De la modélisation sémantique des événements vers l'enrichissement et la recommandation

Houda Khrouf, Raphaël Troncy

Multimedia Communications Department, EURECOM
Campus SophiaTech, 06904 Biot Sophia Antipolis, France
{khrouf,troncy}@eurecom.fr

RÉSUMÉ. *De nombreux sites web ont récemment connu une croissance rapide fournissant des informations à propos d'événements passés ou à venir, et pour certains d'entre eux, accompagnés de photos et de vidéos capturées pendant ces événements. L'information disponible est, cependant, souvent incomplète, erronée et enfermée dans une multitude de sites web. Notre objectif est de fouiller en temps réel la connexion entre ces bases de données distribuées en utilisant les technologies du web sémantique. Cela permet d'assurer une meilleure complétude des données et de construire des vues riches sur les événements. En outre, des milliers d'événements sont créés chaque jour, ce qui nécessite un système de recommandation performant afin d'optimiser l'expérience utilisateur. Nous proposons donc une nouvelle approche de recommandation en exploitant la description sémantique des événements ainsi que la dimension sociale. Finalement, nous présentons une application web dans le but de répondre aux besoins d'utilisateur : revivre et découvrir des événements à partir de médias, et guider la prise de décision pour participer à des événements futurs.*

ABSTRACT. *Many sites have recently witnessed a rapid growth providing information about past and upcoming events, of which some may display media captured at these events. This information is, however, often incomplete and always locked into the sites. Our goal is to mine in real time the connection between these distributed datasets using semantic web technologies. This is a key advantage to ensure a better information completeness and to deliver enriched views of events. Moreover, thousands of events are daily created which require a powerful recommender system in order to enhance the user experience. We propose a new recommendation approach leveraging the semantic description of events as well as social information. Finally, we present a web application that aims to meet the user needs: relive and discover experiences based on media, and support decision making for attending upcoming events.*

MOTS-CLÉS : *LODE, EventMedia, réconciliation de données, web sémantique, recommandation.*

KEYWORDS: *LODE, EventMedia, instance matching, semantic web, recommendation.*

DOI:10.3166/RIA.28.321-347

1. Introduction

Avec le développement du web social, plusieurs référentiels d'événements et de média ont connu ces dernières années une croissance exponentielle. Considérant cette abondance d'information, nous avons effectué plusieurs études exploratoires pour mieux comprendre comment les utilisateurs découvraient et participaient à des événements ou partageaient leur expériences (Fialho *et al.*, 2010). A travers ces études, les participants ont reconnu leur besoin d'accéder à plusieurs sources d'informations pour avoir une image précise et le contexte d'un événement. Dans l'ensemble, les utilisateurs préconisent la nécessité d'une source unique pour explorer les événements, non pas en créant une nouvelle source d'information, mais en centralisant les données existantes tout en assurant une couverture plus large. Ces études soutiennent donc l'idée de développer une application web qui agrégerait des informations disponibles dans des annuaires d'événements avec des témoignages média capturés par les utilisateurs (Troncy, Fialho *et al.*, 2010). Néanmoins, cette agrégation ne devra pas induire une surchage d'information, ce qui nécessite des options de navigation et des mécanismes de recommandation qui répondent aux contraintes des utilisateurs.

Dans ce travail, notre postulat est que l'intégration à large échelle des sources de données hétérogènes peut être assurée par les technologies du web sémantique, unifiant ainsi l'information dans un environnement homogène. Comme les référentiels d'événements et de média sont en constante évolution, notre stratégie est de construire une architecture suffisamment flexible pour être en mesure d'ajouter facilement de nouvelles sources de données et de les interconnecter. Les technologies du web sémantique sont reconnues comme étant les plus adaptées pour obtenir une telle flexibilité à travers l'utilisation d'un modèle de données de type graphe basé sur RDF et la réutilisation d'ontologies existantes. En effet, le web de données [1] a comme double objectif de *i)* publier des descriptions représentées en RDF dont les URIs identifient des documents web, des objets du monde réel et des relations les reliant et *ii)* d'interconnecter ces jeux de données. A l'issue de processus sociaux, certains vocabulaires sont devenus très populaires facilitant ainsi l'interconnexion des données (Vatant, Rozat, 2011). Ainsi, on utilisera plutôt le vocabulaire `Dublin Core` [2] pour attacher un titre ou une description à une ressource, `FOAF` [3] pour décrire une personne ou un groupe et `WGS84` [4] pour représenter les lieux géographiques. Pourtant, nous observons qu'aucun vocabulaire n'a encore véritablement émergé pour représenter la notion d'événement.

Outre l'intégration de données, nous considérons aussi les mécanismes de recommandation pour fournir du contenu personnalisé et réduire ainsi la surcharge d'information. Des milliers d'événements sont partagés chaque jour, ce qui rend un système de recommandation essentiel pour décoder les intérêts des utilisateurs et optimiser l'information perçue. Dans ce contexte, les technologies du web sémantique

1. http://linkeddata.org/
2. http://purl.org/dc/elements/1.1
3. http://xmlns.com/foaf/0.1
4. http://www.w3.org/2003/01/geo/wgs84_pos

permettent une extraction simple et rapide des attributs d'entités qui seront analysés par un système de recommandation. La facilité de cette extraction par rapport aux outils traditionnels est essentiellement due à la structuration sémantique de l'information dans des ontologies. Dans ce travail, nous mettons en place une méthodologie pour exploiter pleinement cette structuration et proposer un nouveau mécanisme pour la recommandation d'événements.

Cet article est structuré de la façon suivante : nous décrivons tout d'abord notre ontologie pour représenter les événements, ainsi que notre jeu de données nommé EventMedia et composé d'une part de descriptions sémantiques d'événements, et d'autre part, de descriptions de photos et vidéos illustrant ceux-ci (section 2). Nous présentons ensuite notre approche pour interconnecter plusieurs jeux de données en temps réel (section 3). Nous décrivons comment exploiter le web sémantique dans un système de recommandation et nous proposons une nouvelle approche hybride pour recommander des événements (section 4). Deux applications sont finalement décrites pour permettre à un utilisateur de découvrir ou de revivre des événements à partir de médias (section 5).

2. Modélisation sémantique des événements

Le terme "événement" est polysémique : il fait tout à la fois référence à des phénomènes passés (décrits dans des articles de presse ou expliqués par des historiens) et à des phénomènes planifiés dans le futur (notés dans un calendrier ou une programmation). Dans des travaux précédents, nous avons analysé les différentes ontologies permettant de représenter la notion d'événement. Nous avons alors proposé l'ontologie LODE qui fourni un modèle simple pour représenter les différentes propriétés composant un événement ainsi qu'un ensemble de correspondances entre de nombreux modèles pour le représenter (Troncy, Shaw, Hardman, 2010).

Dans cette section, nous présentons d'abord l'ontologie LODE à l'aide d'un exemple (section 2.1). Nous décrivons ensuite la fabrication du jeu de données EventMedia qui a fait son apparition dans le nuage de données du web sémantique (section 2.2).

2.1. L'ontologie LODE

LODE [5] est une ontologie minimale permettant la description interopérable des aspects "factuels" d'un événement, ce qui peut se caractériser en terme des "quatre Ws"(*what*, *when*, *where*, *who*) : qu'est-ce qui s'est passé, où et quand cela s'est-il produit, qui était impliqué. Ces relations factuelles décrivant un événement ont comme objectif de représenter une réalité consensuelle et ne doivent donc pas être associées à une perspective ou une interprétation particulière. LODE fournit un ensemble d'axiomes logiques entre de nombreuses classes et propriétés définies dans

5. http://linkedevents.org/ontology/

d'autres modèles tels que les ontologies Event, CIDOC-CRM, DOLCE, SEM (Hage *et al.*, 2009) pour n'en citer que quelques unes. Le tableau 1 montre quelques unes des propriétés du modèle LODE avec leurs correspondances.

Tableau 1. Exemple d'alignements entre les propriétés de plusieurs ontologies événements

ABC	CIDOC	DUL	EO	LODE
atTime	P4.has_time-span	isObservableAt	time	atTime
	P7.took_place_at		place	inSpace
inPlace		hasLocation		atPlace
involves	P12.occurred_in_-the_presence_of	hasParticipant	factor	involved
hasPresence	P11.had_-participant	involvesAgent	agent	involvedAgent

La figure 1 illustre comment l'événement identifié par `350591` sur Last.fm serait décrit avec l'ontologie LODE. Plus précisément, elle montre qu'un événement de type `Concert` a été donné le `13 juillet 2007 à 20h30` au théâtre le `Nouveau Casino` à Paris avec comme vedette la chanteuse irlandaise `Róisín Murphy` connue pour sa musique électronique.

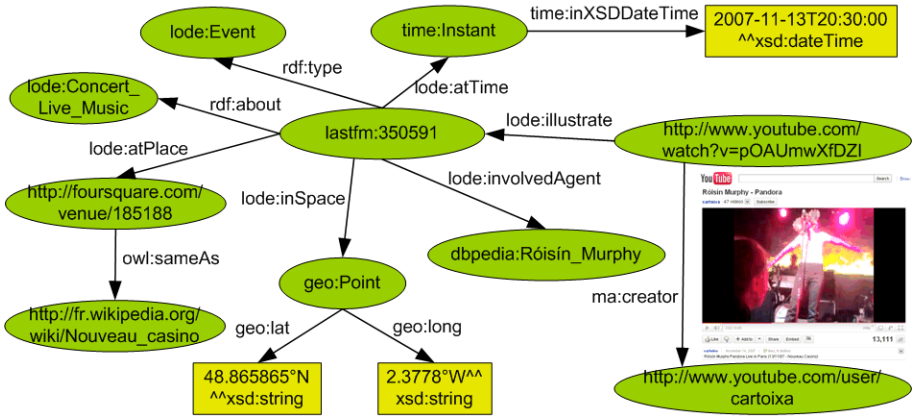

Figure 1. Róisín Murphy au Nouveau Casino à Paris *décrit avec LODE*

2.2. EventMedia = Last.fm + Eventful + Upcoming + Flickr

EventMedia est une bulle [6] du nuage de données (LOD Cloud) apparue dans sa représentation imagée (Cyganiak, Jentzsch, 2010). EventMedia est composée de descriptions d'événements publiés sur Last.fm, Eventful et Upcoming qui ont au moins une photo publiée sur Flickr explicitement associée à ces événements.

6. Voir aussi la description sur CKAN à http://datahub.io/dataset/event-media

Nous utilisons les APIs de ces trois annuaires d'événements pour convertir les descriptions selon l'ontologie LODE. Nous créons nos propres URIs dans notre espace de noms pour représenter les événements (http://data.linkedevents.org/event), les agents (http://data.linkedevents.org/agent) et les lieux (http://data.linkedevents.org/location). Un graphe représentant un événement est ainsi composé du type de l'événement, d'une description textuelle, des personnes impliquées, d'une date (un instant ou un intervalle représenté avec l'ontologie OWL Time (Hobbs, Pan, 2006)), d'un lieu représenté à la fois en termes de coordonnées géographiques et d'une étiquette. Un graphe représentant un agent ou un lieu contient une étiquette et une description textuelle (*e.g.* la biographie d'un artiste), le lieu ayant en plus une adresse structurée.

Une relation explicite entre un événement publié dans un annuaire et une photo hébergée sur Flickr peut être retrouvée à l'aide des tags sémantiques spéciaux tels que `lastfm:event=XXX` ou `upcoming:event=XXX`. Dans un travail précédent, nous avons collecté l'intersection des sites Last.fm, Eventful et Upcoming avec Flickr pour obtenir un jeu de données composé de plus de 140 000 descriptions d'événements associés à plus de 1,7 million de photos (tableau 2).

Tableau 2. Volume de descriptions pour les classes event/agent/location et photo/user dans le jeu de données EventMedia

	Event	Agent	Location	Photos	User
Last.fm	37,647	50,151	16,471	1,393,039	18,542
Upcoming	13,114	-	7,330	347,959	4,518
Eventful	37,647	6,543	14,576	52	12

EventMedia contient plus de 30 millions de triplets RDF accessible en utilisant des requêtes SPARQL [7]. Une API [8] REST a été mise en place en utilisant l'implémentation ELDA de la Linked Data API [9]. ELDA permet d'accéder aux données RDF en utilisant des URL REST simples qui sont traduits en requêtes SPARQL.

Les événements sont généralement catégorisés en taxonomies qui fournissent, sur de nombreux sites, un moyen pratique de parcourir les événements publiés par type. Nous avons manuellement analysé les taxonomies proposées par différents sites tels que Facebook, Eventful, Upcoming, Zevents, LinkedIn, EventBrite, TicketMaster ainsi que les jeux de données encyclopédiques du nuage de données. Nous avons alors appliqué la technique du tri par cartes [10] pour construire un thésaurus de catégories d'événements contenant des renvois à ces sources. Le thésaurus est représenté en SKOS et les termes sont définis dans notre espace de noms à (http://data.linkedevents.org/category/).

7. http://eventmedia.eurecom.fr/sparql
8. http://eventmedia.eurecom.fr/rest/{resource}
9. http://code.google.com/p/linked-data-api
10. http://fr.wikipedia.org/wiki/Tri_par_cartes

3. Enrichissement et interconnexion des données

L'enrichissement sémantique des descriptions a été perçu par les utilisateurs comme un moyen de répondre au problème de la qualité et de la complétude des données que l'on peut trouver dans les annuaires d'événements. D'ailleurs, plusieurs bulles du nuage de données contiennent des informations liées à des événements, des personnes ou des lieux. Les référentiels constituant EventMedia représentent aussi un chevauchement non négligeable qui peut être exploité en vue d'enrichissement. Nous avons donc cherché à interconnecter à large échelle ces jeux de données en développant un framework de réconciliation en temps réel. Cela a pour objectif d'aligner les flux entrants des données afin de soutenir un enrichissement continu et faire face à l'évolution constante des jeux de données. Le gain majeur de cette réconciliation est de rassembler les avantages de chaque service afin de fournir une meilleure vue d'ensemble d'un événement. Par exemple, la description des artistes participant à un événement dans Upcoming est souvent inexistante, ce qui peut être compensé en utilisant l'événement similaire sur Last.fm.

Dans cette section, nous commençons par décrire notre méthodologie pour aligner en temps réel les instances d'EventMedia en se basant sur des mesures de corrélation et de couverture (section 3.1). Nous présentons ensuite les résultats obtenus (section 3.2).

3.1. Réconciliation basée sur la corrélation des prédicats

L'alignement des instances est une tâche d'une importance capitale dans le web sémantique. Il vise à créer des liens exprimant une relation de similarité `owl:sameAs` entre deux instances. Notre objectif est d'aligner des événements, des personnes et des lieux de différents jeux de données qui sont représentés par des vocabulaires variés. Afin de pallier à cette diversité des modèles de représentations, il y a un besoin d'une approche d'alignement indépendante du domaine des instances. En outre, il est probable de rencontrer dans les données des erreurs typographiques ou des attributs différents pour décrire deux instances similaires. Ainsi, nous avons remarqué que certaines propriétés sémantiquement différentes peuvent avoir une relation "latente" entre eux. Par exemple, on trouve qu'il y a un événement de Last.fm dont l'attribut de `dc:title` est *"Cale Parks at Pehrspace"*, tandis que l'attribut du même événement dans Upcoming est *"Cale Parks, The Flying Tourbillon Orchestra, One Trick Pony, Meredith Meyer"* qui énumère les artistes plutôt exprimés par le prédicat `lode:involvedAgent` dans Last.fm. Ce type d'hétérogénéité a été rarement pris en compte dans les outils existants qui se basent en général sur une configuration manuelle des prédicats à comparer ou sur une comparaison des prédicats ayant une sémantique similaire tels que `dc:title` et `rdfs:label`.

Parmi les outils d'alignement, on trouve SILK (Jentzsch *et al.*, 2010) qui se base sur un langage déclaratif de spécification des liens (Silk-LSL) où l'utilisateur doit définir manuellement les propriétés à comparer. Cette méthode est efficace pour des jeux de données simples, mais elle ne permet pas de détecter les similitudes latentes qui peuvent exister entre deux propriétés sémantiquement différentes. Zhisi.Links (Niu *et*

al., 2011) est un autre outil d'alignement qui fonctionne en deux étapes. La première étape permet de sélectionner les candidats potentiellement similaires en utilisant les étiquettes des instances. Ensuite, la relation de similarit'e est déterminée par une métrique de similarité sémantique. Ce système est performant, mais uniquement efficace lorsqu'il existe une similarité exacte entre deux étiquettes (*e.g.* deux événements ont le même titre). Pour résoudre ce problème, Song et Heflin proposent une technique d'apprentissage non supervisé en se basant sur le pouvoir discriminant et la couverture des propriétés afin de détecter les clefs (prédicats) appropriées pour la sélection des candidats (Song, Heflin, 2011). Cependant, cette technique est biaisée et favorise les littéraux de chaîne, ne prenant ainsi pas en compte les autres types de données tels que les valeurs numériques, temporelles et spatiales. Une autre approche non supervisé a été proposée par (Nikolov *et al*., 2012) dans l'outil KnoFuss qui exploite la programmation génétique pour détecter les différents paramètres de la fonction de similarité qu'on va l'utiliser dans notre expérimentation.

Dans ce travail, nous considérons les différents types de données et nous proposons une technique supervisée basée sur la corrélation et la couverture des prédicats. Comme Zhisi.Links (Niu *et al*., 2011) et (Song, Heflin, 2011), notre approche comprend deux étapes : (1) détecte les prédicats clefs pour la sélection des candidats ; (2) utilise une méthode d'optimisation pour déduire la fonction de similarité qui maximise le F-score (une mesure populaire combinant la précision et le rappel). Pour cette approche, nous prenons en compte tous les types de données tels que les chaînes littérales, les valeurs numériques ou temporelles comme décrit dans ce qui suit.

Littéral de chaîne. Pour les chaînes longues (*e.g.* description), nous utilisons l'algorithme Porter pour la racinisation et la métrique *Cosinus* pour calculer la similarité. Pour les chaînes courtes (*e.g.* étiquettes), nous utilisons la métrique hybride *Token-Wise* décrite par la formule suivante (Khrouf, Troncy, 2011a) :

$$\textit{Token-Wise}(S,T) = \frac{\sum_{s\in S, t\in T}\ Levenshtein(s,t)}{max(|S|,|T|)} \tag{1}$$

où S et T sont les ensembles des jetons formant les chaines à comparer (tokenization).

Littéral temporel. Dans EventMedia, la dimension temporelle d'un événement peut être représentée par un point ou par un intervalle. Dans ce cas, la métrique temporelle, qui existe dans les outils d'alignement, mesurant la distance entre deux points est insuffisante. Il y a aussi un besoin de mesurer l'inclusion temporelle entre un point et un intervalle, et le chevauchement temporel entre deux intervalles.

On considère deux événements (e_1, e_2) qui ont respectivement les couples (d_1, d'_1) et (d_2, d'_2) (où d est la date de début et d' est la date de fin d'un événement). La métrique temporelle est représentée par la formule suivante :

$$s\ (e_1,e_2) = \begin{cases} 1 & \text{if } |d_1 - d_2| \leq \theta \text{ où } (d'_1, d'_2) = 0 \\ 1 & \text{if } d_1 \pm \theta \in [d'_1, d'_2] \text{ où } d_2 = 0 \text{ (idem pour } d'_1) \\ 1 & \text{if } min(d'_1, d'_2) - max(d_1, d_2) \geq 0 \text{ où } (d'_1, d'_2) \neq 0 \\ 0 & \text{sinon.} \end{cases} \tag{2}$$

Littéral numérique. On calcule simplement l'inverse de la valeur absolue de la différence entre deux valeurs numériques.

Pour faire face à l'hétérogénéité des données, nous avons besoin tout d'abord de déceler les prédicats qui doivent être comparés. Pour cela, nous nous basons sur la corrélation et la couverture des prédicats mesurées à partir d'une vérité terrain. Nous prenons comme exemple deux ensembles d'instances appariées I_s (source) et I_c (cible). Pour chaque ensemble I_i ($i \in \{s, c\}$), nous formons l'ensemble des littéraux L_i associé à chaque propriété p_i. Si une propriété est utilisée plus d'une fois, nous regroupons les valeurs multiples associées en une seule valeur. La corrélation correspond à l'information mutuelle partagée entre deux propriétés issues des ensembles I_s et I_c. Chaque type de données dans L_i est associé à une fonction de similarité sim_d comme décrit ci-dessus. Nous formalisons la corrélation et la couverture de chaque paire de propriétés comme suit:

$$Correlation(p_s, p_c) = \frac{\sum_{l_s \in L_s, l_c \in L_c} sim_d(l_s, l_c)}{min(|L_s|, |L_c|)} \quad (3)$$

$$Couverture(p_s, p_c) = \frac{min(|L_s|, |L_c|}{|I_s|} \quad (4)$$

Avec ces deux mesures, on peut considérer que les prédicats clefs pour la sélection des candidats sont associés avec des valeurs maximales de corrélation et de couverture. Ensuite, les autres prédicats restants sont utilisés pour calculer le score total de similarité. Pour pondérer la contribution de ces prédicats dans le calcul de similarité ainsi que le seuil de similarité, on utilise une méthode d'optimisation par essaims particulaires (PSO en anglais) (Kennedy, Eberhart, 1995). Cette technique stochastique s'inspire du comportement de groupe dans le monde du vivant tels que les oiseaux et les poissons. Elle initialise une population de solutions aléatoires appelées particules qui sont mis à jour à chaque génération (itération) en vue d'optimiser une fonction prédéfinie. Dans notre approche, une particule est représentée par un vecteur de pondérations et de seuils, et la fonction à optimiser est représentée par le F-score.

3.2. Évaluation

Dans cette section, nous évaluons notre approche pour l'interconnexion des événements, des agents et des lieux en se référant aux deux vérité terrain. Les paramètres de l'optimisation par essaims particulaires définis pendant l'expérimentation sont taille = 25, itérations = 40, coefficients d'accélération c_1 = 1,494 et c_2 = 1,494, et poids d'inertie w = 0,729 (valeurs de c_1,c_2,w recommandées dans la littérature (Eberhart, Shi, 2001)). Les résultats sur le nombre de liens générés entre les jeux de données sont accessibles en ligne [11].

11. http://eventmedia.eurecom.fr/dashboard/statistics.html

3.2.1. Alignement des événements

Intuitivement, la similarité entre les événements dépend de ses propriétés "factuelles", à savoir : le titre (what), la date (when), le lieu (where) et les agents (who). Cependant, la corrélation et la couverture de ces propriétés varient d'un jeu de données à un autre. Dans cette section, on se focalise sur l'évaluation des événements alignés entre Last.fm et Upcoming en utilisant un gold standard de 300 paires appariées qui a été construit manuellement. Le tableau 3 montre les coefficients de corrélation et de couverture obtenus dans l'ordre descendant (corrélation $\geqslant$ 0,3).

Tableau 3. Corrélation et couverture entre les propriétés des événements de Last.fm (source) et Upcoming (cible)

P_{source}	P_{cible}	$Correlation$	$Couverture$
$date_s$	$date_c$	**1**	**1**
$lieu_s$	$lieu_c$	**0,80**	**1**
$titre_s$	$titre_c$	0,59	**1**
$agent_s$	$titre_c$	0,53	**1**
$(lat_s, long_s)$	$(lat_c, long_c)$	(0,43, 0,97)	0,92

D'après le tableau 3, les propriétés $date$ et $lieu$ ont des valeurs maximales de couverture et de corrélation. Cette dimension spatio-temporel peut donc être considérée comme un prédicat clef pour la sélection des candidats. Pour chaque instance dans I_s, on extrait les candidats de I_c qui sont associés à un score moyen de similarité entre les propriétés $date$ et $lieu$, supérieur à un seuil estimé.

Pour l'évaluation, nous avons mis en place quelques expérimentations. D'abord, on évalue l'approche basée sur une combinaison linéaire (LC) des scores de similarité entre toutes les propriétés sans un mécanisme de sélection de candidats. Ensuite, on évalue deux méthodes intégrant la sélection des candidats : (i) la première méthode (Two-step LC) est une combinaison linéaire des scores de similarité entre les propriétés (excepté le prédicat clef) ; (ii) la deuxième méthode (Two-step OR) est basée sur un raisonnement booléen où il suffit que l'un de ces scores soit supérieur à un seuil. On utilise la technique d'optimisation PSO pour déduire le poids de score de similarité pour chaque propriété dans les méthodes LC et Two-step LC. On utilise cette méthode également pour déduire le seuil de similarité pour chaque propriété dans la méthode Two-step OR. Nous avons choisi de comparer ces méthodes avec KnoFuss (Nikolov *et al.*, 2012) qui se base sur l'algorithme génétique (GA) pour déduire les éléments d'une meilleure fonction de similarité. Ces éléments comprennent les paires de propriétés, les métriques, les poids et le seuil. Pour cela, nous avons intégré la métrique *Token-Wise* et la métrique temporelle dans Knofuss. Le tableau 4 résume les résultats obtenus.

On peut voir que Knofuss produit des appariements avec une bonne précision mais un mauvais rappel. Cela est dû à sa stratégie pour maximiser la précision en mode non supervisé étant donné qu'un appariement erroné est moins tolérable qu'un appariement manqué. Les résultats montrent aussi que les méthodes intégrant la sélection des

Tableau 4. Résultats de différentes approches d'alignement d'événements entre Last.fm et Upcoming (avec 50 % de données d'apprentissage)

	Precision	*Recall*	*F-score*
LC KnoFuss (GA)	0,94	0,74	0,83
LC (PSO)	0,88	0,96	0,92
Two-step LC (PSO)	0,91	0,95	0,93
Two-step OR (PSO)	**0,96**	**0,97**	**0,96**

candidats ont la meilleure performance grâce au filtrage et à la réduction du bruit. En particulier, l'approche (Two-step OR) basée sur le raisonnement booléen a réussi à pallier le manque de couverture des prédicats géographiques (lat et long). En effet, le poids attribué à la distance géographique est très faible dans les méthodes à combinaison linéaire, tandis qu'un poids élevé a été attribué par la méthode au raisonnement booléen. Finalement, on évalue cette méthode (Two-step OR) en variant la taille des données d'apprentissage (tableau 5). On a pu obtenir une bonne performance même pour des données d'apprentissage de petite taille.

Tableau 5. Résultats de la méthode d'alignement au raisonnement booléen (Two-step OR) pour différentes tailles de données d'apprentissage

	Precision	*Recall*	*F-score*
30 %	0,95	0,96	0,95
50 %	0,96	0,97	0,96
80 %	**0,99**	**0,98**	**0,99**

Pour l'alignement des événements, nous étudions également la connexion entre EventMedia et DBpedia. Nous observons que les deux ensembles de données contiennent des descriptions d'événements différentes en termes de modèle de données et de granularité des données. En effet, EventMedia fournit des informations à granularité fine détaillant la dimension spatio-temporelle ainsi que d'autres propriétés. En revanche, DBpedia fournit une description générale des événements populaires sans précision fine sur la date et le lieu, à l'exception de quelques événements. Considérant ce fait, nous avons décidé de créer un lien `rdfs:seeAlso` entre les événements de ces jeux de données. Pour ce faire, nous comparons simplement les étiquettes d'événements en fixant un seuil élevé pour éviter un appariement erroné.

3.2.2. Alignement des agents

Les jeux de données pertinents pour aligner les personnes sont : DBpedia générés à partir de Wikipedia ainsi que MusicBrainz [12] et BBC [13] qui contiennent une grande base de données d'artistes et d'albums. Pour aligner les artistes, nous considérons leurs noms comme un prédicat clef pour la sélection des candidats. Cependant, un conflit

12. http://musicbrainz.org
13. http://www.bbc.co.uk

des noms peut se produire, ce qui nécessite d'impliquer d'autres attributs comme la biographie. Pour l'évaluation, on se focalise sur l'interconnexion entre Last.fm et DBpedia. Le tableau 6 montre la corrélation et la couverture des propriétés à partir d'un échantillon de 100 appariements. Les résultats mettent en évidence une corrélation moyenne pour les étiquettes des agents, ce qui confirme l'importance d'intégrer d'autres propriétés dans le calcul de similarité.

Tableau 6. Corrélation et couverture des propriétés des agents de Last.fm et DBpedia

P_{source}	P_{cible}	$Correlation$	$Coverage$
$tiquettes$	$tiquette_c$	0,69	1
$subject_s$	$genre_c$	0,52	0,90
$description_s$	$comment_c$	0,35	0,98

En utilisant un gold standard de 2000 appariements entre Last.fm et DBpedia qui a éé de nouveau construit manuellement, la méthode à base de raisonnement booléen (Two-step OR) a une meilleure performance (voir tableau7).

Tableau 7. Résultats de différentes approches d'alignement des agents entre Last.fm et DBpedia

	$Precision$	$Recall$	$F\text{-}score$
LC (PSO)	0,97	0,95	0,96
Two-step LC (PSO)	**0,99**	0,95	0,96
Two-step OR (PSO)	**0,99**	**0,98**	**0,98**

3.2.3. Alignement des lieux

L'alignement des lieux a été particulièrement simple grâce à la description cohérente et complète des lieux dans plusieurs jeux de données. Cette description inclut plusieurs attributs comme l'adresse, les coordonnées géographiques, le code postal et le pays. Les jeux de données pertinents pour aligner les lieux sont : Foursquare [14] et NokiaMaps [15] qui fournissent une gigantesque base de données de points d'intérêts, et DBpedia généré à partir de Wikipedia. Nous n'avons pas créé de gold standard pour évaluer cet alignement. En revanche, on n'a jamais détecté un appariement erroné, et on a observé qu'il y a un nombre considérable de liens créés notamment avec le référentiel Foursquare.

3.3. Alignement temps réel

Pour pouvoir exécuter un alignement en utilisant de simples requêtes HTTP, nous avons développé un service REST qui permet d'aligner les données fraichement stockées dans le serveur. Le service envoie deux types de requêtes SPARQL. La première

14. http://foursquare.com
15. http://here.com

requête extrait les instances du jeu de données source en les filtrant par la propriété `rdf:type` et la date de stockage (représentée par `dc:issued`). La deuxième requête extrait, pour chaque instance source, les candidats cible en utilisant `rdf:type` et un prédicat clef. Pour garantir un alignement en temps réel, nous avons créé un programme qui exécute deux tâches successives toutes les 10 minutes : (i) La première tâche collecte les dernières photos téléchargées dans Flickr en utilisant son service RSS (taille de 20 photos). Les descriptions des événements, des agents et des lieux correspondants à ces photos sont ensuite chargées dans le triple store. (ii) La deuxième tâche aligne ces données stockées avec d'autres jeux de données en utilisant des requêtes HTTP envoyées au framework de réconciliation. Pour évaluer un scénario en temps réel, nous considérons un échantillon de données collecté durant 3 jours, et nous mesurons les intervalles suivants : (i) L'intervalle de stockage qui représente la différence entre les dates de stockage des photos dans Flickr et dans notre triple store ; (ii) L'intervalle de réconciliation qui représente la différence entre la date de stockage d'une instance et la date de son alignement.

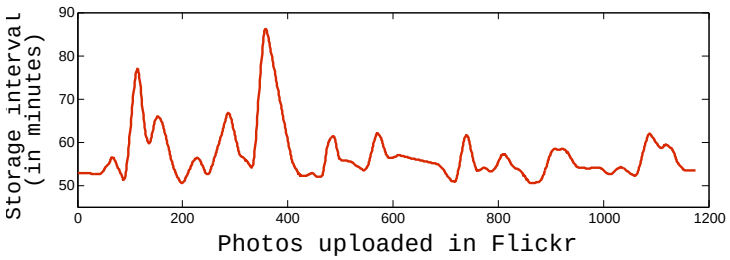

Figure 2. Mesure de l'intervalle de stockage

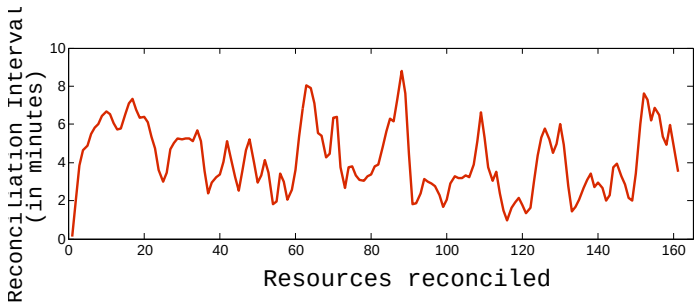

Figure 3. Mesure de l'intervalle de réconciliation

Le temps nécessaire pour exécuter la première tâche (collection des données) varie de quelques secondes à 3 minutes. Cette durée est affectée par le nombre d'entités liées aux événements telles que les artistes et les participants, qui nécessitent chacun une requête vers l'API approprié. Dans la figure 2, nous observons que l'intervalle de stockage varie de 50 à 90 minutes, ce qui confirme que notre système héberge les photos récemment téléchargées dans Flickr. Nous notons que cette variation est corrélée avec le délai entre le téléchargement de photos et les mises à jour RSS de

Flickr. Le temps nécessaire pour exécuter la deuxième tâche (réconciliation) varie de quelques secondes à 6 minutes selon le nombre d'entités à aligner. La figure 3 montre un intervalle court entre les dates de stockage et de l'alignement, ce qui atteste de l'efficacité de notre stratégie de réconciliation en temps réel.

4. Approche sémantique pour la recommandation d'événements

La recommandation a pour objectif de réduire la surcharge d'information et de guider l'utilisateur à prendre une décision qui correspond à ses intérêts. Dans un service qui fournit des milliers d'événements par jour, les options de navigation deviennent insuffisantes et un moteur de recommandation s'avère indispensable pour optimiser l'expérience utilisateur. En particulier, la recommandation d'un événement met en jeu plusieurs facteurs comme le temps, le lieu, la popularité des artistes et la participation d'"amis". Cette pluralité rend inefficace les systèmes de recommandation existants tels que ceux basés sur le contenu ou le filtrage collaboratif. Nous proposons donc un système hybride qui combine ces deux approches tout en exploitant les technologies du web sémantique. Dans ce qui suit, nous expliquons comment construire un tel système de recommandation en utilisant le web des données (section 4.1), et ensuite, nous décrivons notre approche hybride (section 4.3).

4.1. Recommandation thématique dans le web des données

La recommandation basée sur le contenu ou la recommandation thématique s'appuie sur le contenu des objets pour proposer des profils similaires à ceux qui ont été précédemment appréciés par l'utilisateur (Pazzani, Billsus, 2007). Le système compare le profil d'un objet (composé de descripteurs) avec d'autres profils intéressants afin de prédire l'opinion de l'utilisateur sur cet objet. Cette comparaison consiste à calculer la similarité entre les profils des objets, ce qui peut être mesuré par le biais de plusieurs métriques comme la similarité de Pearson ou la similarité Cosinus. Pour représenter le profil d'un objet, la méthode la plus commune est la représentation des métadonnées en utilisant TF-IDF (*Term Frequency-Inverse Document Frequency*) (Généreux, Santini, 2007) qui permet d'évaluer un mot-clé par sa fréquence dans un document et par sa présence dans tous les autres documents du corpus. Une telle représentation nécessite des techniques d'extraction des caractéristiques pour transformer une description non structurée en une forme structurée soulignant les métadonnées. Cette extraction devient extrêmement simple avec les technologies de web sémantique grâce à la structuration des données dans des ontologies. En outre, le modèle sémantique permet d'enrichir la description d'un objet avec des informations supplémentaires en exploitant la richesse du web des données. Nous expliquons par la suite l'approche proposée pour appliquer la recommandation thématique dans un espace sémantique.

Tout d'abord, on a décidé d'adopter la méthode proposée par (Di Noia *et al.*, 2012) pour calculer la similarité entre les objets dans le web des données. Cette méthode considère intuitivement que deux ressources dans un graphe RDF sont simi-

laires si elles sont le sujet de deux triplets ayant le même prédicat et le même objet (où un triplet = <sujet,prédicat,objet>). Elle est basée sur le modèle d'espace vectoriel (VSM) (Salton *et al.*, 1975), une technique bien connue dans le domaine de la recherche d'information. L'application de ce modèle sur un graphe RDF projette le web des données dans un espace tensoriel, où chaque tranche du tenseur représente une matrice d'adjacence pour chaque propriété de l'ontologie (voir l'exemple de la figure 4). En fait, le graphe RDF peut être défini comme un graphe $G = (R, P)$, où R est l'ensemble des ressources, et P est l'ensemble des prédicats entres les ressources. Pour chaque prédicat dans P, la matrice d'adjacence représente les liaisons entre les sujets (lignes) et les objets (colonnes) de R.

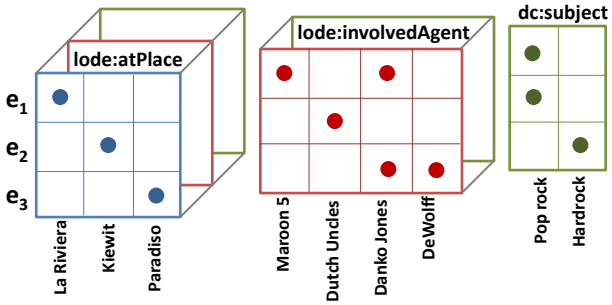

Figure 4. Un exemple de tenseur pour des propriétés liées à un événement

Cette approche est applicable dans le cas où les prédicats sont sémantiquement indépendants, ce qui est le cas pour notre modèle d'événement. Pour chaque objet o lié à l'événement e_i suivant le prédicat p, le poids TF-IDF est :

$$w_{o,i,p} = f_{o,i,p} \ * \ log \left(\frac{N}{m_{o,p}} \right) \tag{5}$$

où $f_{o,i,p} = 1$ si un lien existe entre l'événement e_i et l'objet o via le prédicat p, sinon $f_{r,i,p} = 0$, N est le nombre total des événements, $m_{o,p}$ est le nombre des événements liés à l'objet o via le prédicat p. Soit t le nombre total des objets, la similarité entre deux événements e_i et e_j suivant le prédicat p est calculée à l'aide de la métrique Cosinus :

$$sim^p(e_i, e_j) = \frac{\sum_{r=1}^{t} w_{o,i,p} * w_{o,j,p}}{\sqrt{\sum_{o=1}^{t} w_{o,i,p}^2} * \sqrt{\sum_{o=1}^{t} w_{o,j,p}^2}} \tag{6}$$

L'approche décrite ci-dessus peut être appliquée pour calculer d'une façon simple les similitudes entre les sujets ou les objets de triplets RDF. Il a été utilisé avec succès pour recommander des films et améliorer la performance d'un système basé sur le contenu (Di Noia *et al.*, 2012). Néanmoins, cette approche est limitée lorsque la matrice d'adjacence est creuse, comme c'est le cas pour les matrices associées aux prédicats `lode:atPlace` et `lode:involvedAgent`. En fait, ces prédicats sont considérés comme des propriétés discriminantes en RDF caractérisées par la diversité importante de leurs objets. Par exemple, le vecteur représenté par le prédicat

`lode:atPlace` a une seule valeur non nulle puisqu'un événement ne pourra être associé qu'avec un seul lieu.

4.2. Problème du pouvoir discriminant des prédicats

Afin de réduire les cases vides des matrices d'adjacence associées aux prédicats discriminants, nous avons décidé d'interpoler des valeurs fictives en se basant sur la similarité entre les objets. Tout d'abord, nous définissons une métrique pour mesurer le pouvoir discriminant d'un prédicat (Song, Heflin, 2011) comme suit :

$$Discriminability(p) = \frac{\mid \{o \mid t =< s, p, o > \in G\} \mid}{\mid t =< s, p, o > \in G \mid} \quad (7)$$

où G est un graphe RDF, t est un triplet représentant le lien entre le sujet s et l'objet o via le prédicat p. Cette métrique reflète le fait qu'un prédicat est lié à plusieurs objets différents. Par exemple, dans un échantillon de 1700 événements (liés à 10,323 agents et 627 lieux), on trouve un pouvoir discriminant assez élevé pour les prédicats `lode:involvedAgent` (de l'ordre de 64 %) et `lode:atPlace` (de l'ordre de 45 %).

Pour résoudre ce problème, on propose d'interpoler les valeurs de similitudes entre les objets dans la matrice d'adjacence de la manière suivante : si un objet o_k est similaire à un objet o_h et si $f_{o_h,i,p} = 1$ et $f_{o_k,i,p} = 0$, alors $f_{o_k,i,p} = sim(o_k, o_h)$. Si o_k est similaire à plusieurs objets liés à l'événement e_i via le prédicat p, le poids $f_{o_k,i,p}$ est égal au score de similarité maximal. Finalement, pour chaque objet o_k, l'équation 5 devient :

$$w_{o_k,i,p} = \max_{o_h \in H} sim(o_k, o_h) \ * \ log \left(\frac{N}{m_{o_k,p}} \right) \quad (8)$$

où H est l'ensemble des objets qui sont déjà liés à l'événement e_i via le prédicat p. Pour calculer la similarité entre les objets, plusieurs techniques peuvent être utilisées et cela dépend de la nature de l'objet lui-même. Dans notre cas, on a utilisé les valeurs de similarité entre les artistes fournies par les services sociaux comme Last.fm, et on a calculé la distance géographique normalisée entre deux lieux. Avec l'interpolation de ces similitudes, on a réussi à diviser le nombre de cases vides des matrices d'adjacence par 3. Finalement, la similarité entre deux événements e_i et e_j est une combinaison linéaire normalisée de toutes les valeurs de similarité associées à chaque prédicat.

4.3. Recommandation d'événements

La recommandation des événements est particulièrement difficile vu qu'elle doit faire face à des entités éphémères qui deviennent inutiles après un certain temps. En fait, un événement est différent d'un objet classique largement utilisé dans les systèmes de recommandation comme les films, musiques, livres, etc. Ces objets reçoivent en permanence des appréciations d'utilisateurs qui sont regroupées dans "une matrice d'usages". Dans notre cas, cette matrice reflète la participation des utilisateurs (lignes)

aux événements (colonnes). Elle est largement creuse de l'ordre de 98 % (la majorité des entrées sont nulles) étant donné qu'un événement est associé qu'avec un nombre très limité d'utilisateurs. Une solution connue pour pallier au manque de données d'usage est d'intégrer l'approche basée sur le contenu qui permet de comparer les métadonnées des objets avec le profil d'utilisateur. Cependant, cette approche ne prend pas en compte la dimension sociale qui répond à la question "quels sont mes amis qui vont assister à un événement donné ?". Pour apporter une réponse, la recommandation basée sur le filtrage collaboratif permet de mettre en avant la dimension sociale. Nous proposons ainsi une approche hybride qui combine un système basé sur le contenu et le filtrage collaboratif.

4.3.1. Approche basée sur le contenu

Le principe du système basé sur le contenu est de recommander un événement futur dont la description est similaire aux événements passés pour un profil d'utilisateur. Nous supposons qu'il existe un nombre suffisant d'événements passés dans le profil utilisateur afin d'éviter le problème de phase de démarrage à froid qui concerne un nouveau produit ou un nouvel utilisateur. Pour prédire la participation d'un utilisateur u à un événement e_i, on utilise les similitudes entre les événements de la manière suivante :

$$rank_{cb}(u, e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \, sim^p(e_i, e_j)}{\mid P \mid \, * \mid E_u \mid} \tag{9}$$

où P est l'ensemble des prédicats partagés entre les événements e_i et e_j, E_u est l'ensemble des événements passés dans le profil d'utilisateur u, et α_p est le poids attribué au prédicat p reflétant sa contribution dans la recommandation. Nous utiliserons par la suite des méthodes d'optimisation pour estimer les valeurs de α_p.

Suivant notre modèle RDF, les prédicats utilisés pour calculer la similarité entre les événements sont ceux qui sont liés aux lieux (`lode:atPlace`), artistes (`lode:involvedAgent`) et thèmes (`dc:subject`). La dimension temporelle n'est pas prise en compte dans ce travail et elle peut être le sujet d'une contribution future.

Proximité géographique

D'une manière générale, un utilisateur a tendance à participer à un événement quand celui-ci est géographiquement proche de sa ville (Quercia *et al.*, 2010). La distance géographique entre les événements suivant le prédicat `lode:atPlace` peut être normalisée en fonction d'un seuil θ fixe. Comme l'emplacement des villes des utilisateurs est absent dans EventMedia, on a donc mesuré la distance moyenne entre tous les événements passés dans le profil utilisateur. Selon la figure 5, la participation devient extrêmement rare après une distance égal à 80 Km, ce qui correspond au seuil utilisé dans notre calcul.

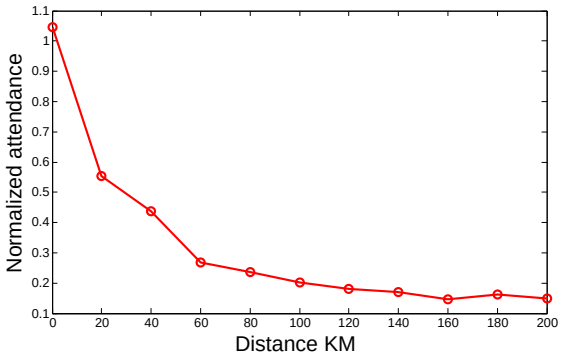

Figure 5. Taux de participation normalisé par distance en KM

Enrichissement des données

Afin d'enrichir le profil d'un objet ou d'un utilisateur, un système de recommandation peut bénéficier de la richesse du web des données. En particulier, le jeu de données DBpedia fournit des informations riches dans une variété de domaines. En utilisant les appariements entre les artistes de DBpedia et EventMedia, nous avons pu enrichir la description d'un événement par les thèmes des artistes de DBpedia. La raison de notre intérêt aux thèmes (genres musicaux) fournis par DBPedia est leur classification dans une structure hiérarchique bien définie avec un étiquetage cohérent.

Diversité thématique du profil utilisateur

L'objectif d'un système de recommandation est de proposer des produits qui répondent au profil de l'utilisateur. Cette tâche devient difficile si ce profil est marqué par une diversité thématique reflétant les objets avec qui l'utilisateur a interagi. En particulier, un événement social peut être associé à un seul thème général comme le cas de majorité des conférences, et il peut être associé à plusieurs thèmes comme le cas des grands festivals. Cependant, un utilisateur qui participe à un événement peut être intéressé à un seul ou plusieurs thèmes. En conséquence, la recommandation basée sur le contenu peut échouer à détecter les intérêts des utilisateurs à cause de la diversité thématique des événements passés. Pour atténuer cet impact, nous proposons une méthode pour modéliser les intérêts de l'utilisateur à l'aide de la technique LDA (Allocation de Dirichlet Latente (Blei *et al.*, 2003)) qui permet de modéliser des thèmes dans un corpus. Ensuite, nous classifions les événements dans le profil utilisateur en deux catégories : la première catégorie inclut les événements qui correspondent aux grands intérêts (pics) de l'utilisateur, la deuxième catégorie contient le reste des événements. Chaque catégorie est associé à un poids β que nous allons estimer par des méthodes d'optimisation décrites par la suite. Ainsi, la prédiction basée sur le contenu devient :

$$rank_{cb++}(u, e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \, \beta_p \, sim^p(e_i, e_j)}{\mid P \mid \, * \mid E_u \mid} \tag{10}$$

où $\beta_p = 1$ si le prédicat p est différent de `dc:subject`, sinon $\beta_{subject}$ est un poids qui dépend de la catégorie de l'événement e_j.

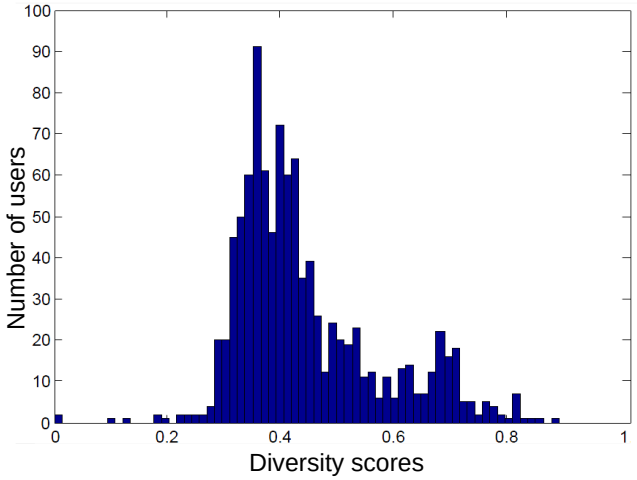

Figure 6. Distribution des scores de la diversité thématique sur un échantillon de 1 000 utilisateurs avec T = 30

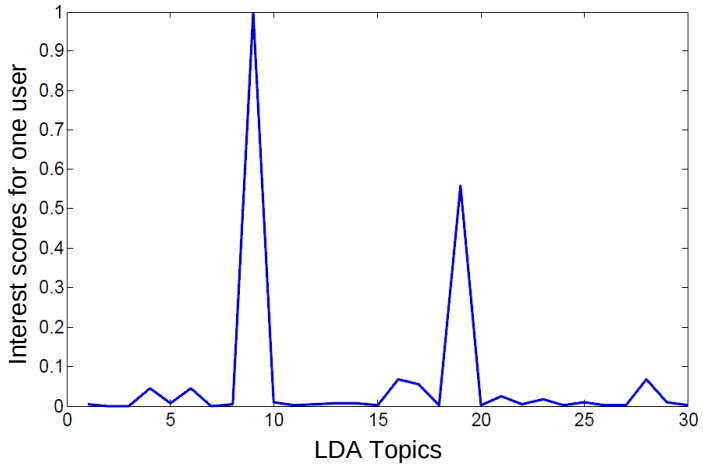

Figure 7. Distribution des intérêts thématiques pour un seul utilisateur avec T = 30

Afin de détecter les pics d'intérêts de l'utilisateur, on s'est inspiré de l'approche proposée par (Wu *et al.*, 2012) basée sur LDA. Chaque événement est considéré comme un document représenté par un ensemble de mots-clés. LDA génère un vecteur de dimension T (nombre de thèmes) pour chaque événement e_i indiquant la distribution des thèmes $\Theta_i = [\theta_i^1, \theta_i^2, ...\theta_i^T]$. Ensuite, on calcule la variance pour chaque thème t pour tout l'ensemble des événements E dans le profil utilisateur où $\Theta^t = [\theta_1^t, \theta_2^t, ...\theta_E^t]$ correspond au degré d'intérêt de l'utilisateur pour chaque thème

t. La diversité thématique d'un profil utilisateur est la moyenne des variances de tous les thèmes (moyenne de $\Theta^1, \Theta^2 ... \Theta^T$). Sur la figure 6, on montre les scores de diversité obtenus à partir d'un échantillon de 1 000 utilisateurs de Last.fm. La plupart des scores varie de 0,3 à 0,5 attestant que la majorité des utilisateurs ont des pics d'intérêt pour seulement quelques thèmes. Les scores qui sont proches de zéro correspondent aux utilisateurs qui sont quasiment inactifs rappelant le problème de démarrage à froid (cold-start). Sur la figure 7, on peut voir la modélisation des intérêts pour un seul utilisateur qui est fortement intéressé par le thème 9.

4.3.2. Filtrage collaboratif

Une forme d'interaction sociale est la participation collaborative (co-participation) aux événements. Le réseau social construit par le biais de la co-participation aux événements sociaux s'avère assez cohérent (Liu *et al.*, 2012). On suppose donc que le nombre des événements communs entre deux participants est relatif au degré de leur lien "d'amitié" ou leur similarité. Pour cela, on exploite les RSVP qui existent dans EventMedia afin d'intégrer le filtrage collaboratif dans le système de recommandation. Dans notre approche, nous visons non seulement à considérer la similarité entre deux participants, mais aussi la contribution d'un groupe d'amis. L'équation suivante prédit la décision de l'utilisateur u_i pour participer à l'événement e en se basant sur les RSVP de ses co-participants :

$$rank_{cf}(u_i, e) = \frac{\sum_{j \in C} a_{i,j}}{\mid C \mid} * \frac{\mid E_i \cap (\cup_{j \in C} E_j) \mid}{\mid E_i \mid} \quad (11)$$

où C est l'ensemble des co-participants de l'utilisateur u_i, E_i est l'ensemble des événements passés dans le profil de l'utilisateur u_i, et $a_{i,j}$ est le rapport du nombre d'événements partagés entre u_i et u_j par la cardinalité de E_j. Cette équation représente deux parties : (1) la première partie considère la participation de chaque co-participant individuellement ; (2) la deuxième partie considère tous les co-participants comme un groupe d'amis, et on suppose que le nombre d'événements partagés entre ce groupe d'amis reflète la force de leur lien.

4.3.3. Recommandation hybride

Pour exploiter à la fois les descripteurs du contenu et le filtrage collaboratif, nous proposons un système hybride par pondération d'une combinaison linéaire. Combinant les équations (10) et (11), la prédiction finale est :

$$rank(u, e) = rank_{cb++}(u, e) + \alpha_{cf}\, rank_{cf}(u, e) \quad (12)$$

où α_{cf} est le poids attribué au filtrage collaboratif, estimé en conjonction avec les poids α_p de l'approche basé sur le contenu.

Pour apprendre les poids de notre fonction de prédiction, nous utilisons trois méthodes : (1) La régression linéaire par la descente de gradient qui minimise la mesure d'erreur RMSE (Root Mean Squared Error) ; (2) L'algorithme génétique (Yeh

et al., 2007) pour maximiser la précision en spécifiant les paramètres suivants durant les expérimentations : taille d'une population = 30, itérations = 80, reproduction (crossover) = 0,9 et mutation = 0,01 ; (3) L'optimisation par essaims particulaires (PSO) (Kennedy, Eberhart, 1995) décrite antérieurement pour maximiser également la précision avec les paramètres suivants : taille d'une population = 30, itérations = 80, c_1 = 1,494, c_2 = 1,494 et w = 0,729 (paramétrage recommandé par (Eberhart, Shi, 2001)).

4.3.4. Evaluation

Pour évaluer notre approche, nous avons utilisé le référentiel Last.fm puisqu'il contient le plus grand nombre d'utilisateurs actifs. En utilisant SPARQL, nous avons extrait 2 436 événements situés dans la capitale *Londres* qui regroupe un nombre important d'utilisateurs dans EventMedia. Ces événements sont associés à 481 utilisateurs dont le taux de participation varie entre 15 et 50 pour éviter le problème de démarrage à froid. Cet échantillon contient 14 748 artistes, 897 lieux et 4 265 thèmes (tags de Last.fm). L'évaluation se fait en mesurant la précision et le rappel de top-N recommandations. La précision est le rapport du nombre des recommandations correctes sur le nombre N dans l'ensemble de test. Le rappel est le rapport du nombre des recommandations correctes sur le nombre des recommandations pertinentes. Le jeu d'apprentissage est représenté par 70 % de la matrice d'usage (30 % pour le jeu de test).

Pour souligner l'importance de l'interpolation des similarités et de l'enrichissement des données, nous évaluons les taux des valeurs nulles dans les matrices d'adjacences (voir tableau 8). On peut observer que notre approche a réussi à réduire les cases nulles dans les matrices associés aux prédicats discriminants (`lode:atPlace` et `lode:involvedAgent`) grâce à l'interpolation des similarités. L'enrichissement des données avec DBpedia a également permis de réduire légèrement ces cases nulles pour la matrice associée au prédicat `dc:subject`.

Tableau 8. Taux des cases nulles dans les matrices d'adjacence (1) avant et (2) après l'interpolation et l'enri chissement

Tâche	lode:atPlace	lode:involvedAgent	dc:subject
(1)	0,9942	0,9174	0,3175
(2)	0,6854	0,7392	0,2843

Afin d'estimer les coefficients α des équations 10 et 12, nous évaluons les méthodes d'apprentissage utilisées en fixant le coefficient $\beta_{subject} = 1$ (voir figure 8). Il est évident que si tous les coefficients sont égaux à 1, le système aura une mauvaise performance comme il n'y a aucune optimisation adaptative. En outre, les algorithmes évolutionnistes (GA et PSO) produisent des résultats nettement meilleurs que la régression linéaire (LR). Cela confirme les travaux récents (Cremonesi *et al.*, 2010) prouvant que les méthodes basées sur la mesure d'erreur n'améliorent pas la précision des top-N recommandations. Finalement, la méthode d'optimisation par essaims particulaires (PSO) a la meilleure performance où on a également observé une conver-

gence rapide vers la solution optimale par rapport à l'algorithme génétique. Dans ce qui suit, nous utilisons la méthode PSO pour estimer les coefficients α et $\beta_{subject}$.

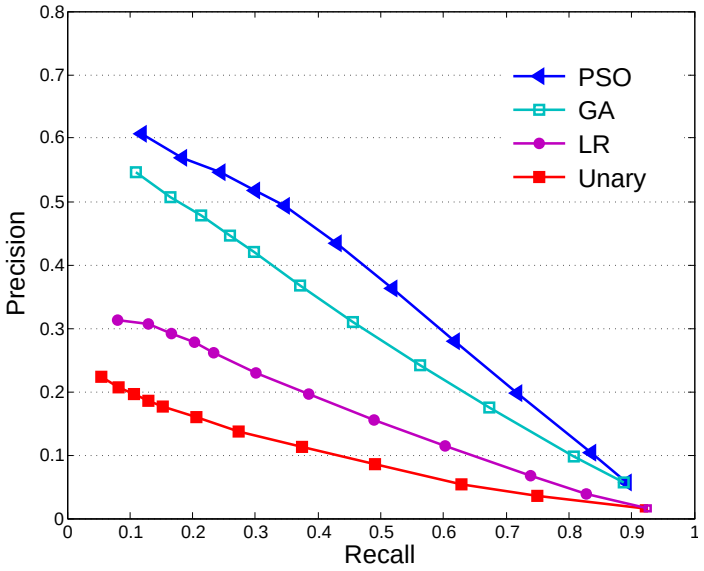

Figure 8. Précision et Rappel pour différentes méthodes d'apprentissage

Pour évaluer la contribution de chaque étape dans notre approche, nous examinons l'évolution de la performance du système en intégrant par ordre l'enrichissement à l'aide de DBpedia, la diversité thématique de profil utilisateur et le filtrage collaboratif (voir la figure 9). On observe que l'enrichissement des données avec DBpedia a légèrement augmenté la précision et le rappel. En effet, l'introduction des données plus cohérentes est un avantage pour réduire le bruit induit par l'annotation collective dans un service social. Les résultats ont été aussi améliorés par la modélisation des intérêts de l'utilisateur. D'ailleurs, on a observé que le poids $\beta_{subject}$ attribué aux événements dans les pics d'intérêt d'utilisateur est quatre fois plus important que le poids attribué au autres événements. Enfin, l'intégration du filtrage collaboratif a augmenté considérablement la performance du système. Cette amélioration est en accord avec l'étude centrée utilisateur concernant la participation aux événements mettant l'accent sur le rôle important de la dimension sociale dans la prise de décision (Troncy, Fialho *et al.*, 2010).

La dernière partie de l'évaluation compare notre approche hybride avec des méthodes existantes basées sur le filtrage collaboratif comme la méthode classique basée sur les utilisateurs (Resnick *et al.*, 1994) et la méthode UBExtended proposée par (Pessemier *et al.*, 2012). UBExtended est une approche basée sur une cascade de deux systèmes au filtrage collaboratif afin de recommander des événements populaires. Les résultats de cette comparaison (voir figure 10) montre que l'approche UBExtended optimise la performance du système par rapport à la méthode classique.

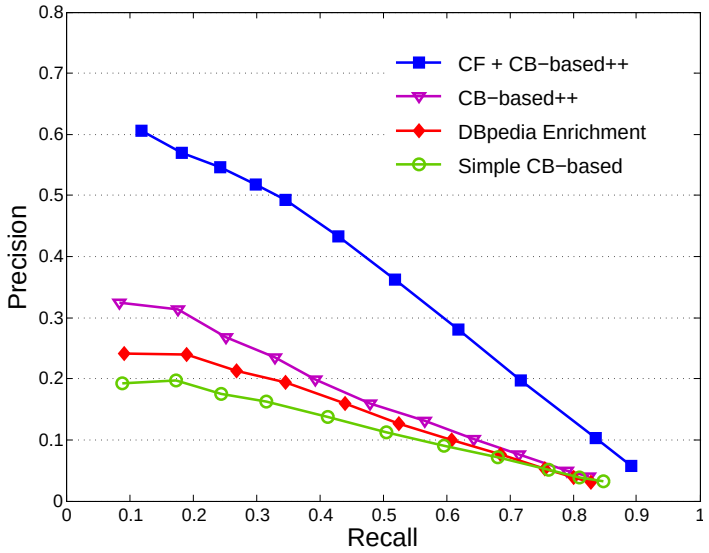

Figure 9. Evolution de la performance du système en intégrant l'enrichissement avec DBpedia, la diversité thématique (CB-based++) et le filtrage collaboratif (CF)

Finalement, la méthode hybride produit une meilleure performance ce qui confirme l'avantage de l'hybridation dans un système de recommandation.

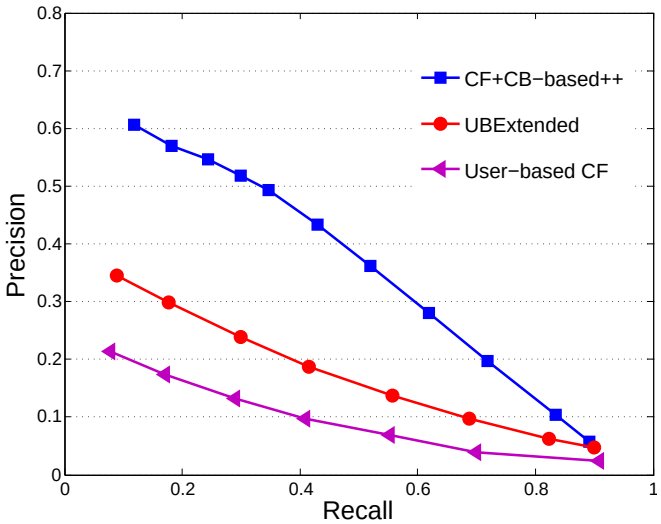

Figure 10. La comparaison de l'approche hybride avec les systèmes basés sur le filtrage collaboratif

5. Applications

Dans cette section, nous présentons deux exemples d'applications qui utilisent le jeu de données EventMedia pour permettre à des utilisateurs de découvrir ou revivre des événements à partir de médias.

5.1. L'interface EventMedia

L'objectif de l'interface EventMedia (Khrouf, Troncy, 2011b) est d'assurer une navigation fluide entre les données interconnectées dans un espace basé sur les événements. D'une manière générale, les utilisateurs souhaitent découvrir les événements par le biais d'invitations et de recommandations, ou en filtrant les événements en fonction de leurs intérêts. Dans notre interface, nous fournissons des mécanismes pour filtrer les événements par lieu, catégorie, date, etc. Une fois qu'un événement est sélectionné, les médias sont présentés pour transmettre l'expérience utilisateur, ainsi que d'autres informations comme la description de l'événement, les artistes ou les utilisateurs qui ont participé (voir la figure 11).

Figure 11. Interface illustrant un concert de Lady Gaga *en 2010*

L'interface d'EventMedia est disponible à http://eventmedia.eurecom.fr [16]. Techniquement, on s'est basé sur ELDA, une implémentation en Java qui permet d'accéder aux données RDF en utilisant des méthodes REST qui sont traduites en des requêtes

16. Vidéo : http://eventmedia.eurecom.fr/demo.html

SPARQL. ELDA sérialise les données en XML ou JSON ce qui peut être facilement utilisé dans des framework Javascript. En particulier, on a utilisé le framework Backbone.js [17] qui facilite le développement d'interfaces utilisateurs adaptées à des tailles d'écran différent. Il fournit une intégration élégante des requêtes REST, ce qui rend très simple son utilisation avec ELDA.

5.2. L'interface EventMap : la navigation à facettes parallèles

La navigation à facettes (ou recherche à facettes (Tunkelang, 2009)) est un paradigme largement utilisé pour l'exploration d'un entrepôt de données, dont chacune a un certain nombre d'attributs (ou facettes) qui peuvent prendre des valeurs différentes. Dans le cas le plus simple, un utilisateur spécifie les valeurs d'une ou plusieurs facettes afin de se focaliser sur un sous-ensemble des entités du référentiel, qui sont ensuite affichées. Cependant, l'utilisateur ne peut examiner que le résultat d'une seule requête à la fois. Cela ne permet pas de comparer les résultats de plusieurs requêtes ou de raffiner les résultats des requêtes en parallèle. Pour assurer une telle navigation, l'interface EventMap introduit le paradigme de navigation à facettes parallèles (Buschbeck *et al.*, 2013) qui permet de construire un arbre de requêtes et leurs résultats dont la structure met en avant les points communs entre ces requêtes.

Figure 12. Interface EventMap basée sur la navigation à facettes parallèles

17. http://backbonejs.org/

Le démonstrateur EventMap (figure 12) est disponible en ligne à http://eventmap-ui.appspot.com/. Les entités représentées dans ce démonstrateur sont des événements associés au projet "la Capitale mondiale du design 2012" qui a eu lieu dans la capitale finlandaise *Helsinki*, ainsi que d'autres événements dans la même période. Le système EventMap est composé de (i) un client qui s'exécute dans le navigateur de l'utilisateur, (ii) une application qui s'exécute dans le serveur AppEngine [18] de Google, et (iii) un référentiel de données fournis par EventMedia.

6. Conclusion

Dans cet article, nous avons présenté une modélisation sémantique d'événements avec l'ontologie LODE, ainsi que la construction du jeu des données EventMedia composé de descriptions d'événements et de média. Nous avons aussi décrit l'architecture de notre système afin d'assurer un alignement en temps réel et faire face à l'évolution dynamique des services sociaux. Cet alignement a été basé sur une approche de réconciliation indépendante du domaine pour lier les événements dans le web des données. En particulier, notre approche montre l'importance de la corrélation et la couverture des prédicats pour optimiser les résultats. De plus, nous avons montré les avantages à utiliser le web sémantique dans un système de recommandation thématique, et nous avons proposé une approche hybride pour recommander des événements. L'évaluation a démontré l'importance de la dimension sociale et la modélisation des intérêts d'un utilisateur pour améliorer la recommandation. Enfin, nous avons décrit deux applications pour explorer, visualiser et comparer des événements partageant des lieux, des personnes ou des thématiques communes.

En matière de perspectives, nous planifions de : (1) étendre la couverture d'EventMedia par l'intégration d'autres sources largement utilisées comme Facebook et Youtube ; (2) mettre en place une étude centrée utilisateur afin d'évaluer l'utilisabilité des interfaces décrites, à savoir : EventMedia et EventMap ; (3) intégrer et évaluer d'autres critères importants dans la recommandation d'événements tels que la popularité d'un événement ou un artiste ainsi que la dimension temporelle.

Bibliographie

Blei D. M., Ng A. Y., Jordan M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, vol. 3, p. 993-1022.

Buschbeck S., Troncy R., Jameson A., Khrouf H., Spirescu A., Suominen O. *et al.* (2013). Parallel Faceted Browsing. In *Acm conference on human factors in computing systems (chi'13), interactivity track.* Paris, France.

Cremonesi P., Koren Y., Turrin R. (2010). Performance of recommender algorithms on top-n recommendation tasks. In *Acm conference on recommender systems (recsys'10).* Barcelona, Spain.

18. https://appengine.google.com/

Cyganiak R., Jentzsch A. (2010). *Linking Open Data cloud diagram.* LOD Community (http://lod-cloud.net/).

Di Noia T., Mirizzi R., Ostuni V. C., Romito D., Zanker M. (2012). Linked open data to support content-based recommender systems. In *8^{th} international conference on semantic systems (i-semantics).* Graz, Austria.

Eberhart R. C., Shi Y. (2001). Particle swarm optimization: developments, applications and resources. In *Ieee congress on evolutionary computation*, vol. 1, p. 81-86.

Fialho A., Troncy R., Hardman L., Saathoff C., Scherp A. (2010). What's on this evening? Designing User Support for Event-based Annotation and Exploration of Media. In *1^{st} international workshop on events - recognising and tracking events on the web and in real life*, p. 40–54. Athens, Greece.

Généreux M., Santini M. (2007). Défi: Classification de textes français subjectifs. In *Actes de l'atelier de clôture du 3ème defi fouille de textes*, p. 83–93. Grenoble, France.

Hage W. van, Malaisé V., Vries G. de, Schreiber G., Someren M. van. (2009). Combining Ship Trajectories and Semantics with the Simple Event Model (SEM). In *1^{st} acm international workshop on events in multimedia (eimm'09).* Beijing, China.

Hobbs J., Pan F. (2006). *Time Ontology in OWL.* W3C Working Draft. (http://www.w3.org/TR/owl-time)

Jentzsch A., Isele R., Bizer C. (2010). Silk - Generating RDF Links while publishing or consuming Linked Data. In *9^{th} international semantic web conference (iswc'10).* Shanghai, China.

Kennedy J., Eberhart R. C. (1995). Particle swarm optimization. In *Ieee international conference on neural networks*, p. 1942–1948. Perth, Australia.

Khrouf H., Troncy R. (2011a). EventMedia Live: Reconciliating Events Descriptions in the Web of Data. In *6^{th} international workshop on ontology matching (om'11).* Bonn, Germany.

Khrouf H., Troncy R. (2011b). EventMedia : Visualizing Events and Associated Media. In *Demo session at the 10^{th} international semantic web conference (iswc'11).* Bonn, Germany.

Liu X., He Q., Tian Y., Lee W.-C., McPherson J., Han J. (2012). Event-based social networks: Linking the online and offline social worlds. In *18^{th} acm sigkdd conference on knowledge discovery and data mining(kdd'12).* Beijing, China.

Nikolov A., d'Aquin M., Motta E. (2012). Unsupervised learning of link discovery configuration. In *9^{th} extended semantic web conference (eswc'12).* Heraklion, Crete, Greece.

Niu X., Rong S., Zhang Y., Wang H. (2011). Zhishi.links results for oaei 2011. In *6^{th} international workshop on ontology matching.* Bonn, Germany.

Pazzani M. J., Billsus D. (2007). The adaptive web. In, p. 325–341. Springer-Verlag.

Pessemier T. D., Coppens S., Geebelen K., Vleugels C., Bannier S., Mannens E. *et al.* (2012). Collaborative recommendations with content-based filters for cultural activities via a scalable event distribution platform. *Multimedia Tools and Applications*, vol. 58, n° 1, p. 167-213.

Quercia D., Lathia N., Calabrese F., Di Lorenzo G., Crowcroft J. (2010). Recommending social events from mobile phone location data. In *10^{th} ieee international conference on data mining*. Sydney, Australia.

Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In *Acm conference on computer supported cooperative work (cscw'94)*, p. 175–186. Chapel Hill, North Carolina, USA.

Salton G., Wong A., Yang C. S. (1975). A vector space model for automatic indexing. *Communications of The ACM*, vol. 18, n° 11, p. 613-620.

Song D., Heflin J. (2011). Automatically generating data linkages using a domain-independent candidate selection approach. In *10^{th} international semantic web conference (iswc'11)*. Bonn, Germany.

Troncy R., Fialho A., Hardman L., Saathoff C. (2010). Experiencing Events through User-Generated Media. In *1^{st} international workshop on consuming linked data (cold'10)*. Shanghai, China.

Troncy R., Shaw R., Hardman L. (2010). LODE: une ontologie pour représenter des événements dans le web de données. In *21^{st} journées d'ingénierie des connaissances (ic'10)*, p. 69–80. Nîmes, France.

Tunkelang D. (2009). *Faceted search*. Morgan & Claypool Publishers.

Vatant B., Rozat L. (2011). *VOAF (Vocabularies of a Friend) vocabulary*. Mondeca (http://www.mondeca.com/foaf/voaf-doc.html).

Wu H., Sorathia V., Prasanna V. (2012). When diversity meets speciality: Friend recommendation in online social networks. *ASE Human Journal*, vol. 1, n° 1, p. 52-60.

Yeh J. yuan, Lin J. yi, Ke H. ren, Yang W. pang. (2007). Learning to rank for information retrieval using genetic programming. In *Sigir workshop on learning to rank for information retrieval*.