiques, les requêtes sont souvent étendues en agrégeant les synonymes par l'opérateur *ou*. Nous avons $\Pi(S_h, D_j) = \max_i(\Pi(t_{hi}, D_j))$ et $N(S_h, D_j) = \max_i(N(t_{hi}, D_j))$. En considérant que le poids ρ_{hij} est un degré intermédiaire entre la possibilité et la nécessité pour le terme de décrire le document, des degrés de possibilité et nécessité seront calculés ainsi [10] :

si $\rho_{hij} < \frac{1}{2}$, $\Pi(t_{hi}, D_j) = 2\rho_{hij}$; $N(t_{hi}, D_j) = 0$;
 $\Pi(t_{hi}, D_j) = 1$; $N(t_{hi}, D_j) = 2\rho_{hij} - 1$ sinon.
 Cette transformation permet de distinguer entre trois situations remarquables des représentations probabiliste (P) et possibiliste (Π, N) :
 i) certitude : $P=1$, $\Pi=N=1$;
 ii) certitude du contraire : $P=0$, $\Pi=N=0$;
 iii) indétermination : $P=1/2$, $\Pi=1, N=0$.
 On pourrait envisager une évaluation de la possibilité et de la certitude de pertinence d'un document par rapport à un mot-clé, et plus généralement par rapport à une requête, qui prendrait en compte non seulement la statistique d'apparition des mots, mais aussi leur contexte d'apparition (par exemple titre, introduction, corps du texte, conclusion,...). Par ailleurs, cette approche introduit de nouveaux problèmes. En effet, l'ontologie regroupe des concepts, qui peuvent être des groupes de mots,

et doivent donc être reconnus comme tels dans le texte. La méthode habituelle est d'utiliser un « lématiser » pour réduire les mots à leur forme canonique, et un analyseur syntaxique pour identifier les concepts et expressions, comme dans [4] par exemple, mais cet aspect n'est pas considéré ici. Une alternative sera abordée plus loin (cf. partie 3.4).

Un document est donc indexé à partir de mesures statistiques de ses termes significatifs (s'ils figurent dans l'ontologie). A titre d'exemple, considérons un document anglais D , ayant les degrés (Π, N) suivant donnés par la transformation ci-dessus, et représenté par la table suivante. Ceci suggère que ce docu-

Term	ρ	Π	N
Computer Science	0	0	0
Database	0.6	1	0.2
Artificial Intelligence	0.2	0.4	0
AI	0.7	1	0.4
Machine learning	0.8	1	0.6

ment parle d'intelligence artificielle, plus précisément d'apprentissage automatique, appliqué aux bases de données. Même si « artificial intelligence » et « AI » ont la même signification, leurs poids sont différents puisque d'un point de vue statistique, le terme « AI » est plus fréquent que « artificial intelligence » dans le document. Le degré (Π, N) entre le synset {Artificial Intelligence, AI} et D est donc $(\max(0.4, 1), \max(0, 0.4)) = (1, 0.4)$. Le degré de *Computer Science* est 0, car même si le document traite d'informatique, ce terme n'y apparaît pas.

3.3 Évaluation de requête

Évaluer une requête revient à estimer dans quelle mesure le document constitue une réponse pertinente au besoin d'information exprimé par la requête. Illustrons l'évaluation sur un exemple. La figure 5 montre un fragment d'une ontologie entre français et anglais. Notons que dans cette ontologie, la nécessité entre *Apprentissage* et *IA* n'est que de 0.8 car ce

terme peut aussi avoir le sens d'*éducation* par exemple. Cependant, dans ce contexte, il se traduit parfaitement par *Machine Learning*. Soit la requête : $R = BD \wedge$ Intelligence Artificielle. Appliquons (3a)-(3b) avec des degrés d'importance égaux à 1. Nous avons $\Pi(BD, Database) = N(BD, Database) = 1$ puisqu'ils appartiennent à des synsets en correspondance directe. En utilisant (1) et (2), on peut déduire que, au pire, $N(BD, D) = 0.2$ et $\Pi(BD, D) = 1$. Pour évaluer la partie *Intelligence Artificielle* de la requête sur D , {*Artificial Intelligence, AI*} et {*ML*} doivent être pris en compte. En effet, même si le terme « Artificial Intelligence », qui est la traduction directe de la requête, est moins fréquent que « Machine Learning » dans ce document, nous savons que *ML* (le concept) EST dans *AI* (le domaine), le document peut donc être pertinent. Prendre en compte {*Artificial Intelligence, AI*} est évident puisque le cas est identique à *BD* et *Database*, et donc $(\Pi, N) = (1, 0.4)$, mais il y a deux façons d'évaluer les valeurs de possibilité et nécessité pour {*ML*}, en utilisant la transitivité : soit spécialiser puis traduire (via *Apprentissage*), soit traduire puis spécialiser (via *AI*). Ainsi, avec l'ontologie :

$IA \rightarrow AI \rightarrow ML$ donne : $\Pi(IA, ML) = 1$ et $N(IA, ML) = 1$,

$IA \rightarrow Apprentissage \rightarrow ML$ donne :

$\Pi(IA, ML) = 1$ et $N(IA, ML) \geq 0.8$.

De plus, l'index donne $\Pi(ML, D) = 1$ et $N(ML, D) = 0.6$. Ceci nous donne les degrés en passant par *ML* : $\Pi'(IA, D) = 1$ et $N'(IA, D) = 0.6$. Puisque les deux valeurs (directement par *IA* ou en passant par *ML*) sont possibles, elles sont considérées comme disjointes (comme classiquement dans les systèmes de RI) et la valeur max entre $(1, 0.6)$ et $(1, 0.4)$ (cf. section 3.2) est prise en compte, et donc $\Pi(IA, D) = 1$ et $N(IA, D) = 0.6$.

Nous supposons ici que les documents plus spécifiques sont aussi pertinents. Si l'utilisateur veut seulement extraire les documents généraux, la désactivation de l'expansion vers la spécialisation doit être permise, au moins la pertinence des documents ainsi trouvés doit-elle être

atténuée. Notez que seul le degré issu de *ML* influence le résultat, comme si la requête avait été $BD \wedge ML$ (pour ce document en particulier), puisque *ML est en fait IA* ($N(IA,ML)=1$). De même, si la requête avait été *Informatique*, l'évaluation n'aurait pas été nulle grâce à l'expansion de la requête, contrairement à une évaluation classique. L'évaluation finale de la requête pour ce document sera : $\Pi(R, D) = \min(\Pi(BD, D), \Pi(IA, D)) = 1$ et $N(R, D) = \min(N(BD, D), N(IA, D)) = 0.2$. Les documents pourront être ordonnés comme en section 2. Cet exemple simple peut être étendu en pondérant les termes de la requête comme indiqué en section 2.

3.4 Identification d'expressions

Les concepts apparaissant dans les ontologies peuvent être des groupes de mots ou des expressions et doivent être identifiés comme tels dans le texte. Une méthode simple (mais naïve) pour identifier des expressions plutôt que des mots isolés est de regrouper les mots selon leur proximité et de faire correspondre ces groupes avec l'ontologie. La méthode classique serait d'utiliser une analyse de langage naturel, mais ce n'est pas le propos de cette partie. Considérons à titre d'exemple la base de données de titres d'articles suivante. Les

1	Dealing with vagueness of natural languages
2	Tolerant fuzzy pattern matching
3	A hierarchical model of fuzzy classes
4	Resolution principles in possibilistic logic
5	Weighted fuzzy pattern matching
6	Flexible queries to a crisp database

titres sont vus comme des listes (conjonctions) de termes significatifs, les considérations statistiques n'ayant pas de sens pour des textes si courts. L'utilisation du voisinage de chaque terme permet de résoudre le problème suivant, illustré par le titre 6. Supposons l'on cherche des articles sur les « fuzzy databases », et même si l'on sait que « flexible » et « fuzzy » sont souvent synonymes, c'est la requête (« query »)

qui est fbue dans ce titre, et non la base de données ! Pour traiter de tels cas, nous devons tenir compte de la proximité des termes dans le texte. Il y donc deux aspects de proximité à gérer : l'identification d'expressions et le contexte des mots pour la portée de la requête. Comme déjà mentionné, nous ne voulons pas traiter des aspects relatifs à l'analyse du langage naturel, mais seulement présenter une illustration supplémentaire des potentiels de notre approche.

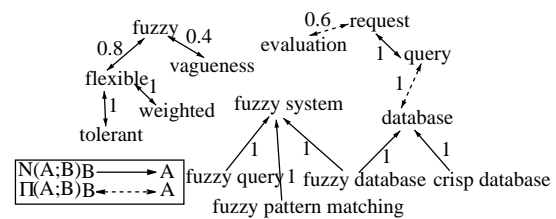


Figure 6 – Ontologies pour les titres d'articles

Considérons l'exemple de la base de données ci-dessus et l'ontologie de la figure 6 montrant un fragment d'une ontologie spécifique de mots-clés ou de sujets d'articles. Évaluons d'abord une requête simple : $R = fuzzy$. On a $N(fuzzy, vagueness) = 0.4$, donc $N(R, D_1) = 0.4$. Trivialement, $N(R, D_3) = 1$ et $N(R, D_4) = 0$. Comme $N(fuzzy, flexible) = 0.8$, l'article 6 est aussi retrouvé avec $N(R, D_6) = 0.8$. De même, $N(fuzzy, tolerant) = 0.8$, puisque « tolerant » et « flexible » sont presque synonymes. Par conséquent, $N(R, D_2) = \max(0.8, 1) = 1$, car D_2 contient les deux termes (idem pour D_5). Une requête plus complexe recherchant des articles traitant de « fuzzy request » est interprétée a priori comme $fuzzy \wedge request$ car l'expression *fuzzy request* n'est pas identifiée dans l'ontologie. Puisque $N(request, query) = 1$ et $N(fuzzy, flexible) = 0.8$, D_6 est le seul article satisfaisant avec $N = \min(0.8, 1) = 0.8$. Examinons maintenant la requête *fuzzy database*. Le terme *fuzzy database* existant dans l'ontologie, la requête est interprétée $R = fuzzy_database$ et non comme $fuzzy \wedge database$. De même, en regroupant les mots

du titre D_6 , le concept *crisp database* de l'ontologie est aussi identifié, ce qui conduit à $N(\textit{fuzzy_database}, \textit{crisp_database}) = 0$ et $\Pi(\textit{fuzzy_database}, \textit{crisp_database}) = 0$, et aucun résultat n'est obtenu. Si les mots de la requête n'avaient pas été regroupés, l'évaluation de *database* aurait donné D_6 comme résultat, puisque $\Pi(\textit{fuzzy_database}, \textit{database}) = 1$, ce qui ne satisfait pas la requête. Ainsi, regrouper les mots et utiliser une ontologie adaptée permet, dans une certaine mesure, de résoudre le problème de l'identification d'expressions dans un texte.

4 Remarques et conclusion

Cet article est préliminaire, et différents aspects doivent être développés : i) l'évaluation des degrés de nécessité et de possibilité, particulièrement dans les ontologies de chaque langue, ii) la gestion de l'importance des mots clés dans la requête, iii) la modélisation de la portée exacte de la requête pour exclure les documents trop généraux ou trop spécifiques. Notons enfin que l'usage d'ontologies possibilistes peut s'appliquer aussi bien à l'indexation « sémantique » de documents qu'à leur recherche. Un prototype a été implémenté afin de valider les exemples fournis dans cet article. Il tend à prouver la faisabilité de l'approche proposée, au moins dans le cadre de base de données avec des termes linguistiques. Cependant, pour l'application à la recherche documentaire, l'implémentation est encore trop préliminaire pour fournir des résultats vraiment significatifs sur des évaluations.

Remerciements :

Ce travail a été supporté par le projet européen E-Court (IST-2000-28199)

Références

- [1] T. Andreasen, H. Christiansen, and H. L. Larsen, editors. *Flexible Query Answering Systems*. Kluwer, 1997.
- [2] A. Bidault, C. Froidevaux, and B. Safar. Proximité entre requêtes dans un contexte médiateur. In *Actes RFIA 2002, Angers*, volume 2, pages 653–662, 2002.
- [3] D.A. Buell. An analysis of some fuzzy subset applications to information retrieval systems. *Fuzzy Sets and Systems*, 7(1) :35–42, 1982.
- [4] H. Bulskov, R. Knappe, and T. Andreasen. On measuring similarity for conceptual querying. In *Flexible Query Answering Systems, LNAI 2522*, pages 100–111. Springer, 2002.
- [5] M. Cayrol, H. Farreny, and H. Prade. Fuzzy pattern matching. *Kybernetes*, 11 :103–16, 1982.
- [6] D. Dubois and H. Prade. Resolution principles in possibilistic logic. *Int. Jour. of Approximate Reasoning*, 4(1) :1–21, 1990.
- [7] D. Dubois, H. Prade, and C. Testemale. Weighted fuzzy pattern matching. *Fuzzy Sets and Systems*, 28 :313–331, 1988.
- [8] D. Kraft, G. Bordogna, and G. Pasi. Fuzzy set techniques in information retrieval. In J. Bezdek et al., editor, *Fuzzy Sets in Approximate Reasoning and Information Systems*, chapter 8, pages 469–510. Kluwer, 1999.
- [9] Y. Loiseau and H. Prade. Qualitative pattern matching with linguistic terms. In T. Vidal and P. Liberatore, editors, *STAIRS 2002*, pages 125–134. Starting AI Researchers Symp., Lyon, IOS Press, July 2002.
- [10] H. Prade and C. Testemale. Application of possibility and necessity measures to documentary information retrieval. *LNCS*, 286 :265–275, 1987.
- [11] P. Resnik. Semantic similarity in a taxonomy : an information -based measure and its application to problem of ambiguity in natural language. *J. Artif. Intellig. Res.*, 11 :95–130, 1999.
- [12] J.P. Rossazza, D. Dubois, and H. Prade. A hierarchical model of fuzzy classes. In R. De Caluwe, editor, *Fuzzy and Uncertain Object-Oriented Databases*, pages 21–62. World Pub. Co., 1997.