Linking and Mining Event-centric Data in the Semantic Web

Houda Khrouf

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2014-ENST-0035

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Sciences de l'Information et de la Communication »

présentée et soutenue publiquement par

Houda KHROUF

le 30 Juin 2014

Alignement et Fouille de Données Événementielles dans le Web Sémantique

Directeur de thèse : **Raphaël TRONCY**

Jury

M. Talel ABDESSALEM, Professeur, TELECOM ParisTech	Président
M. Geert-Jan HOUBEN, Professeur, Delft University of Technology	Rapporteur
Mme. Catherine FARON ZUCKER, Maître de Conférences, Université de Nice	Rapporteur
M. John DOMINGUE, Professeur, The Open University	Examinateur
M. Tommaso DI NOIA, Maître de Conférences, Polytechnic University of Bari	Examinateur

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

THÈSE

Linking and Mining Event-centric Data in the Semantic Web

Houda KHROUF

A doctoral dissertation submitted to:

TELECOM ParisTech

in partial fulfillment of the requirements for the degree of:

Doctor of Philosophy

Specialty : COMPUTER SCIENCES AND INFORMATION TECHNOLOGY

June 3^{th}, 2014

Approved by the following jury

President of the jury:

Prof. Talel ABDESSALEM	Telecom ParisTech, France

Reviewers:

Prof. Geert-Jan HOUBEN	Delft University of Technology, Netherlands
Dr. Catherine FARON ZUCKER	University of Nice Sophia Antipolis, France

Examiners:

Prof. John DOMINGUE	The Open University, United Kingdom
Dr. Tommaso DI NOIA	Polytechnic University of Bari, Italy

Supervisor:

Dr. Raphaël TRONCY	EURECOM, France

Abstract

Recently, the widespread growth of social media has shifted the way people explore and share information of interest. Part of this evolution is the event landscape increasingly augmented by the user-generated content leading to vast amount of event-centric data. In today's Web, numerous are the websites that provide facilities to organize and publish events, and to share related thoughts and captured media. However, the information about the events, the social interactions and the representative media are all spread and locked into the sites providing limited event coverage and no interoperability of the description. To fully benefit from the event, users are constrained to monitor different channels some of which suffer from the information overload. The goal of this thesis is to provide a unified environment that provides broad event coverage along with complete description and illustrative media, and to investigate efficient approaches that can benefit content personalization. The major challenge is to face the complex nature of events as multifaceted, ephemeral and social entities.

Various distributed platforms host a wide variety of scheduled events along with related media and background knowledge, making the user-contributed Web a primary source of information about any real world happening. Mining in real-time the connections between these heterogeneous and spread data fragments is a key factor to improve data quality and to enable opportunistic discovery of events. Towards this goal, we integrate different sources using Linked Data, so that we can explore the information with the flexibility afforded by the Semantic Web technologies. More precisely, we leverage the wealth of information derived from event-based services, media platforms and social networks to build a Web environment that allows users discovering meaningful connections between events, media and people.

On the other hand, users tend to be overwhelmed by the massive amount of information available in event-based websites. This fact requires valuable personalization solutions that cope with the information overload and help organize data. In particular, recommendation and community detection are two promising solutions that have been widely investigated in research. Yet their applications in event domain are still elusive. Thus, we propose a hybrid recommender system that capitalizes on ontology-based event representation along with the collaborative filtering techniques. Second, we propose an approach based on semantic modularity maximization to discover overlapping semantic communities in event-based social network.

Résumé

La croissance exponentielle de l'usage des médias sociaux a changé la façon d'explorer et de partager l'information. Une partie de cette évolution concerne la manière dont notre activité sociale est structurée autour d'événements. Avec le développement du Web 2.0, de nombreux sites de partage fournissent une grande quantité de données décrivant des événements passés ou à venir, et certains d'entre eux affichent des médias et des interactions sociales attachés à ces événements. Cependant, l'information disponible est souvent incomplète, erronée et restreinte dans une multitude de sites Web. Dans cette thèse, nous étudions l'intégration des données événementielles dans un environnement centralisé, et nous étudions de nouvelles approches qui pourraient améliorer la personnalisation du contenu. La thèse est organisée autour de deux parties principales portant sur les défis majeurs liés à la nature complexe d'événements qui sont des entités éphémères, sociales et multidimensionnelles. Dans la première partie, nous étudions l'enrichissement des données en exploitant les technologies du Web sémantique afin d'intégrer des sources hétérogènes telles que les référentiels d'événements, les plates-formes de médias et les réseaux sociaux. Dans la deuxième partie, nous abordons le problème de la surcharge d'information. Elle comprend une étude de nouvelles approches de personnalisation afin d'aider les utilisateurs à découvrir des événements et des personnes qui correspondant à leurs centres d'intérêts. Notre étude souligne l'importance de la modélisation ontologique et le filtrage collaboratif dans un système de recommandation. Nous proposons ensuite une nouvelle solution pour la détection de communautés recouvrantes et sémantiques dans les réseaux événementiels. Les approches proposées dans cette thèse fournissent de nouvelles bases pour la construction d'un environnement Web intégrant de nouveaux mécanismes d'exploration et d'organisation des données événementielles.

Acknowledgments

Working as a PhD student in Eurecom was a great experience that would not be achieved without the help and support of many people, who I would like to acknowledge here.

First and foremost, I would like to thank my supervisor Dr. Raphaël Troncy for his invaluable support and great guidance throughout my thesis. I would like to express my gratitude to him for provided me with a lot of freedom to pursue my research. This work would not have been possible without his scientific knowledge and constructive advice.

I would like to extend my sincere thanks to my committee members, the reviewers Prof. Geert-Jan Houben and Dr. Catherine Faron-Zucker, and the examiners Prof. Talel Abdessalem, Prof. John Domingue and Dr. Tommaso Di Noia for their precious time and shared insights.

I am particularly indebted to my colleagues who more or less directly contributed to my Ph.D. Precisely, I would like to acknowledge the support of my friends: Vuk, Ghislain, José Luis and Giuseppe for their inspiring collaboration and fruitful discussions. It was a pleasure to work and exchange with them. Also, I thank all those working at EURECOM, they made my stay very pleasant.

Above all, I would like to pay my deepest gratitude to my family for their unwavering support and devotion. I would never thank enough my parents for their unconditional love, trust and sacrifice. Last but not least, special thanks go to my friends for their constant friendship, moral and infinite support.

Contents

List of Figures

List of Tables

Glossary

Here are the main acronyms used in this document. The meaning of an acronym is usually indicated once, when it first appears in the text.

AI	Artificial Intelligence
API	Application Programming Interface
CB	Content Based recommender system
CDF	Cumulative Distribution Function
CF	Collaborative Filtering
EAV	Entity–Attribute–Value model
EBSN	Event-based Social Network
ELDA	Epimorphics Linked Data API
FOAF	Friend Of A Friend
GA	Genetic Algorithm
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
IR	Information Retrieval
ISBN	International Standard Book Number
JSON	JavaScript Object Notation
LBSN	Location-based Social Network
LDA	Latent Dirichlet Allocation
LOD	Linked Open Data
LODE	Linking Open Descriptions of Events
NE	Named Entity
NER	Named Entity recognition
NERD	Named Entity Recognition and Disambiguation
NLP	Natural Language Processing
OWL	Web Ontology Language
PLSA	Probabilistic Latent Semantic Analysis
PSO	Particle Swarm Optimization
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
REST	Representational State Transfer
RSS	Really Simple Syndication
RSVP	Request for a Response
SIOC	Semantically-Interlinked Online Communities
SKOS	Simple Knowledge Organization System

SPARQL	Query Language for RDF
SUS	Stochastic Universal Sampling
TDT	Topic Detection and Tracking
URI	Universal Resource Identifier
URL	Universal Resource Locator
VSM	Vector Space Model
W3C	World Wide Web Consortium
XML	Extensible Markup Language

CHAPTER 1

Introduction

Many services such as event directories, social networks and media platforms host an ever increasing amount of event-centric data. Recently, they have attracted people to organize and distribute their personal data according to occurring events, to share related media and to create new social connections. Still, this data needs to be structured and integrated in order to enhance different tasks such as content presentation and personalization.

1.1 Context and Motivation

Roughly speaking, "*event*" is a phenomena that has happened or scheduled to happen at a specific place and time. According to recent studies in neuroscience [144], event is also considered as past experience with which humans remember their real life. A common practice for humans is to naturally organize their personal data according to occurring events: wedding, conference, concert, party, etc. They would like to plan activities according to future events or to record what happened during past events. Along with the emergence of Web 2.0, people become more involved in online activities sharing rich content to describe events and engaging in social interactions. This is reflected in many social sites where a large amount of data exists in multiple modalities such as the event details (e.g., time, location) and the explicit RSVP (i.e., expressing the user intent to join social events) in event directories (e.g., Eventful, Last.fm, Lanyrd, Facebook); the photos and videos captured during events and shared on media platforms (e.g., Flickr, YouTube), and finally the digital chatter generated by reactions to events in network sites (e.g., Twitter, Facebook). Yet this knowledge forms a huge space of disconnected data fragments providing limited event coverage [40]. For instance, while Last.fm sustains a broad coverage on event attendance, other valuable details are often missing such as description, price and media. Users tend to use other channels to complement the event overview. Moreover, most of event directories provide limited browsing options (e.g., lack of location map) and unreliable event recommendation (e.g., no consideration of like-minded users). These limitations have been notably highlighted in an exploratory user centered study conducted to assess the perceived benefits and drawbacks of event websites [136]. Having in mind the findings of this study, we focus on two major tasks which are data reconciliation and personalization.

1.1.1 Data Reconciliation

A large amount of event-centric data is spread over the Web, however, often incomplete and always locked into multiple sites. How to leverage the wealth of this data is a serious challenge towards providing a broad coverage of events. As a solution, data integration is a prominent way in order to deliver more complete and accurate information. In particular, the recent use of the Semantic Web technologies has proven its efficiency to ensure a large-scale and flexible data integration. In fact, the Semantic Web is predicated on the availability of large amount of structured data as RDF, not in isolated islands, but as a Web of interlinked datasets. Moreover, with the use of ontologies, developers can structure heterogeneous data into unified model independently of particular applications, while explicitly representing enriched semantics. One success of the Semantic Web is the Linked Data cloud [1] which is an ongoing project that interlinks RDF datasets on a large scale and follows the principles outlined by Tim Berners-Lee [2] in 2006. A fundamental concern in this context is the data interlinking or reconciliation required to achieve the vision of the Linked Data. However, data sources do not often share commonly accepted identifiers (e.g., DOI identifier or ISBN codes) and they usually make use of different vocabularies. As a solution, many approaches have been proposed to address two main sub-tasks of data reconciliation: the former is the ontology matching which refers to the process of determining correspondences between ontological concepts; the latter is the instance matching which refers to the process of determining correspondences between individuals. In this thesis, we focus on the instance matching task to discover identical individuals referring to the same real-world entity. Indeed, it has been shown that data reconciliation at the instance level is an advantageous asset to enhance data quality by improving both completeness and accuracy [102]. For example, while a data source contains few details about involved artists, another one may provide more information in form of biography with complete discography. Hence, we propose to link identical event-centric entities so that a user would be able to navigate from one entity to another as if he is in a homogeneous environment.

On the other hand, real-world events often trigger a tremendous activity on numerous social media platforms. Participants share captured photos and videos during events, and engage in discussions with microposts on social networks. It would be of great benefit to augment the event views with user-contributed social media. In fact, mining the intrinsic relationships between events and media has been the subject of many research studies. Most of them focus on event detection from user-generated content that describes breaking news or social events [86, 7, 124]. Automatic event detection is essentially a clustering problem aiming to group together media documents discussing the same event. Few other existing works have studied the connection betwen events and media within the field of data reconciliation [121, 34]. The idea behind is to compare instances of different ontological classes

1. http://linkeddata.org/
2. http://www.w3.org/DesignIssues/LinkedData.html

(e.g., event class and media class) using their related features such as named entities and contextual information. In this thesis, we exploit this idea and we attempt to bridge the gap between structured events and unstructured media data.

Reconciling event-centric entities or enriching events with media have in common some challenges induced by the use of online, heterogeneous and distributed sources. First, the same real-world entity is often represented in different ways across the disparate data sources. Some of these entities may be related with short descriptions and featuring noisy information. Moreover, the user-generated content exists typically at a large scale and evolves dynamically providing a daily and significant amount of events, locations, media, etc. These challenges demand a scalable, real-time and efficient techniques to reconcile data.

1.1.2 Personalization Techniques

Personalization in online social sites have gained momentum over the recent past years. Providing assistance to make decision and select reliable products become part of primary concerns in the e-service area. More specifically, integrating personalization techniques in event-based services is a key advantage to attract people to attend relevant events and discover new social connections. Such techniques recently start to draw attention as has been attested by the VP (Vice President) Operations of Eventful who reported that "*When we really got serious about personalization, we started talking about it a few years ago and we really got busy a couple of years ago*" [3].

One personalization technique is to build a recommender system that decodes the user interests and optimizes accordingly the information perceived. To help such system predict items of interest, various clues are available ranging from the user profile, explicit ratings, to past activities and social interactions. Different from a classic item, an event occurs at a specific place and during a period of time to become worthless for recommendation. While the classic items continuously receive useful feedback, an event is attended only once by participants. This fact makes very sparse the user preferences related to events. The transient nature of events leads to very limited number of participants who attended an event. Given this high sparsity, traditional recommender systems fail to handle event recommendation where both content and social information need to be considered [26].

Another innovative technique is to position the user within one or more communities, instead of an isolated individual [109] so that he/she can discover new social connections. In order to enable community-driven personalization, the system needs to analyze networked data and reveal the underlying communities. This demands an efficient method to detect meaningful communities which can in turn benefit various tasks such as people recommendation, customer segmentation, recommendation and influence analysis. In research, several studies have been devoted to the problem of community detection, but mostly focused on the linkage structure of the network. They assume that the proximity of users is reflected solely

3. Paul Ramirez, MarketingSherpa Email Summit 2014.

by their interactions strength. However, such methods do not consider the semantic dimension and often group users having different interests. This problem becomes important when a user interacts with different social objects (e.g., events) inducing highly diverse topics in his/her profile. Consequently, there is a need to incorporate the semantic information along with the linkage structure for detecting meaningful and overlapping communities [28, 146].

In this thesis, we tackle the problems related to event recommendation and to community detection in event-based social network. The challenge is to deal with the complex nature of events where social and content information are both important.

1.2 Thesis Contributions

As a multidimensional, ephemeral and social entity, the notion of "*event*" presents significant challenges for research community. In this thesis, we attempt to overcome these challenges and we propose some approaches related to data reconciliation and personalization. In summary, the main contributions of this work are as follows:

- We built a framework that aggregates in real-time event-centric data retrieved from heterogeneous sources. Our strategy is to design a new architecture flexible enough in order to accommodate ongoing growth. Such flexibility is ensured by the capability to add new Web services and by the use of Semantic Web technologies. The data, continuously collected in real-time, is converted to RDF using existing vocabularies and then stored in a triple store. The entire dataset is called EventMedia.
- We propose heuristics to mine the intrinsic connections of event-centric data derived from event directories, media platforms and Linked Data. Given the dynamics of social sites, our approach ensures a real-time reconciliation maintaining a dynamic content enhancement. First, we propose a domain-independent reconciliation approach that identifies identical entities residing at heterogeneous sources. Then, we tackle the problem of aligning structured events with unstructured media items based on Natural Language Processing (NLP) techniques.
- We built some friendly Web applications that consume Linked Data and meet the user needs: relive experiences based on background knowledge and help create events with consistent details. Then, we highlight the benefits of Linked Data to steer the behavioral analysis and to improve the user profiling.
- We propose a hybrid system to recommend events based on content features and collaborative participation. This system exploits the ontology-enabled feature extraction and enriches an event profile with Linked Data. It is also enhanced by an effective modeling of user interests.
- We introduce a novel approach that detects overlapping semantic communities within event-based social network. Our approach exploits the hierarchical clustering and combines both the semantic features and the the linkage structure.

1.3 Thesis Outline

The work presented in this thesis first describes how to integrate event-centric data into Linked Data. Then, it focuses on consuming this data to build some Web applications and to propose novel personalization approaches. The rest of this manuscript is organized as follows:

Chapter 2 is dedicated to overview the background of our work including the research in event domain and some paradigms related to the Semantic Web. We first introduce the important aspects related to events and the basic concepts in the Semantic Web. Then, we describe the evaluation criterion used throughout this work. The rest of this manuscript is composed of two major parts:

1. In the first part, we focus on the building task that retrieves event-centric data from distributed sources and integrates them into one semantic knowledge base called EventMedia. Such task includes crawling, structuring and linking data, which needs to be ensured with the flexibility afforded by the Semantic Web technologies. The contributions of this part have been published in [68, 69, 66, 63]. This part is composed of two chapters:
   - **Chapter 3** describes how data has been extracted, structured and published following the best practices of the Semantic Web. In particular, we pay attention to create a flexible framework that performs those tasks, and eases the addition of event and media Web sites.
   - **Chapter 4** studies the problem of data reconciliation in a heterogeneous environment. We present our approach to detect identical entities in event-centric data by the use of instance matching techniques. Then, we propose an NLP-based approach to align events with microposts, thus bridging the gap between structured and unstructured content.
2. In the second part, we exploit the constructed knowledge base EventMedia for various applications. The goal is to highlight the benefits of Linked Data to improve the event view and to explore solutions for advanced personalization. The contributions of this part have been published in [67, 65, 64, 70, 71]. This part is composed of three chapters:
   - **Chapter 5** presents three Web applications that support better visualization and help users search, browse and create events. Besides, it underlines the benefits of our knowledge base, as part of Linked Data, to understand some facts about the user behavior.
   - **Chapter 6** presents our approach built on top of the Semantic Web to recommend social events. The idea is to leverage structured and expressive representation of events to predict what a user likes. Our approach is then augmented by collaborative

filtering recommendation that takes into account the social dimension.

- **Chapter 7** describes how the event-centric activities have been exploited to construct event-based social network in online and offline worlds. Then, it presents our approach proposed to detect overlapping semantic communities taking into account the semantic topics and the linkage structure within a network.

Chapter 8 concludes the presented work and outlines new research directions.

CHAPTER 2

Background

In the last few years, an increasing interest in event domain has led to diverse contributions in research. In this chapter, we provide a background analysis on the definition of "*event*" in the Social Web and on the perceived qualities of available event directories. Then, we overview the Semantic Web technologies considered as powerful means to ensure a large scale data integration. Finally, we present some evaluation metrics used throughout this thesis. For more details about basic concepts and involved techniques, we provide an extension of this background in Appendix B.

2.1 Events on the Web

An ever increasing amount of event-centric knowledge is spread over multiple websites, either materialized as calendar of events or illustrated by cross-media documents. Determining what an event is and how people use those sites are two important research questions. In this section, we present the event definition adopted in this thesis, and we provide an overview of some social sites as well as the perceived benefits and drawbacks of using them.

2.1.1 Event Definition and Characterization

What is meant by the word "event"? has always been a research question leading to several meanings. This term has received substantial consideration across different fields such as philosophy [20] and computer science [2]. From a broader point of view, a real event is considered as something that happens: a happening, an occurrence, an event [126]. This definition has been extended in a philosophical study to characterize events as an abstract concept in which the meaning depends on the target type such as activity, state or action [20]. From technical point of view, an earlier work in Topic Detection and Tracking (TDT) field defines an event "as something that happens at a particular time and place" [2]. This definition puts emphasis on the spatial-temporal aspect, which seems to be adopted by many other researchers [84, 145]. However, while events can happen at a specific time, other events continue over a long period of time. Moreover, associating a specific location to events fails to handle some events which may happen in different venues. These facts have led to other definitions in the literature attempting to cast an event to just a temporal entity [114] or to stress on the geographical dimension [130]. To sum up, by drawing together all these defini-

tions, three important views appear to identify what an event is. These views are represented by three *Ws* questions: *what*, *when* and *where*.

Later on, some researchers point out a missing concept that could define an event. They attempt to pay attention to "*who*" was involved in the event. Although events can happen without participants, it seems important to consider this aspect when it comes to describe the people's experiences. Thus, the definition in [2] has been extended to "an event is something that has a specific time, location, and people associated with it" [1]. For instance, it has been shown that the "*who*" view is important to define a historical event which is described by five elements: object, person, location, time and cause [101]. While causality appear in some definitions, it is of less significance to our work since we are not primarily interested in linking events by cause/effect relationships. In [129], the authors proposed a study to compare existing semantic models that attempt to represent events in a structured format. They propose an interoperable model to represent intersubjective "consensus reality" over all event definitions. Based on this model, we define an event in terms of the four *Ws* questions as follows:

1. *What* happened: represented by a set of descriptive terms.
2. *Where* it happened: associates an event with any number of places.
3. *When* it happened: associates an event with a specific time or period of time.
4. *Who* was involved: distinguishes between people having "active" or "passive" role.

2.1.2 Social Websites

Events on the Web exist in two different types: *unstructured* and *structured*. On the one hand, unstructured events are mostly represented in form of natural language phrases which require complex parsing and extraction mechanisms. On the other hand, structured events are represented in a well-defined structure that may differ from one site to another. Currently, there exists a large variety of websites that host structured information about past and upcoming events, some of which may display media. In this thesis, we focus on structured events as provided by some popular event websites. In the following, we provide an overview about these sites as well as the platforms which host related media.

Event Webites

Many websites aim to help users search and share information about past and upcoming events. Whilst some websites focus on a specific type of events (e.g., musical, conference), other ones provide a wide span of different types including film, theater, exhibition, etc. In this thesis, we use some popular event sites described as follows:

- **Last.fm** [1]**:** is the largest music based platform founded in 2002 and having more than 30 million active users. It allows to build a user profile based on listening preferences

1. http://www.last.fm

of music collection or radio station. In October 2006, Last.fm incorporated a system that lets users post musical concerts with some details (date, venue, location, artists, etc.). Users are also able to express their intent to attend events using RSVP (e.g., *I'm going*). They can register in any group which may be linked to artists or countries, and can add other users as friends. Finally, tags and comments are also possible on almost any item such as a user, event, artist or track. Figure 2.1 depicts the homepage of Last.fm.

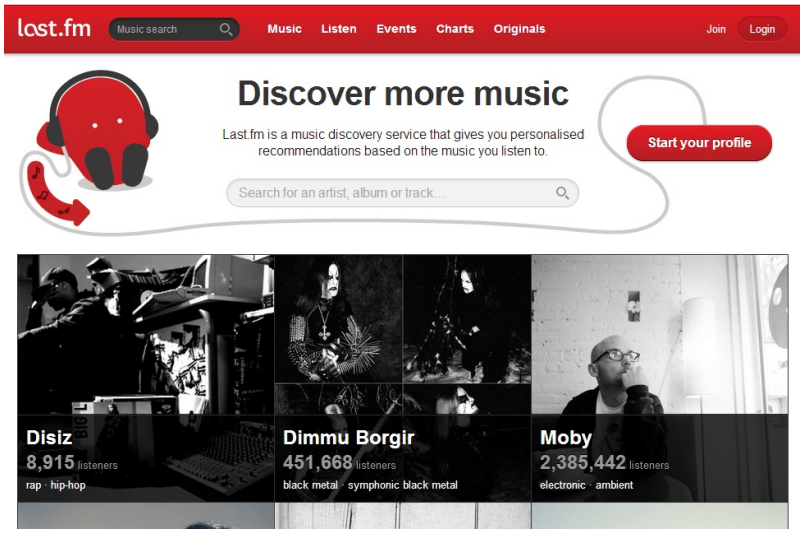

Figure 2.1: Last.fm homepage

- **Eventful** [2]**:** is a popular event-based service founded in 2004. It boasts one of the world's largest databases of events covering a wide variety of domains such as sport, cinema, family, education and other local entertainment. It allows users searching for events by location, time, category, artist and descriptive keywords. It also provides functionality to view and manage a list of favorite artists and venues. Figure 2.2 depicts the homepage of Eventful.
- **Lanyrd** [3]**:** was founded in 2010 and provides a social directory of conferences and other professional events. It enables users to enter some conference details such as schedule, location and speakers. Users can be identified through their Twitter [4] or LinkedIn [5] accounts and they are invited to list the conferences to which they will attend. Figure 2.3 depicts the homepage of Lanyrd.

2. http://www.eventful.com
3. http://www.lanyrd.com
4. http://www.twitter.com
5. http://www.linkedin.com

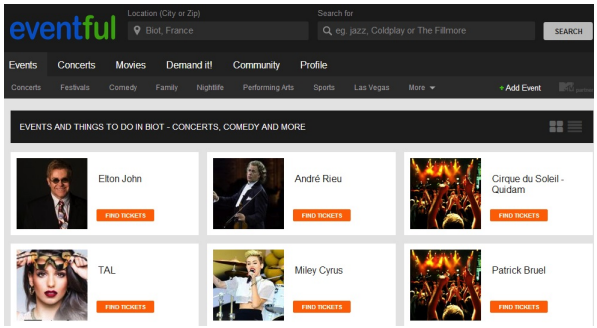

Figure 2.2: Eventful homepage

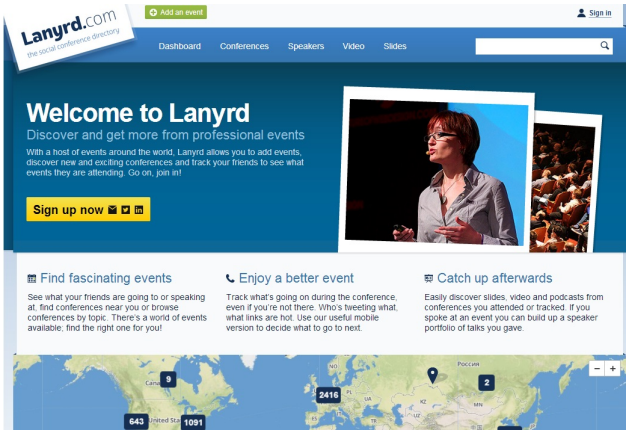

Figure 2.3: Lanyrd homepage

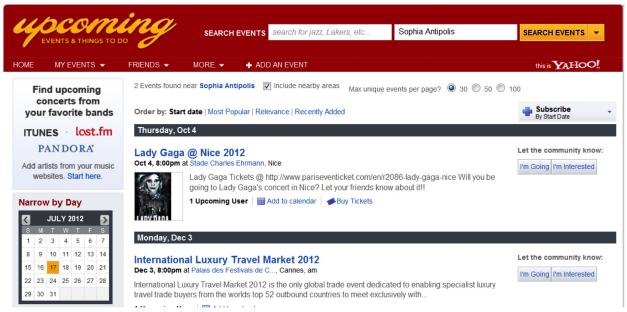

Figure 2.4: Upcoming homepage

- **Upcoming:** was another event-based service launched in 2003, and acquired by Yahoo! in 2005 but retired in 2013. It was a competitor of Eventful and offers similar functions. Upcoming hosted different types of events such as conferences and art exhibitions along with useful details including time, location, etc. Users can create and manage events and have a "friend" relationship with each other. Figure 2.4 depicts the homepage of Upcoming.

Media Websites

Many participants use social media platforms to engage in discussions and share media captured during events. In the following, we describe some popular media sites used in this thesis.

- **Flickr**[6]**:** is one important photo and video sharing website founded in 2004. The site claimed 6 billion of hosted photos in 2012 witnessing a significant growth in the past few years. It provides rich metadata about photos that have been widely exploited by research community. These metadata describe several attributes such as title, description, uploading time, geo-coordinates, tags, etc. One popular attribute is the so-called "machine tag" or "triple tag" which is exploited in this thesis. It is based on a special syntax which is meaningful to be processed by machines. It comprises three parts: (1) the namespace to denote the classification of a tag ('flickr', 'geo', etc.); (2) the predicate to represent the property of the namespace ('latitude',' 'user', etc.); (3) and the value of the tag. For instance, "geo:lat=25.070173" is a tag for the geographical latitude attached to the value of 25.070173.
- **Twitter:** is the most famous micro-blogging service emerged in 2006. The site claimed 200 million of active users in February 2013. It allows users and organizations to publish text messages limited to 140 characters. These text messages are called "tweets" or more generally "microposts" and the subscribers are called "followers". Users can also reply or "retweet" messages. Retweets can be comparable to forwarded emails indicated by "RT" character. Moreover, a message can contain a sort of tag preceded by the hash character which is known as a "hashtag" (e.g., #tag). Finally, users have the option to follow other users, thus becomes a "followee" of them.

2.1.3 Exploratory User Study

The initial motivation behind this thesis lies in an exploratory user study conducted by Fialho et al. [40]. The goal is to understand the event-related activities (e.g., searching, attending, sharing) and to collect insights about existing Web-based technologies. This study consists of a user survey completed by 28 participants and two focus-group sessions (10 and 25 participants). The questions were elaborated to assess the perceived benefits and drawbacks of using: event directories, media directories, social networks, and a merger of these services. In the following, we describe some highlights of this survey:

- **Finding and attending an event:** Participants reported to discover events mainly through invitations, recommendations, friends' posts or some traditional media (e.g., news articles, ads, etc). They also refer to previously attended events or venues to find new events, and they use search engines particularly when they knew what to look for. Moreover, it is found that decision about attending an event seems to prioritize some

6. http://www.flickr.com

significant constraints such as time, location and price. Social information about which friends will attend an event has also an important role in decision making. Other additional details appear to have slight influence like the case of subjective factors (type, topic, performer). To share their experiences, participants tend to use media directories and social networks by posting comments, photos and few videos.

- **Use of social directories:** According to participants, an event directory or website is the best source to provide a general overview of an event context within a single channel. It also enables a user-friendly event exploration from various views (what, when, where) along with other features (e.g., tickets, comments). However, it appears that the information perceived are often incomplete and insufficient for decision support (lack of media and geographic map). To overcome this issue, media directories have been considered as one valuable outlet that better illustrate the event context based on visual information. Similarly, social networks seem to be precious channel to enrich the event context by some features such as attendance, opinions and invitations. Besides, some other functionalities have been mentioned to be desirable for reducing the information overload. For example, it is of great importance to support recommendation of events based on friend's attendance and user interests. Another functionality is to better visualize events by improving search features (e.g., geographic map) and enriching descriptions (e.g., price, attendance).

- **Recapitulation:** To sum up, lack of coverage of event directories and frustration of being locked in isolated sites are the recurrent issues perceived during the study. Participants recognized that there is a need to access several social channels to gather information. One participant reported "*I don't like always having to go from one site to another to find out things about the event*". Overall, users advocate the need for a single source to explore events, not by creating another information source, but by centralizing all available information leading to broader coverage. In addition, they highlight the role of photos and videos to provide powerful means of identifying several event characteristics. Media is thus useful to convey the experience and to support decision making. Nevertheless, a common concern of information overload suggests that the environment should avoid cluttered information and provide advanced browsing and personalization mechanisms. Motivated by this study, we decided to build a platform based on the Semantic Web technologies in order to integrate information spread in many silos, and to improve event discovery and content personalization.

2.2 Events in Research

In the last few decades, a growing corpus of research has been centered on the notion of event. Such particular attention sheds light on the inherent complex nature of events. This

is behind the fact that even the definition of what an event is fails to reach a real consensus. Recently, the growth of social networks along with the technological improvements that made connected devices easy to use, made the user-contributed Web a primary source of information about any kind of real world happening. Studying events on the Web has been the subject of an attractive diversity of research works. Summarizing the various challenges surveyed in these studies, we discern three major key aspects that will drive our strategy to design a reliable system.

First of all, an event is an entity that handles in essence contextual dimensions, each of which is related to one attribute such as time, location, topic and participants. This multi-faceted aspect has driven the design of many programs which aim, for example, to detect events from social media [2, 140] or to explore meaningful relationships between them [25]. Recently, a research study proposed by Ramesh Jain [56] has explored the multi-dimensionality to introduce a coherent definition of the so-called "*The Web of events*". Indeed, this term has been conceived as the Web in which nodes represent events having informational and experiential attributes with links describing its structure and relationships. Informational attributes provide descriptive metadata of events including title, location, participants and so on. Experiential attributes describe the sensory data highlighting the event experience such as image and video. Various links can exist in the Web of events such as the one which connects events with experiential attributes. Other links may capture the natural relationships that exist among events such as identity, temporality and causality. Among all these dimensions, it appears that the temporal one has received a substantial attention in research. Several studies in TDT field have been based on time series analysis of media content to identify events. A typical example is the work of Weng et al. [140] that considers an event as a burst of words in a specific temporal window. Another earlier work proposed by Allen et al. [4] in AI field provides a logical model known as Allen's interval temporal algebra proposed to represent the temporal relationships between pairs of events.

Beside to multidimensionality, the second key aspect of events is the short lifetime. Broadly speaking, an event is an ephemeral item that only exists between two time instants. This period seems also to be correlated with peaks of user activities in social networks where people engage in discussions about this event. Such transiency has constrained the design of many real-time systems which should support high scalability and online processing of streaming data. For example, Sakaki et al. [124] proposed a real-time system to identify earthquake events in Twitter. Becker et al. [8] used an online clustering technique to detect in real-time groups of topically similar tweets that correspond to events. Recommender systems have been also perceived to suffer from the fleeting nature of target items. They can only acquire a limited history about event participation which induces highly sparse rating data. This is a well-known problem in recommendation which appears when an item has not received enough ratings to be meaningfully used. Such items require an advanced system such as the one proposed by Cornelis et al. [26] based on the hybridization of existing and

popular recommendation techniques (e.g., collaborative filtering, content-based).

The third key aspect is the social information that an event holds. In reality, people regularly attend various events or share their experiences, thus forming a dynamic space of rich social interactions. As such, social networks can be directly constructed from event-centric activities which can be offline in the physical world or online on the Web. This so-called *event-based social network* has been studied in some research works. For example, Liu et al. [86] proposed a formal definition of an event-based social network, and they extensively studied its underlying properties along with community detection and information diffusion. Liao et al. [85] used them to reveal the latent social relations between users which are then exploited in event recommendation.

2.3 The Semantic Web

The current Web, as introduced by Tim Berners-Lee in 1989, is a huge information space mostly represented in the form of interlinked HTML documents. While the interpretation of the information is delegated to human beings, computers serve merely as storage and communication platform. This fact prevents machines from achieving many tasks based on automated data processing such as search and query answering. Since it has been designed for human consumption, the Web still needs a high human involvement to interpret, combine and categorize data. To overcome this limitation, many efforts have been spent in some fields such as Information Retrieval, Machine Learning and Natural Language Processing (NLP). They have produced complex systems trying to automatically extract meaning from unstructured data. Typical examples are the search engines such as Yahoo [7] and Google [8]. They mainly rely on NLP routines to index data without any knowledge about the meaning of terms and the relationships between them. Although the emergence of search engines was a success for the Web, there is still a semantic gap between what the machine understands and what the user knows about the data [94]. This is where the Semantic Web intervenes trying to fill the knowledge gap. In this context, Tim Berners-Lee et al. [11] provide the following definition:

> *"The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation".*

How to expand the Web of documents for users with the Web of information for machines is the vision of the Semantic Web. The objective is to automate the human data processing without using full-fledged NLP or reasoning methods by giving meaning to resources and linking them. In the Semantic Web, an intelligent document has awareness about its own content making it exploitable by automatic process. This way enables machines to answer

7. http://www.yahoo.com
8. http://www.google.com

complex queries which are currently not possible without human involvement. For example, one user may want to find an event that will take place next weekend in Nice, covering one specific topic and having a suitable price. For this, he/she currently needs to trawl through various websites and look at different fields (e.g., location, topic, price). On the contrary, the answer in the Semantic Web can be provided by an intelligent Web agent that decodes the query and exploits Linked Data to deliver relevant information. In order to realize such a vision, a series of technologies and standards have been proposed. They provide ability to add meaning to the Web content and to represent it in a machine understandable format. In the following, we describe some of these standards along with the trend of Linked Data.

2.3.1 Resource Description Framework (RDF)

Resource Description Framework (RDF) [77] is a recommendation of the World Wide Web Consortium (W3C) that describes the Web resources. In the Semantic Web, a resource is anything that has an identity and it can be a person, document, image, location, etc. Each resource is assigned a Universal Resource Identifier (URI) [10] which is formatted as string and identifies an abstract or physical resource. A common type of URI is the Universal Resource Locator (URL) used to identify resources located on the Web. RDF is originally designed as a simple metamodel for describing information in a direct graph with labeled nodes and arcs. In this model, the nodes represent the Web resources and the arcs represent the properties which link together these resources. Note that a property can be a specific aspect, characteristic, attribute, or relation used to describe a resource [77]. In RDF, resources can be described and linked by a set of statements forming a graph, also known as a semantic network. Each statement is a triple which is usually denoted as <s,p,o> and composed of:

- Subject: the resource which the statement refers to. It is identified by a URI.
- Predicate: describes a property of the subject and expresses the relationship between the subject and the object.
- Object: specifies the value of the property. It can be a resource identified by a URI or an atomic value named literal. Note that a literal can be plain or typed. A plain literal is a string combined with an optional language tag (e.g., "thesis"@en). A typed literal is a string associated with a datatype URI (e.g., "0.52"^^datatypeURI). The datatype URI specifies the datatype of the literal which can be integer, float or date, as defined by the specification of XML Schema Datatype [9].

Figure 2.5 depicts an example of RDF graph-based representation about “France” which is identified by a URI on the Web. Note that this URI identifies a subject resource which is assigned the type *Country* and has *France* as label.

Several methods exist for serializing the RDF data model. The most common format is RDF/XML. There exists other text-based formats introduced by W3C such as Turtle [10] and

9. http://www.w3.org/TR/xmlschema-2
10. http://www.w3.org/TeamSubmission/turtle

Figure 2.5: Example of RDF representation about France

N-Triples [11] which are easier to read than RDF/XML. To query the RDF graph, W3C has defined a query language called SPARQL [12]. It contains triple patterns along with their conjunctions (e.g., logical "and") and disjunctions (e.g., logical "or"). It also supports extensible value testing and constraining queries by named RDF graph.

2.3.2 RDF Schema

RDF is a very simple and flexible data model that allows users to describe resources using properties and values. However, it does not provide means to define vocabularies and to specify domain specific classes and properties. Hence, other terms are needed to describe the classes of resources and the relationships between them. As a solution, an extension of RDF called RDF Schema or RDFS [18] provides a basic vocabulary to interpret RDF statements. RDFS vocabulary simply describes taxonomies of classes and properties and defines very basic restrictions. In RDFS, URIs have `http://www.w3.org/2000/01/rdf-schema#` as a namespace conventionally associated with the prefix *rdfs:*. In summary, (1) A resource is an instance of one class (*rdfs:class*) or more classes where classes are organized in a hierarchy using *rdfs:subClassOf* property; (2) Properties have *rdf:Property* as class and are organized in a hierarchy using *rdfs:subPropertyOf*. Some restrictions on properties are specified such as *rdfs:domain* to define the class of the subject, and *rdfs:range* to define the class of the object.

2.3.3 Ontology Vocabulary

RDF and RDFS both have limited expressivity. While RDF describes a simple way to represent structured data, RDFS provides only basic hierarchies associated with simple restrictions. However, there is a need for more expressivity to be able to define a formal explicit description of concepts in some complex domains. Therefore, the concept of ontology has

11. `http://www.w3.org/TR/n-triples`
12. `http://www.w3.org/TR/rdf-sparql-query`

been adopted as an extension of RDFS with more expressive constructs. Ontology was originally defined by Artificial Intelligence (AI) community as explicit formal specification of a conceptualization in domain of interest [45]. It typically describes the concepts of the domain and the semantic interconnections that hold between them, along with some logic and inference rules. In general, ontology is the reflection of a shared and common understanding of a domain that can be communicated between people and/or machines. Given the different websites containing heterogeneous data, the use of common ontology will enable Web agents to have a unified view on data and to answer complex queries. In the following, we list some core elements of an ontology:

- Class: defines a concept, type or collection in a specific domain. It groups objects that share some properties and organized into a hierarchy. For instance, in a university domain, the class Student is more specialized than the class Person.
- Individual: also known as an instance or object and it represents a member of a specific class. For instance, *Nelson Mandela* is an instance of the class Person.
- Property: is a binary relation that describes how classes and individuals can be related to each other. There are two types of property: a datatype property which associates individuals with literals, and an object property which connects between individuals of two classes. For example, *ex:livesIn* is an object property that relates an instance (e.g., John) from the class Person to an instance (e.g., London) from the class Location.

To author an ontology, the Web Ontology Language (OWL) [44] is the current markup language endorsed by W3C. Compared with RDF and RDFS, OWL defines a vocabulary with additional formal semantics. It provides more relations between classes (e.g., *disjointWith*), logical properties (e.g., *intersectionOf*, *sameAs*) and enumerations (e.g., *oneOf*, *allValuesFrom*), among others.

2.3.4 Linked Open Data

The Semantic Web is predicated on the availability of large amount of structured RDF data, not in isolated islands but as a Web of interlinked data. A major milestone to realize this vision is the Linked Open Data (LOD or Linked Data) project [29] that connects RDF datasets on a large scale. Linked Data captures a growing knowledge from various domains forming an open "Web of Data" freely available to access, download and use it. Today's Linked Data comprises billions of RDF triples including millions of links between different datasets. Formally, Linked Data has been defined as about "data published on the Web in such a way that it is machine readable, its meaning is explicitly defined, it is linked to other external datasets, and can in turn be linked to from external datasets" [14]. Linked Data follows the principles outlined by Tim Berners-Lee to publish information on the Web, which are:

- Use URIs as names for things

- Use HTTP URIs so that people can look up those names.
- When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL)
- Include links to other URIs. so that they can discover more things.

Overall, these principles stress on the accessibility and the linkage of data that adhere to the architecture and standards of the Web. Figure 2.6 shows the significant number of published datasets in 2011, covering information from diverse areas such as encyclopedic, government, geographic, entertainment, publications and so on. For instance, DBpedia [13] is one of the largest RDF repository in the Linked Data focusing on extracting multilingual knowledge from Wikipedia. At the time of writing, the English edition of DBpedia consists of 470 millions RDF triples that describe 4.0 million things covering a wide range of topics, and contains 45 million RDF links to several hundred external datasets.

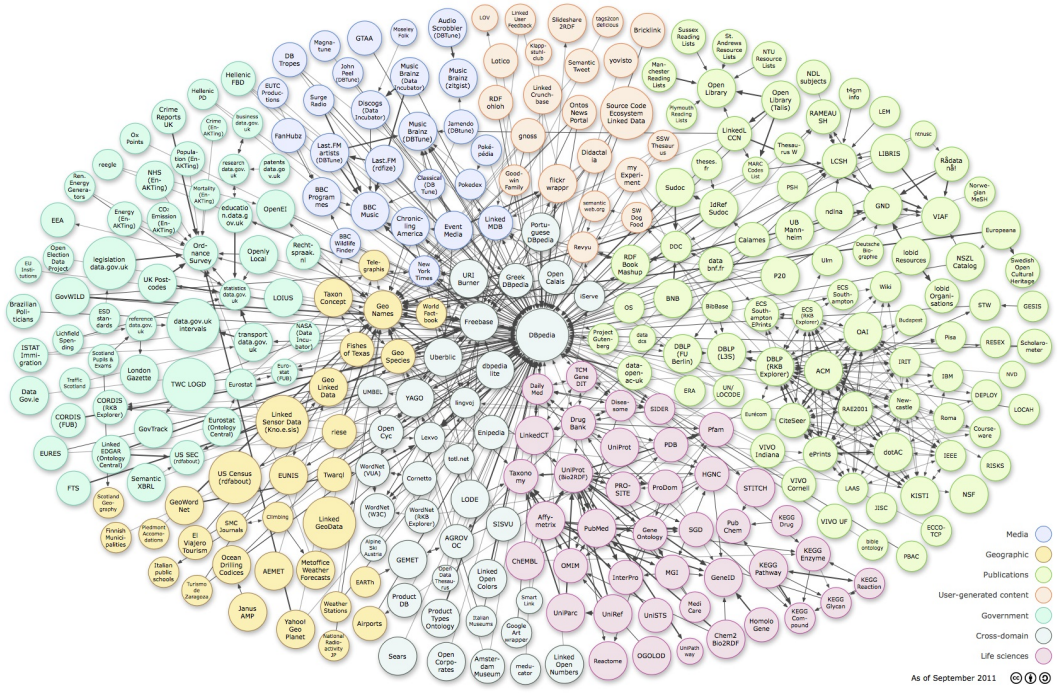

Figure 2.6: Linked Open Data (LOD) Cloud in September 2011

Client applications can access and use RDF links to navigate between datasets and to discover additional information. In order to be part of Linked Data, datasets need to create links to other datasets. To cope with the large amount of instances, it is a common practice to draw on automated or semi-automated tools which are able to automatically reconcile data and generate links between datasets. Yet, this is still a challenging task and significant

13. http://www.dbpedia.org

research efforts have been devoted to address it.

2.4 Evaluation Metrics

In this section, we overview the mostly used evaluation functions in this thesis namely, Precision, Recall and F-score. These measures are widely exploited in data reconciliation field. For a reconciliation task, results can be classified into 4 categories which are: *true positives (tp)*, *true negatives (tn)*, *false positives (fp)* and *false negatives (fn)*. The terms *positive* and *negative* refer to the system's prediction, and the terms *true* and *false* refer to whether this prediction is correctly corresponding to the ground truth or not. Precision computes the percentage of correctly matched reference pairs (tp) over all matched reference pairs (tp and fp) (Equation 2.1). Recall computes the percentage of correctly matched reference pairs (tp) over pairs of references in the ground truth (tp and fn) (Equation 2.2).

$$Precision = \frac{tp}{tp + fp} \tag{2.1}$$

$$Recall = \frac{tp}{tp + fn} \tag{2.2}$$

In practice, F-score is also popularly used and it combines both precision and recall as follows:

$$F\text{-}score = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

2.5 Conclusion

In this chapter, we have first reviewed several definitions given to the notion of event and we have adopted an interoperable definition that describes essential aspects. Then, some popular social websites hosting event related data have been described. The drawbacks perceived by people to use these websites particularly motivated us to carry out this work. Finally, we have detailed the fundamentals of the Semantic Web as well as some evaluation criteria used in this thesis.

Part I

Structuring and Linking Event-centric Data on the Web

Overview of Part I

In Part I, we prpose to develop a framework that retrieves and aggregates event-centric data derived from event directories, media platforms and social networks. We capitalize on Semantic Web technologies to ensure a flexible and large-scale integration of disparate data sources, some of which overlap in their coverage. The ultimate goal is to provide a Web environment for exploring events associated with media and for discovering meaningful connections between them.

In Chapter 3, we present the different steps involved in building a new large dataset called EventMedia which is composed of events descriptions associated with media. These steps include data aggregation and structuring into a unified knowledge model using ontologies. One fundamental requirement is to set a flexible architecture, so that it can support the capability to easily add further event and media websites.

In Chapter 4, we focus on the fourth element of the Linked Data principles which is to link data together. The goal is to explore the implicit overlap of disparate data sources trying to overcome data heterogeneity. We mainly investigate the following questions: what heuristics are suitable to reconcile semantic event-centric data? and how to reconcile structured events with unstructured media?

CHAPTER 3

Data Aggregation and Modeling

Along with the advent of Web 2.0, a substantial amount of high-demand information continue to be created and expanded over multiple websites. In particular, information about events, illustrative media and social interactions are in constant growth. However, this information is often incomplete and locked into the sites, providing limited event coverage and no interoperability of the description. Integrating these distributed data sources into one unified platform is a key factor to enable rich representation of events and to foster search capabilities. One major concern is how to flexibly integrate data and easily add data sources. The goal is to achieve data integration in reasonable level of efforts and to face the dynamics of Web 2.0. In this chapter, we present our framework to integrate data where we ensure a certain level of flexibility to add further websites. Moreover, we explore the intrinsic connections between events and media based on explicit metadata.

3.1 Data Aggregation

In this section, we overview the definition of a Web service. Then, we describe how data from event and media Web services has been collected and interlinked in a flexible way.

3.1.1 The Notion of the Web Service

The notion of the Web Service has been defined by the W3C as "a software system designed to support interoperable machine-to-machine interaction over a network." [17]. It provides an application-programming interface (API) which describes a specification of remote request-response calls that could be consumed by other systems. In this context, a Web service is sometimes considered as a synonym of a Web API. In Web 2.0, the most common API is based on REST architecture in which "the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of *stateless* operations" [17]. REST stands for Representational State Transfer, and it has emerged in the last few years as a predominant design model of a Web service. It has been introduced in 2000 in the doctoral dissertation of Roy Fielding, one of the principal authors of the HTTP specification. REST strictly refers to a collection of network architecture principles which outline how resources are defined, addressed and transferred over HTTP. With REST, each resource is referenced with a global identifier (e.g., URI in HTTP). To interact with a resource, an application needs to know the identifier of the resource, the action required and the format of

the response. Most of existing Web APIs are currently based on REST architecture, and they define a set of HTTP request methods, along with associated responses usually serialized in XML and JSON formats.

3.1.2 REST-based Scraping Framework

Web services such as Eventful, Last.fm and YouTube become increasingly important for creating Web content mash-ups. Thus, collecting data from these sites implies the studying of related API specifications which differ in terms of policy, HTTP request methods and response schema. To alleviate this task, one typical solution is to design a unified interface that combines various APIs and manages some tasks such as policy management, requests chaining and merging response schemata. Some tools providing this solution have been emerged with the aim to save developers' efforts. One example is the API BLENDER [42] which is an open-source tool that integrates five websites, namely: Twitter, Facebook, Flickr, Google+ and YouTube. It describes a Web API using a set of JSON objects including the definition of access policies and API methods. For example, the "Policy" object describes the number of requests per hour and the too-many-calls response code. Although API BLENDER supports a high flexibility to collect data, it does not address the heterogeneity of response schemata. Another tool is the media collector developed for MEDIA FINDER application [116]. It enables a parallel key-search over a variety of social networks and exports results into a unified output schema. It is based on the alignment of response schemata into a common one in order to be agnostic of a particular social network. This common schema describes a set of metadata such as url, type (e.g., photo or video), message (e.g., description of media item), etc. However, there is no support of policy management and the response schema provides only very basic information. Given the shortcomings of these tools, there is a need for a novel tool that provides a unified interface and exploits the similarity between the Web APIs. To meet this need, we propose the framework [1] illustrated in Figure 3.1 and composed of two main components: the Unified REST Module and the Scraping Processor.

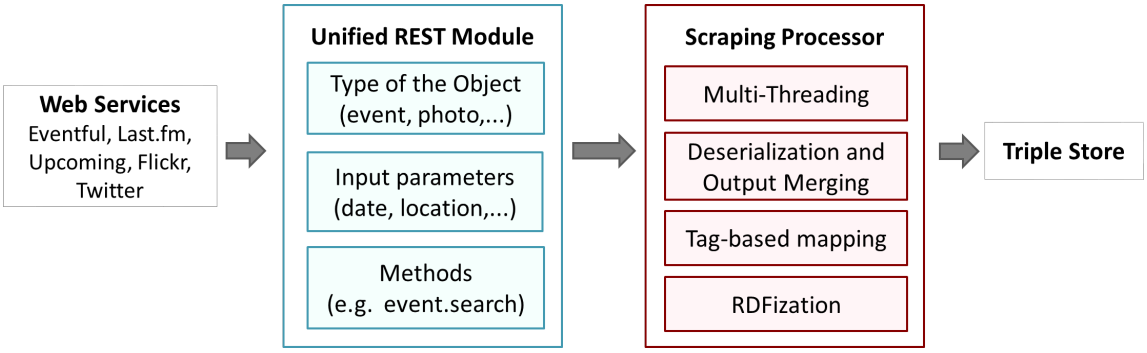

Figure 3.1: Rest-based Scraper Architecture

1. http://eventmedia.eurecom.fr/scrap

• **Unified REST Module:** It is based on a RESTful service that unifies various Web APIs by exploiting their commonality in terms of HTTP methods, objects and input parameters. Each source API (e.g., Eventful API) is associated with a descriptor file serialized in JSON which provides useful information to handle REST requests. More precisely, this file contains global parameters such as API key, API root path and a dictionary of URLs as depicted in Listing 3.1. In addition, the descriptor file contains an array of query objects. Each query object defines a new REST method (e.g., search.events) and maps it with the similar one from the source API (e.g., geo.getevents in Last.fm API). It also defines the mapping between the input parameters. An example of a query object is depicted in Listing 3.2.

```
{
    "APIName": "Lastfm",
    "APIRootURL": "http://ws.audioscrobbler.com/2.0/?method=",
    "APIKey": "c650...",

    "Prefixes":{
                "publisher": "http://www.last.fm",
                "event": "http://www.last.fm/event/",
                "venue": "http://www.last.fm/venue/",
                "agent": "http://www.last.fm/music/"
            }
}
```

Listing 3.1: Global parameters in Last.fm descriptor file

```
"Query": [
          {
            "Type": "search.events",
            "Method": "{0}geo.getevents&api_key={1}",
            "Inputs": [
                {
                    "Name": "Location",
                    "Format": "&location={0}",
                    "Required" : "true"
                },
                {
                    "Name": "LocationRadius",
                    "Format": "&lat={0}&long={1}&distance={2}",
                    "Required" : "true"
                },
                {
                    "Name": "PageNumber",
                    "Format": "&page={0}"
                },
                {
                    "Name": "PageSize",
                    "Format": "&limit={0}"
                }
            ]
          }
  ]
```

Listing 3.2: Query object for collecting events in Last.fm

In order to manage the requests chaining, we first retrieve the description of main elements (e.g., event, photo, video), and then we perform sub-queries to fetch additional information about artists, attendees, etc. Our newly defined REST methods can have as a parameter the list of desirable sources to be queried (e.g., last.fm, eventful, etc.) along with other filters (e.g., category, location, date, etc.). Thus, the user can request in parallel multiple sources into one request. This RESTful service is flexible enough, so that new methods can be conveniently created and a new similar REST-inspired Web API can be simply integrated by adding the associated descriptor file.

• **Scraping Processor:** It has been designed to manage requests and process data. It provides a scraping engine to enable multi-threading, where each new request is associated with a thread instance of scraping process. This engine allows only a limited number of threads in parallel trying to respect the Web APIs limits. Moreover, the Scraping Processor handles other tasks for processing data, starting from JSON de-serialization to RDF conversion and loading into a triple store. More precisely, data retrieved is de-serialized and exported into a common schema providing descriptions of a set of objects, namely event, location, agent, user, photo and video. Then, we employ a tag-based mapping consuming some metadata in order to establish links between events and media (details in Section 3.1.3). We note that our scraping framework is meant to ease the addition of new services for collecting events and media. It also offers other REST methods for monitoring tasks such as tracking or stopping the ongoing scraping processes.

3.1.3 Explicit Linkage of Events with Media

The explicit linkage between resources is straightforward in the presence of shared keys (e.g., ISBN). Thus, we explore the overlap in metadata between some repositories as follows:

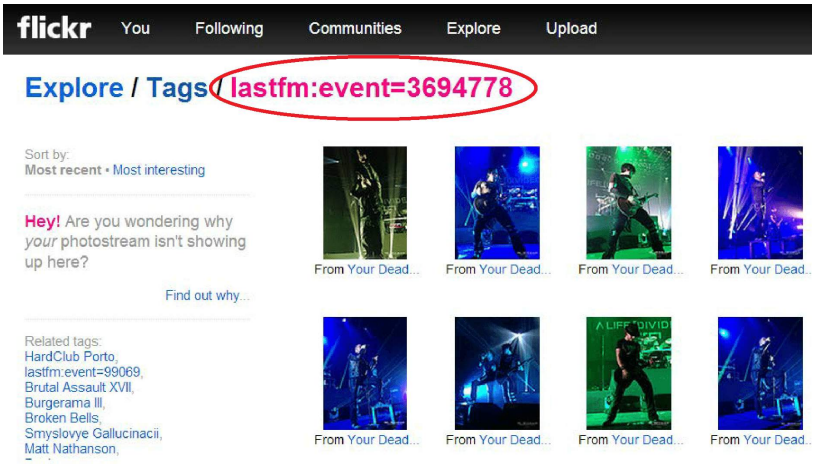

Figure 3.2: Flickr Photo with a machine tag identifying one Last.fm event

1. (Last.fm and Upcoming) with Flickr: Explicit relationships between events and photos exist in Flickr using machine tags such as `lastfm:event=ID` where `ID` is the identifier of a specific event (Figure 3.2). These tags are used as filters when searching for photos. Then, each photo is linked with the event to which it refers.

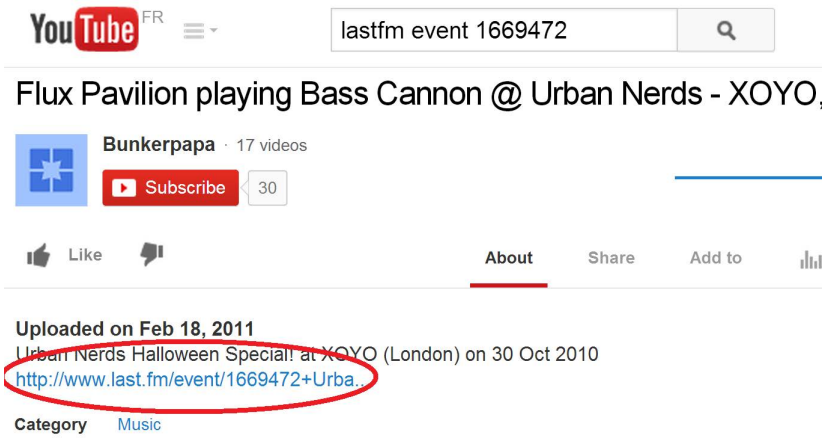

Figure 3.3: YouTube Video in which description includes a Last.fm event URL

2. Last.fm with YouTube: Similarly, explicit relationships between events and media exist in YouTube, where some descriptions of videos contain the URL of the targeted event. Thus, videos can be retrieved by a simple keyword search such as "lastfm event". An event identifier could also be added when collecting videos for a specific event. Figure 3.3 illustrates an example of one video associated with the event resource identified by *ID=1669472* in Last.fm.

Figure 3.4: Lanyrd conference associated with the Twitter hashtag "#uxim2014"

3. Lanyrd with Twitter: We also benefit from the overlap between Lanyrd and Twitter, where a hashtag associates each conference with its related tweets. These hashtags are provided by Lanyrd website as depicted in Figure 3.4

3.1.4 Real-time Scraping

New events are taking place everyday and people keep sharing an ever increasing amount of related media. Such evolution requires a real-time processing that retrieves fresh data and updates the triple store. To achieve this, we developed a live extractor which consumes the feeds provided by some Web services. More specifically, we use the Flickr feeds [2] including the tag "*:event=". Then, a scheduled process reads the feeds every 10 minutes, and trigger accordingly the scraping requests to retrieve the descriptions of events and photos. On an average week, we observe 2000 new photos associated with 160 events (Figure 3.5). Similarly, we also use the Lanyrd feeds [3] that provides fresh conference information including the main hashtag required to retrieve related tweets.

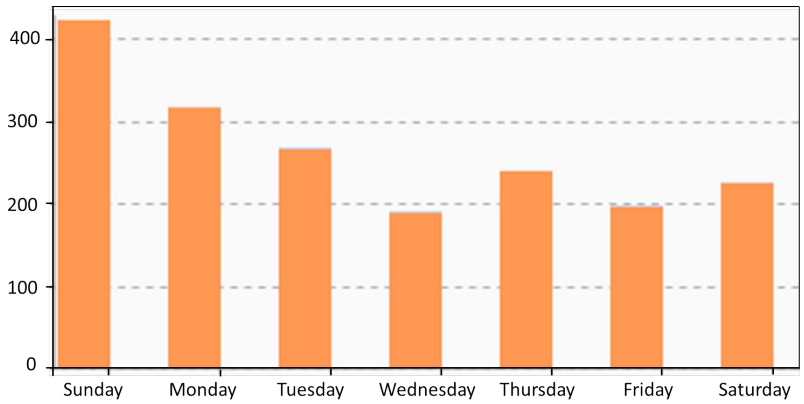

Figure 3.5: Number of photos with the tag "*:event=" posted in Flickr per day

3.2 Web Dashboard

A Web dashboard has been developed in order to offer graphical functionalities that help monitor the scraping task. The dashboard is available online at http://eventmedia.eurecom.fr/dashboard and it is composed of four menus. The *Collect* menu provides practical widgets to help build a query by specifying some parameters as depicted in Figure 3.6. In order to visually track the progress of scraping processes, a timer has been set to query the progress service of our framework and to update accordingly the dashboard (Figuree 3.7). The same timer updates also the log section which provides status messages in different types, namely debug, warning and error. Finally, the dashboard provides *Statistics* menu to show useful information about the dataset such as the number of collected instances per type and per day. Figure 3.8 depicts, for example, the number of events per each category. Technically, the languages used are HTML 5 and Javascript with the simple and powerful library jQuery UI.

2. http://api.flickr.com/services/feeds/photos_public.gne?tags=*:event
3. http://api.lanyrd.com/conferences

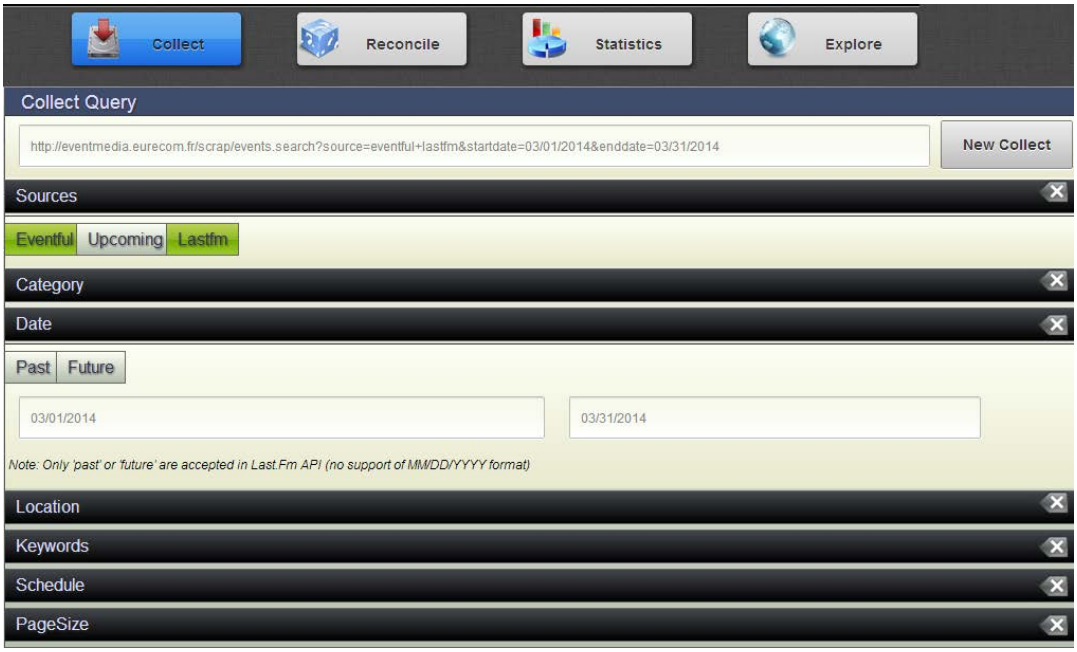

Figure 3.6: *Collect* menu - Building a query to collect events

Figure 3.7: *Collect* menu - Tracking the ongoing process for collecting media

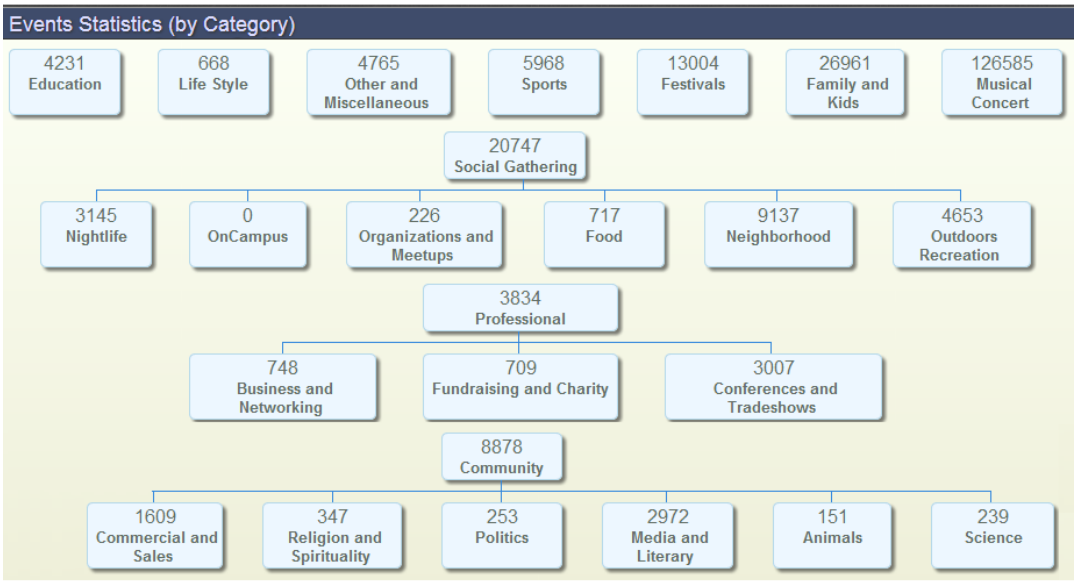

Figure 3.8: *Statistics* menu - Number of events per category

3.3 Semantic Data Modeling

The motivation behind the use of the Semantic Web technologies is their prominent success to provide a flexible support for large-scale data integration. Indeed, a long standing challenge in information systems is to integrate data from distributed, heterogeneous and autonomous data sources. This refers to the problem of combining data spread across different sources, and providing the user with a unified view of these sources. The crucial task lies in forming the mapping between the heterogeneous data sources and the global schemata representing the unified view. Yet data sources, some of them being available on the Web, are autonomously designed and operated. As a consequence, they use different systems (e.g., flat files, relational database), data models and access queries. Combining these distributed sources within one application needs an additional layer. This layer has to integrate data dynamically and facilitate interoperability between different schemata. In research, several integration layers have been proposed as joint efforts from various fields such as Database, Artificial Intelligence and Semantic Web. One solution widely adopted in recent years is the use of ontologies, which is favored in various disciplines such as biology, medicine and e-government [127]. In the context of Web services, Szomszor et al. [135] have shown the efficiency of ontology-based representation to achieve data harmonization when a mismatch occurs between data formats. The underlying goal of using ontology is to provide a conceptual model that can be shared by different applications. There is an emphasis on knowledge reuse and on the creation of common ontologies which can be extended for more specific applications. This has led to different vocabularies which describes resources across various

domains and facilitate semantic interoperability of metadata. In our case, we use ontologies to enable large-scale integration of data provided by event and media websites. But, what vocabularies are suitable for describing events and related entities such as time, location, agent and media? Given the event definition introduced in Section 2.1.1 and the intrinsic connection between events and media, we consider events as:

- A natural way for referring to any observable occurrence grouping persons, places, times and activities.
- Observable experiences that are often documented by people through different media (e.g., videos, photos and tweets).

In order to formalize this definition, we propose the following ontological models that represent events as well as the related media.

3.3.1 Event Modeling: the LODE Ontology

To represent events, we use the LODE ontology [4] proposed in [129]. LODE is a minimal model that encapsulates the most useful event properties, and complies with our event definition. It is not yet another "event" ontology *per se*. It has been designed as an *interlingua* model that solves an interoperability problem by providing a set of axioms expressing mappings between existing event models. Hence, the ontology contains numerous OWL axioms stating classes and properties equivalence between event models such as EO [113], CIDOC-CRM [33] and ABC [74] to name a few. Overall, the goal of LODE is to enable an interoperable modeling of the "factual" aspects of events, where these can be characterized in terms of the four Ws:

- *What* happened
- *Where* did it happen
- *When* did it happen
- *Who* was involved

"Factual" relations within and among events are intended to represent intersubjective "consensus reality" and thus are not necessarily associated with a particular perspective or interpretation. We use the LODE ontology together with properties from FOAF [19], Dublin Core [16] and vCard [53]. Our strategy is to separate events from their interpretations with an emphasis on factual aspects, a design approach different from the other event models.

Figure 3.9 depicts the LODE representation of an event identified by *ID=3163952* on Last.fm. More precisely, it indicates that this event categorized as a *Concert* has been given on *May 21th, 2012 at 12:45 PM* in *The Paramount Theater*, featuring the *Snow Patrol* rock band and having participant named *earthcapricor*. This event also exists in Upcoming directory but with another identifier *ID=3163952*. To sum up, the following types of entities are described as follows:

4. http://linkedevents.org/ontology/

- Event: category, text description, date which can be an instant or an interval represented with OWL Time [51]), location expressed in terms of geographical coordinates (latitude, longitude), venue and finally involved agents (e.g., artists) and attendees.
- Location: label, different address fields such as street, city, postal code and country.
- Agent: label, description (e.g., biography), tags and often a photo.
- User: label, user's real name and an avatar.

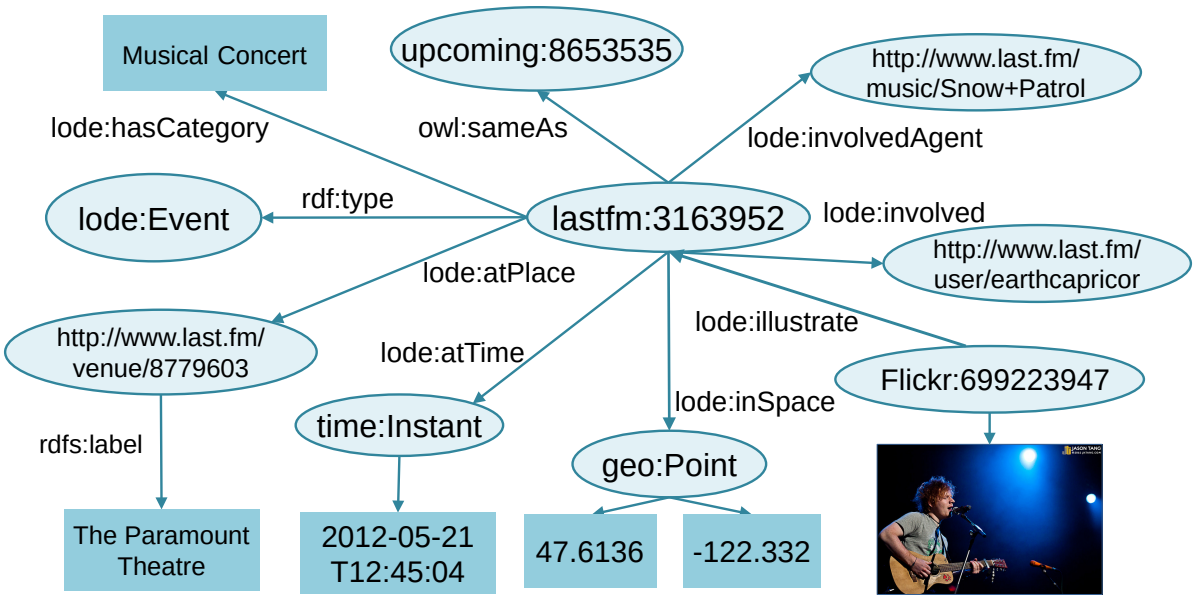

Figure 3.9: The *Snow Patrol Concert* described with LODE ontology

Finally, we propose to organize events in a taxonomy that solves the interoperability of existing classifications. In general, events are categorized in lightweight taxonomies that provide facets when browsing event directories. We manually analyzed the event taxonomies used in various websites, namely Facebook, Eventful, Upcoming, LinkedIn [5], Eventbrite [6] and Ticketmaster [7], and we used card sorting techniques in order to build a rich SKOS thesaurus of event categories. SKOS [95] stands for Simple Knowledge Organization System. It provides a vocabulary to represent knowledge organization systems. Such representations include classification schemes, taxonomies and other structured controlled vocabularies. Our SKOS thesaurus contains axioms expressing mapping relationships between the different event taxonomies on the Web. The event taxonomy in our own namespace is accessible online at `http://data.linkedevents.org/category`. We also show the top categories of our taxonomy in Listing 3.3.

5. `http://www.linkedin.com`
6. `http://www.eventbrite.com`
7. `http://www.ticketmaster.com`

```
<skos:ConceptScheme rdf:about="http://data.linkedevents.org/category">
        <skos:prefLabel xml:lang="en">Event Taxonomy</skos:prefLabel>
        <skos:hasTopConcept rdf:resource="Food"/>
        <skos:hasTopConcept rdf:resource="Family"/>
        <skos:hasTopConcept rdf:resource="Education"/>
        <skos:hasTopConcept rdf:resource="SocialGathering"/>
        <skos:hasTopConcept rdf:resource="Professional"/>
        <skos:hasTopConcept rdf:resource="Community"/>
        <skos:hasTopConcept rdf:resource="LifeStyle"/>
        <skos:hasTopConcept rdf:resource="PerformingArts"/>
        <skos:hasTopConcept rdf:resource="VisualArts"/>
</skos:ConceptScheme>
```

Listing 3.3: Top categories of events in our taxonomy

3.3.2 Media Modeling

In order to represent media, we use two popular vocabularies namely, the W3C Media Resource Ontology [79] and the SIOC vocabulary [12]. The link between events and media is realized through the `lode:illustrate` property.

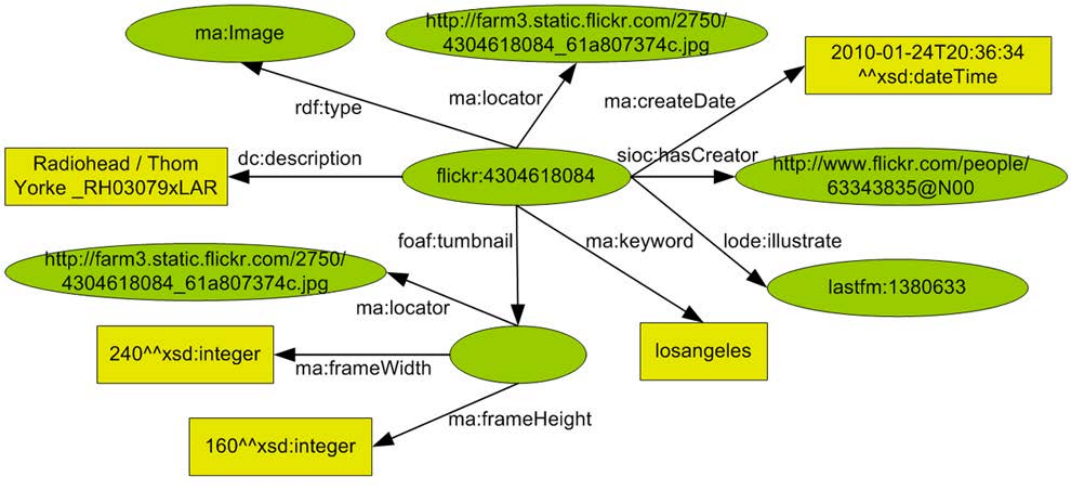

Figure 3.10: A photo taken at the *Radiohead Haiti Relief Concert* described with the W3C Media Resource Ontology

The Media Resource ontology is a W3C initiative that defines a core vocabulary to cover the most common annotation properties of media resources (e.g., image, audio, video). Such properties include different types of metadata such as locator, creation date, genre, rating,

thumbnails, among others. Media fragments can also be defined to have a smaller granularity and attach keywords or formal annotations to parts of a media item. The ontology contains a formal set of axioms that define the mapping between different metadata formats for multimedia. We use this ontology together with properties from SIOC, FOAF and Dublin Core to convert into RDF the descriptions of photos (figure 3.10) and videos.

SIOC stands for Semantically-Interlinked Online Communities. It provides a core ontology about the main concepts required to describe information about online communities (e.g., wikis, blogs). Such information can include post title, author, keywords, date or the full post text in community sites. SIOC becomes a standard way to model the underlying structure of the user-generated content from social media sites. We use it together with Dublin Core properties to convert into RDF the descriptions of microposts as depicted in Figure 3.11.

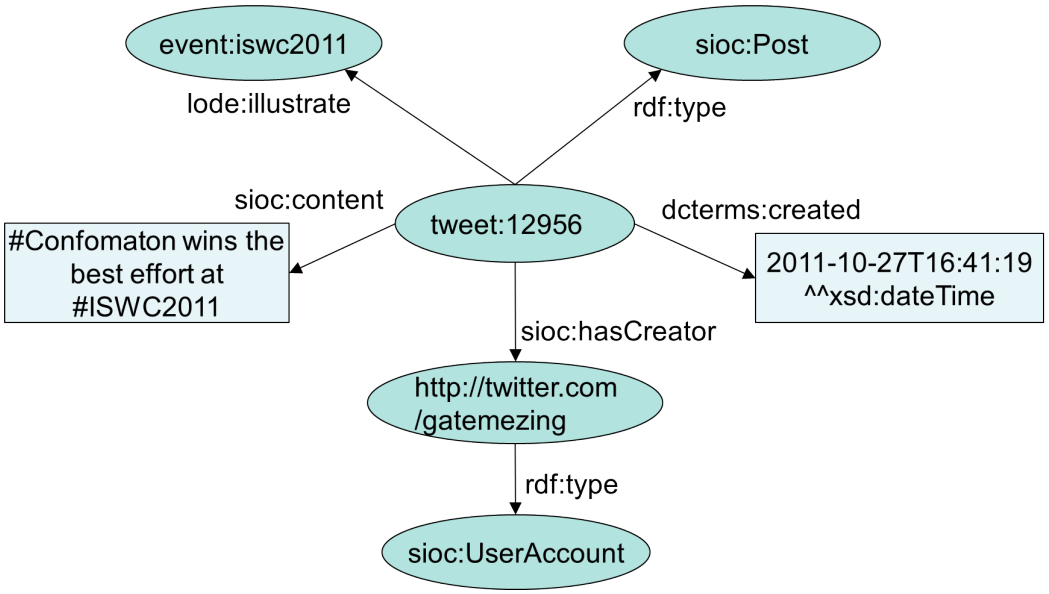

Figure 3.11: RDF modeling of microposts using the SIOC Ontology

3.4 EventMedia

We have collected data from four public event directories (Last.fm, Eventful, Upcoming and Lanyrd), and from three public media directories (Flickr, Youtube, Twitter) [69]. Thus, we built the so-called EventMedia dataset which has become a hub in the Linked Data cloud since September 2010. EventMedia consists of more than 30 millions RDF triples providing descriptions of events and related media based on LODE, Media Resource and SIOC ontologies. We mint new URIs into our own namespace such as for events (`http://data.linkedevents.org/event/`) and agents (`http://data.linkedevents.org/agent/`). All

URIs are Dereferenceable and served as static RDF files serialized in many formats such as RDF/XML, N3 and N-Triples. They are also accessible using a SPARQL endpoint [8] and a RESTful API [9] powered by the Linked Data API (detailed in Section 5.1.2). Table 3.1 provides an overview about the number of resources per type and source, and Figure 3.12 illustrates the main components of EventMedia.

		Event	Agent	Location	Media	User
Event Sites	Last.fm	69,185	81,006	18,653	7,795	213,351
	Upcoming	29,418	78	14,372	29	23,977
	Eventful	84,225	11,226	30,572	15,532	547
	Lanyrd	2,151	-	624	-	-
Media Sites	Flickr	-	-	-	1,879,343	25,219
	Youtube	-	-	-	517	-
	Twitter	-	-	-	1,060,879	267,138

Table 3.1: Number of different resources in EventMedia dataset per type and source

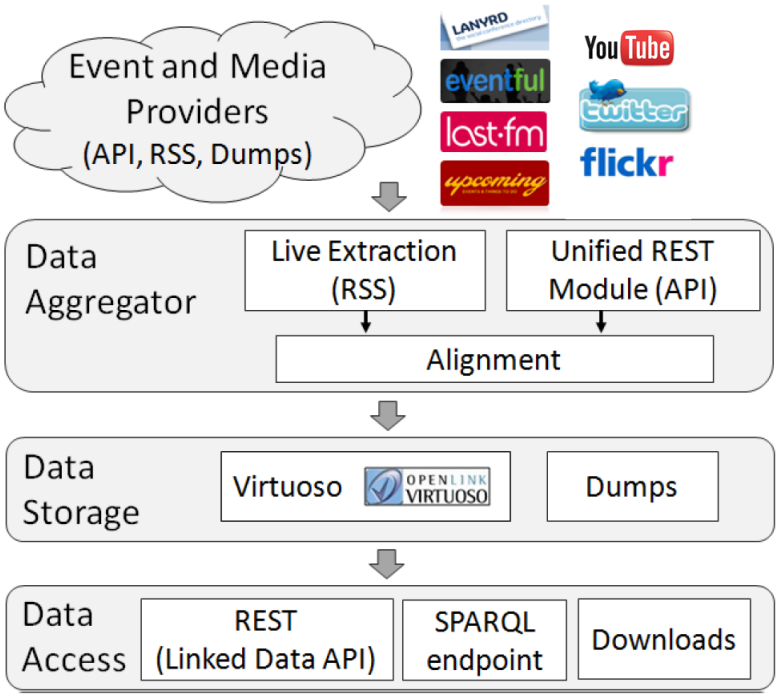

Figure 3.12: Overview of the different components in EventMedia

8. http://eventmedia.eurecom.fr/sparql
9. http://eventmedia.eurecom.fr/rest/{resourceId}

3.5 Conclusion

In this chapter, we have described our scraping framework designed to aggregate data with the aim to ensure a certain level of flexibility. We have also exploited the Semantic Web technologies to integrate at large scale the information hosted by event and media Web directories. As for the semantic modeling, our design is based on the LODE, Media Resource and SIOC ontologies used to describe events and different types of media (e.g., photo, video, micropost). Data collected has been converted to RDF and then stored in EventMedia dataset. Ultimately, we aim at providing an event-based Web environment that delivers enriched views and enhances the event discovery.

CHAPTER 4

Event-centric Data Reconciliation

Multiple websites host a huge amount of user-generated content and they may have an overlap in their coverage. Exploring the overlap of event information from these different sources is a key factor to expand the reach of events at different stages. As has been proved in [40], an event directory contains basic descriptions of events (e.g., time, location), however, often incomplete and locked into the site. Overall, these sites provide a limited event coverage and do not support remembering past experiences. There is a need to improve the event coverage which can be addressed by reconciling event-centric data. Reconciliation will mutually leverage the benefits of each directory and improve the data quality. However, it poses outstanding challenges that fall under the mantle of heterogeneity, a well-known problem in the data integration field. Data integration refers to the problem of combining data residing at different sources, and providing the user with a unified view of these data. One important impediment is to resolve references at the instance level. Such references include "identity link" or any other relationships that express semantic relatedness between two entities. In this chapter, we study the data reconciliation with a focus on two types of semantic relationships: the identity link "owl:sameAs" used to represent a co-reference which determines whether different URIs refer to the same real world entity, and the link "lode:illustrate" which associates events with their related media.

This chapter is composed of two main sections. The first section addresses the co-reference resolution in structured data. The goal behind is to explore identity relationships between events, agents and locations based on instance matching techniques. The second section studies the reconciliation of structured events with unstructured media data. We propose an approach based on NLP techniques to overcome the lack of explicit linkage (e.g., machine tag) and the noise of unstructured media.

4.1 Domain-independent Matching of Events

At the core of our system is the real-time reconciliation framework that aligns every incoming stream of overlapping but highly heterogeneous data sources. This will sustain a continuous content enhancement, a crucial task to cope with the dynamics of social sites. Viewing an event page from one site underlines an incomplete content that needs to be further

enriched. For example, Figure 4.1 shows two Web pages from Eventful [1] and Last.fm [2] describing the concert given by the music band *Coldplay* on *August 8th, 2012* in *Chicago*. We can see that Eventful provides a full text description, while Last.fm gives more details about the event context (e.g., artists, attendees, media). We believe that reconciling event-centric data will mutually leverage the benefits of each service and achieve a better event view.

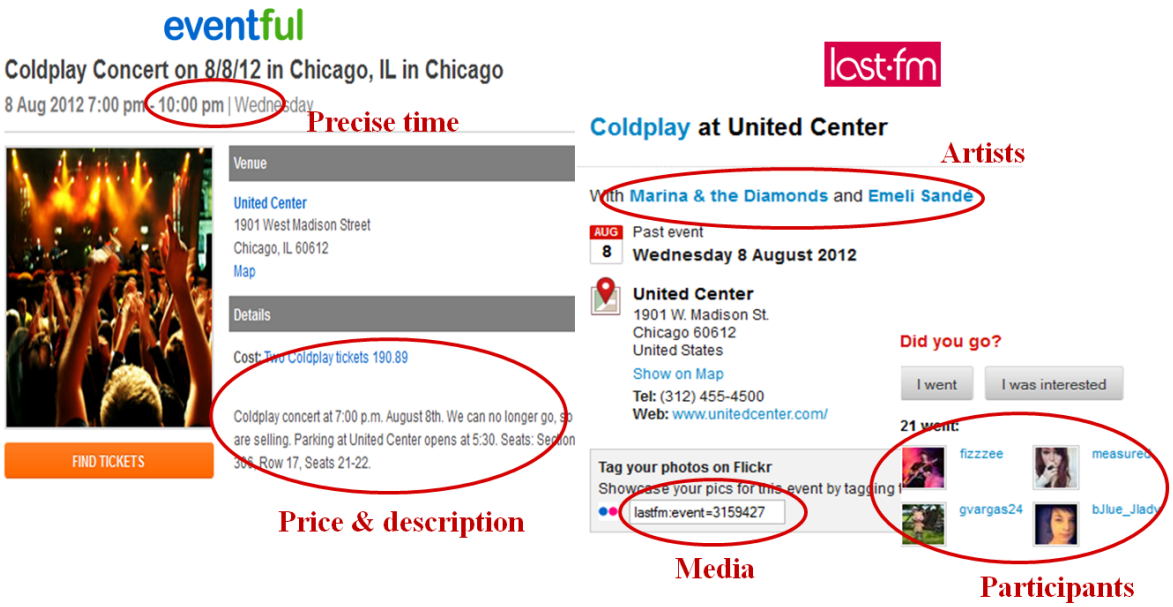

Figure 4.1: Comparison between Eventful and Last.fm Web pages showing a concert of *Coldplay*

4.1.1 Challenges and Related Work

Instance matching has gained importance in the Semantic Web with the emergence of Linked Data cloud. This task is studied under different names such as reference reconciliation and link discovery. One aim is to discover identity relationship among structured data to link same real-world entities using "owl:sameAs" property, which is also known as duplicate detection or identity resolution. Indeed, discovering the identity links is advantageous to ensure higher information coverage and enhance data reuse. Providing an effective support to handle duplicates in the information system is a key factor to improve the data quality. In particular, two main dimensions are directly augmented that are accuracy and completeness [102]. Accuracy is the extent to which data are correct, reliable and free of error. It is usually improved as relying on multiple representations from different sources can highlight some conflicts and thus inaccurate data. On the other hand, completeness is the the extent

1. http://www.eventful.com/events/E0-001-050047180-427
2. http://www.last.fm/event/3159427

to which data are of sufficient breadth, depth, and scope for the task at hand. Intuitively, it can be improved since the multiple representations can cover different properties yielding to more complete description (see the example in Figure 4.1).

We exploit the matching techniques to discover identity links between heterogeneous sources for different instance types, namely event, agent and location. There is, therefore, a need to overcome the diversity of vocabularies used to describe those entities, which can be solved by a domain-independent matching approach. Moreover, data sources can use different ways to represent the same real-world entity. This is due to multiple reasons such as abbreviations, mis-spellings, naming variations over time or different naming conventions. In particular, the likelihood to encounter typographical errors or different values on same property is higher in event websites rather than, for example, in encyclopedic websites. We particularly noticed the presence of some properties semantically dissimilar but holding a latent relationship. For example, the `dc:title` of one Last.fm event is *"Cale Parks at Pehrspace"*, whereas the `dc:title` of the same Upcoming event is *"Cale Parks, The Flying Tourbillon Orchestra, One Trick Pony, Meredith Meyer"* which lists all involved artists rather expressed by `lode:involvedAgent` in Last.fm. Table 4.1 illustrates other examples of syntactically different titles referring to the same event. In research, such heterogeneity has been rarely addressed by the existing matching tools. These tools mostly utilize manual configuration that specifies which properties to be compared or attempt to automatically compare properties having similar semantics (e.g., `dc:title` and `rdfs:label`).

Title 1	Title 2
The Fling	The Fling, So Many Wizards, Tape Deck Mountain
In Space!!!, Hollis Brown, Raccoon City, and Ninefold	In Space!!!, Raccoon City, and Ninefold
The Monolators Don't Dance record release	The Monolators record release, You Me & Iowa, Summer Darling, Correatown
Maailma Kylässä (World Village Festival)	World Village Festival

Table 4.1: Titles related to same events retrieved from different sources

Comparing two instances needs to decide which similarity metric to apply to which data properties with which parameters. In other words, the matching configuration involves selecting the relevant properties to be compared, the similarity function (e.g., Jaro, Leveinshtei) applied to each property and its weight in the final score, the aggregation functions (e.g., average, max, min) and the threshold determining whether a pair of instances should be linked or not. This task is highly dependent on the used schemata, the domain of data and the writing conventions. It is supported by a variety of semi-automatic and automatic matching tools. Among them, Silk [59] draws on a declarative language with which the user must manually define all the parameters of the matching configuration. However, with this manual

intervention, the user mainly follows his intuition and he may skip latent similarities or set unoptimized weights. As an effort to fully automatize the process, Nikolov et al. proposed a system named KnoFuss based on genetic programming which learns the required parameters of the matching configuration. In KnoFuss, the objective function favors the solutions that increase precision and attempts to "cautiously" supervise the recall. Zhisi.Links et al [108] proposed a two-step matching tool that first filters candidates using their labels and then it utilizes a specific semantic metric to compute the final score. In fact, the use of candidate selection mechanism allows the filtering of unnecessary comparisons and thus reducing the computation time. This selection is realized through a blocking scheme which attempts to groups similar instances according to a predefined key. For instance, Zhisi.Links et al. use the entity label as candidate selection key, and thus entities having different labels would not be compared. Yet, this strategy fails in some cases where similar entities may have different labels as depicted in Table 4.1. Song et al. [131] proposed another blocking scheme based on the properties that discriminate and cover the instances. The discrimination of a property reflects the diversity of its object values. A high discrimination means that few instances have the same object values on this property, which can help reduce the unnecessary comparisons. The coverage reflects the number of times a property is used by all instances. The goal is to discover the candidate selection key which is sufficiently discriminating and covering the majority of the dataset. Although this approach is interesting, it is mostly biased to string literals and no consideration of other data types was made. In this work, we exploit this approach and we extend it to other data types such as temporal and numeric.

The problem of reconciling events has been studied in some research works. In philosophy, Quine [112] argued that two identical events refer to the same "something that happens" at the same time and place. This definition has been extensively used in TDT field, where the main goal is to identify events in social media and to cluster together media associated with the same event. Indeed, the event co-reference in TDT is the task of finding clusters that refer to the same event under the mantle of "topic" [9, 2]. Instead, our goal is to create links between structured events based on the four factual aspects (e.g., what, when, where, who). However, there appears to be little work addressing the event reconciliation. Most of existing studies aim to identify a specific type of linkage (e.g., composition, dependency) by discovering temporal or causal relationships [25, 100]. In this work, we aim to discover identity link between events represented in different ways across multiple websites.

4.1.2 Similarity Metrics

As a first step of data reconciliation, we have surveyed the existing similarity metrics tailored to different data types such as string, geo-coordinates and time. Earlier experiments using those metrics underline our need for more efficient metrics to reconcile event-centric data. We therefore propose two similarity functions [68] described as follows:

Temporal-inclusion

Intuitively, two events are similar if they share the same time or temporal interval among other attributes. Thus, one main concern is to consider not only the distance between two date-time values, but also the inclusion of a date-time value in a time interval or the overlap of two time intervals. To consider these facts, we have defined the Temporal-inclusion metric that computes the difference between two instants and detects the temporal inclusion or overlap. Moreover, one event can have distinct time values across multiple websites with few minutes or hours of difference. To overcome this heterogeneity, the Temporal-inclusion metric should tolerate a predefined number of hours θ. As a result, it returns either 0 or 1 to indicate whether there is a match or not on the temporal property.

Given two events (e_1, e_2) which have respectively start dates (d_1, d_2) and end dates (d'_1, d'_2) where $d_1, d_2 \neq 0$ and d'_1, d'_2 can be null, the Temporal-inclusion metric is defined as follows:

$$\textit{Tmp-Inc}\ (e_1, e_2) = \begin{cases} 1 & \text{if } |d_1 - d_2| \leq \theta \text{ where } (d'_1, d'_2) = 0 \\ 1 & \text{if } d_1 \pm \theta \in [d_2, d'_2] \text{ where } d'_1 = 0 \text{ (idem for } d_2) \\ 1 & \text{if } min(d'_1, d'_2) - max(d_1, d_2) \geq 0 \text{ where } (d'_1, d'_2) \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

As an example, the Temporal-inclusion metric returns 1 when computing the similarity between an event which lasts two days from 30^{th} `at 08:00 PM to` 31^{th} `at 07:00 PM of May 2012`, and another event given on 31^{th} `May 2012 at 9:00 PM`. We note that the second event is shifted by 2 hours with respect to the first event, and so we set 2 hours of tolerance.

Token-wise String Similarity

There exists three main categories of string similarity metrics as detailed in Appendix B.1 and consist of character-based distance, token-based distance and hybrid distance. In particular, the hybrid metrics preserve the advantages of character-based metrics used to overcome misspellings and typos, as well as the advantages of token-based metrics used to overcome a swap of words. These hybrid metrics are the most adapted solution to deal with the noise in our dataset. Indeed, the user-generated content in event websites is in general characterized by a high level of heterogeneity. While the user should provide meaningful terms to describe event details (e.g., title, description, location), there are no common conventions or rules that must be respected. The same event on two websites may have two differently written titles due, for example, to syntactic and spelling variations. To solve this heterogeneity, we have defined a new metric called Token-wise which has the same rationale as the Extended Jaccard similarity [139, 5]. The goal is not only to compute the fuzzy overlap between two token sets, but also to penalize the unmatched tokens. Given two strings s and t, and their respective token sets $S = s_1, s_2, ...s_n$ and $T = t_1, t_2, ...t_n$, the Extended Jaccard distance defines the following sets:

- Set of similar tokens between S and T according to a character-based metric sim' (e.g., Jaro, Levenshtein) and a predefined threshold δ.

$$Shared = \{(s_i, t_j) | s_i \in S \wedge t_j \in T : sim'(s_i, t_j) \geqslant \delta\} \tag{4.2}$$

- Set of unique tokens which refer to unmatched tokens for each set S and T:

$$Unique(s) = \{s_i | s_i \in S \wedge t_j \in T \wedge (s_i, t_j) \notin Shared\} \text{ (resp. for } t) \tag{4.3}$$

The Token-wise metric uses those sets and introduces a new parameter α that controls the weight of unmatched tokens. With low values of α, it is sufficient that two strings share in common few tokens to get similar, even if their lengths are disproportionate. In our experiments, we set α equal to $\frac{min(|S|,|T|)}{max(|S|,|T|)}$. This means that more the lengths of strings are disproportionate, lower is the weight of unmatched tokens. Finally, the Token-wise metric is defined as follows:

$$Token\text{-}Wise(s,t) = \frac{2 \times \sum_{(s_i,t_j) \in Shared} sim'(s_i, t_j)}{2 \times |Shared| + \alpha \cdot (|Unique(s)| + |Unique(t)|)} \tag{4.4}$$

For example, let s=*"Treasre Island Music"* and t=*"Island Treasure"* tokenized based on whitespace and forming the following token sets S=$\{$*Treasre, Island, Music*$\}$ and T=$\{$*Island, Treasure*$\}$. Using the Levenshtein distance as character-based function (sim') and $\delta = 0.7$, we obtain the following results:

$$Shared = \{(Treasre, Treasure); (Island, Island)\}$$
$$Unique(s) = \{Music\} \text{ and } Unique(t) = \varnothing$$

$$Token\text{-}Wise(s,t) = \frac{2 \times (0.875 + 1)}{2 \times 2 + 0.66 \times (1 + 0)} = 0.8$$

Table 4.2 shows the comparison of Token-wise with the most used string similarity metrics based on the example given above.

Similarity Metric	Score
Levenshtein	0.25
Jaro	0
Jaccard	0.25
Cosine	0.40
MongeElkan (Levenshtein)	0.66
Token-wise (Levenshtein)	**0.80**

Table 4.2: Comparison of the Token-wise metric with some popular string similarity metrics where s = *Treasre Island Music* and t = *Island Treasure*

.

4.1.3 Domain-independent Matching Approach

Like Zhisi.Links [108], we propose a two-step approach that exploits a blocking scheme for candidate selection in the first step. Then in the second step, we use a training method that discovers the best weights and thresholds of similarity function. As a blocking scheme, we consider the work of Song et al. [131] and we extend it to take into account the different data types. The idea is to compute the correlation and the coverage of properties using various similarity metrics depending on the types of data (string, date-time and numeric). For each type, we compute the similarity between object values as follows:

- **String.** For string data type, we first lowercase the literals and remove the stop-words. To compute the similarity score, we use Cosine distance enhanced by Porter stemming [132] for long strings (e.g., description) and Token-wise distance (Equation 4.4) for short strings. In particular, we use the Levenshtein variant of Token-wise which means that Levenshtein is used as a character-based metric to compute the *Shared* set as illustrated in Equation 4.2.
- **DateTime.** We employ the Temporal-inclusion metric as defined in Equation 4.1.
- **Numeric.** We compute the reciprocal of the absolute value of the difference between two numeric values.

To gain insights into which properties worthy to be compared, we lean on the correlation and the coverage rates measured from labeled data. These rates will also discern the candidate selection key used to maximize the coverage of true matches in the first step of our approach. We take as input two matched instance sets I_s (source) and I_t (target). For each set I_i ($i \in \{s,t\}$), we retrieve the set of literal values L_i associated with each property p_i at a distance $n\text{-}path$ from individuals in I_i. If a property is used more than one time, we group the associated multiple values into one value. The correlation reflects the mutual information in terms of shared values between two properties from the source and the target sets. Each data type (e.g., string, datetime, numeric) of L_i is associated with similarity function $sim_{datatype}$ as explained above. We formalize the correlation and the coverage of each property pair as follows:

$$Corr(p_s, p_t) = \frac{\sum_{l_s \in L_s, l_t \in L_t} sim_{datatype}(l_s, l_t)}{min(|L_s|, |L_t|)} \tag{4.5}$$

$$Cov(p_s, p_t) = \frac{min(|L_s|, |L_t|)}{|I_s|} \tag{4.6}$$

Useless predicates having very low correlation and coverage are filtered out. We consider that the candidate selection key is formed from the predicates which exhibit high correlation and maximum coverage. Then, the remaining properties are used to compute the overall similarity score. We explain how to compute similarity in Section 4.1.5 where we present our experiments. As a training method to discover the weights and thresholds, we employ the Particle Swarm Optimization (PSO) method [62] described in Appendix B.2. It is a

population-based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling. Our choice is motivated by the success of this method in a wide range of optimization problems. In our approach, a particle is represented by a vector of weights and thresholds, and the fitness function aims at maximizing the F-score.

4.1.4 Real-time Matching

One fundamental concern in our system is the real-time reconciliation required to efficiently cope with the growing amount of events daily created on the Social Web. To achieve this, we build a RESTful framework that manages the execution of instance matching on freshly stored data. More precisely, the framework retrieves data from the triple store using two kinds of SPARQL queries. The first query fetches the set of instances of the source dataset filtering data by using the predicate `rdf:type` and the start/end storage dates expressed via `dc:issued` predicate. The second query retrieves, for each instance, a set of candidate solutions from the target source using the predicate `rdf:type` and the candidate selection key. Then, we apply our similarity function between the source instance and the selected candidates, and we create an identity link when the similarity score is above a predefined threshold. The reconciliation can be performed using our Web dashboard at http://eventmedia.eurecom.fr/dashboard/reconciliation.html.

4.1.5 Experiments and Results

In this section, we describe a set of experiments conducted to reconcile the resources of type: event, agent and location. We demonstrate the effectiveness of our approach by means of two ground truths. Statistics about the resulting linksets are accessible on the Web dashboard [3]. The PSO parameters used for the experiments are shown in Table 4.3.

Population size	25
Iterations	40
Acceleration coefficients	c_1=1.494 and c_2=1.494
inertia weight	0.729

Table 4.3: Setting of PSO parameters for data reconciliation

The ground truth used in this evaluation consists of:

- Event ground truth manually constructed and composed of 300 pairs of events occurred in 2009. It matches Last.fm with Upcoming.
- Agent ground truth composed of 2000 pairs which match Last.fm artists with DBpedia. It was constructed from the common links of Last.fm and DBpedia with the open music dataset called Musicbrainz [4].

3. http://eventmedia.eurecom.fr/dashboard/statistics.html (Reconciliation Stats)
4. http://musicbrainz.org

Evaluation of Similarity Metrics

First of all, we start by evaluating the Token-wise and the Temporal-inclusion metrics as follows.

Evaluation of the Token-wise metric: Using the agent ground truth, we compute the similarity scores between 150 pairs of agents' names randomly selected. These names consist mostly of short strings where the longest one contains 28 characters. In record linkage, it has been proved that Jaro distance is efficient to compare names of persons [24]. We decided therefore to compare the Token-wise metric with Jaro, after lower casing strings and removing stop-words. Figure 4.2 shows a scatter plot of the results. The points along the diagonal indicate equal similarity for both metrics. We clearly observe the presence of scores equal to 1 on Token-wise axis, while they range from 0.7 to 1 on Jaro axis. In other words, Token-wise succeeds to discover more exact matches than Jaro.

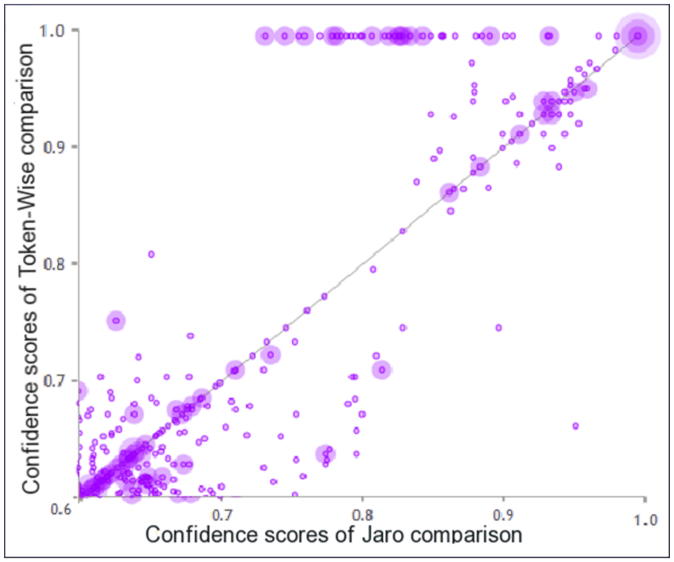

Figure 4.2: Distribution of Jaro and Token-wise similarity scores

Using the event ground truth, we compare different string similarity metrics applied on the titles of events. Figure 4.3 shows the results obtained (Token-wise is denoted as Tw). Compared with character and token based metrics, Token-wise has significantly higher performance. Moreover, it slightly outperforms the hybrid metrics and particularly the Monge-Elkan distance. The difference from the Monge-Elkan metric is that Token-wise best handles the penalization of unmatched tokens as detailed in Appendix B.1.

Evaluation of the Temporal-inclusion metric: Using the event ground truth, we evaluate the Temporal-inclusion metric. We run the experiment by varying the parameter θ (i.e., the number of tolerated hours) in Equation 4.1. To compute the similarity, we use Silk framework [59] where we define a simple linear function that combines all event attributes.

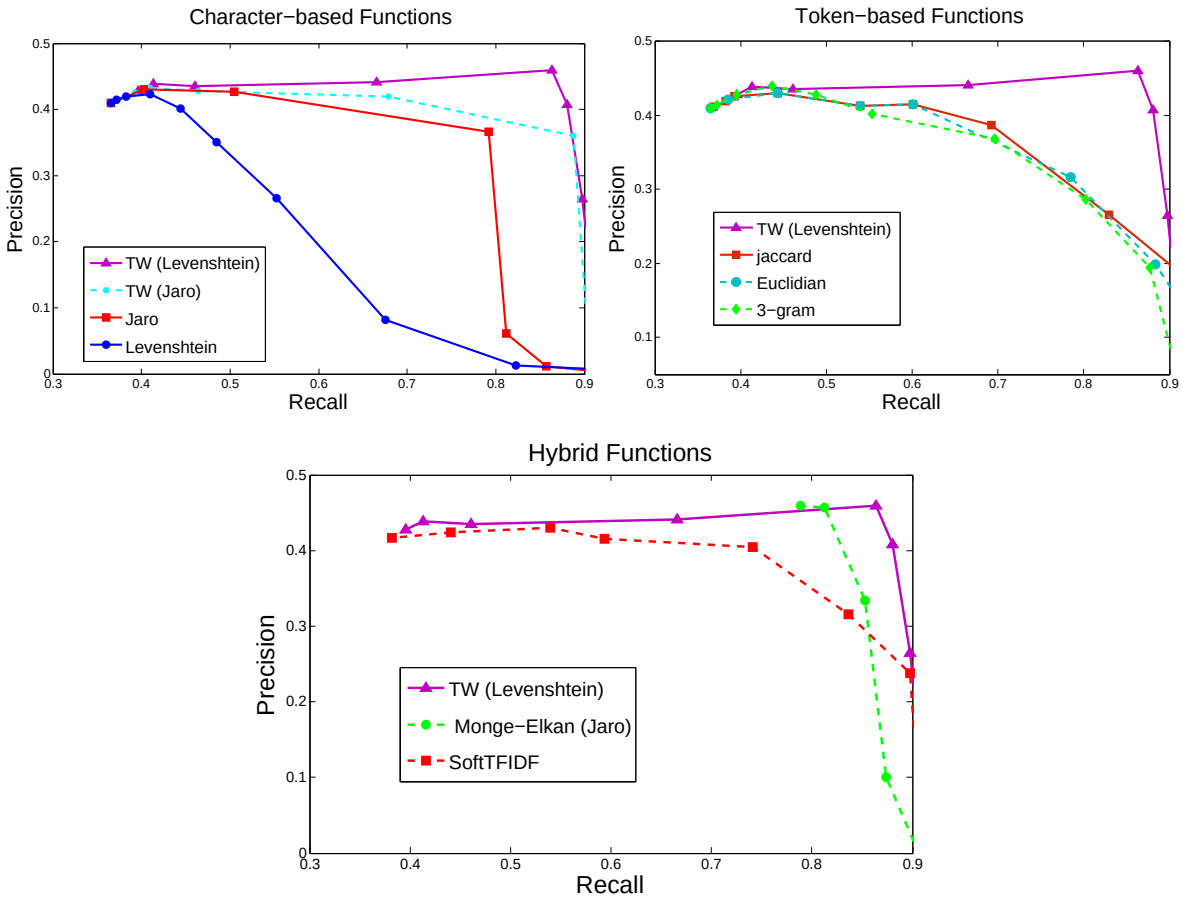

Figure 4.3: Comparison of Token-wise metric with popular string similarity metrics

Table 4.4 shows the results when $\theta = 0$ and $\theta = 24$ and for different thresholds. We can observe a significant increase of the recall when varying the parameter θ especially for high thresholds. This is caused by the different time values of the same real-world event across multiple websites. We often discover a time span between identical events ranging from few minutes to 3 hours. We also detect some events where the exact time is missing and replaced by the midnight time (i.e., 00:00:00). Moreover, one event may last 2 days in one website, while it is represented by two events for each day in another website. This reflects the granularity variation used to express the time across multiple websites.

	$\theta = 0$ H		$\theta = 24$ H	
Threshold	Precision	Recall	Precision	Recall
0.8	0.95	0.44	0.94	0.85
0.75	0.94	0.46	0.93	0.87
0.74	0.78	0.71	0.81	0.92
0.6	0.67	1	0.67	1

Table 4.4: Precision and Recall when $\theta = 0$ and $\theta = 24$ in Temporal-inclusion metric

Evaluation of the event reconciliation

Two events are meant to be identical when there is a mutual agreement in terms of their factual properties, namely: title (what), time (when), location (where) and involved agents (who). In other words, identical events are those which have similar values on their factual properties. In this experiment, we focus on matching events derived from Last.fm and Upcoming directories, and we evaluate it using the event ground truth. We retrieve literal values at a distance 3-path due to the presence of blank nodes. In Table 4.5, we show the correlation and the coverage rates of the most correlated properties (correlation $\geqslant 0.3$) computed on 100 event pairs in the ground truth. This size is sufficient to recognize which properties are worth to be compared.

P_{source}	P_{target}	$Correlation$	$Coverage$
$time_s$	$time_t$	**1**	**1**
$place_s$	$place_t$	**0.80**	**1**
$title_s$	$title_t$	0.59	**1**
$agent_s$	$title_t$	0.53	**1**
$(lat_s, long_s)$	$(lat_t, long_t)$	(0.43, 0.97)	0.92
$agent_s$	$description_t$	0.24	0.48

Table 4.5: Correlation and Coverage rates between properties of 100 events from Last.fm (source) and Upcoming (target)

From Table 4.5, it can be noticed that both *time* and *place* properties exhibit a total coverage and a high correlation, thus their combination forms the blocking key which will be used for candidate selection. Note also that a significant correlation exists between $agents_s$ and $title_t$ corresponding to semantically dissimilar properties, but conveying a connotative relationship. To select candidates for each instance in the source set I_s, we retain the entities from the target set I_t where the combined similarity based on *time* and *place* is greater than a threshold α. The remaining correlated properties are used to find the correct match among the selected candidates. To evaluate our approach, we have conducted various tests comparing the weighted Linear Combination (LC) of similarity measures over all properties (without candidate selection), and the methods using the candidate selection namely, the two-step linear combination *Two-step LC* and the two-step boolean reasoning *Two-step OR*. The Two-step OR method uses the key $(time + place)$ to filter candidates. Then, we assume that it is sufficient whether one score obtained from the remaining most correlated properties is larger than a trained threshold. For comparison purpose, we select KnoFuss [107] which is an domain-independent matching tool based on a genetic algorithm (GA). KnoFuss automatically discovers the components of the best similarity decision including the property pairs, the metrics, the weights and the threshold. We integrated our metrics the Token-wise and the Temporal-inclusion in KnoFuss and we report the results in Table 4.6. We can observe that KnoFuss yields high precision but the lowest recall. This is owed to its strategy

that maximizes a pseudo F-score with bias to precision optimization, given that the cost of an erroneous mapping is higher than the cost of a missing true mapping. It is also shown that the two-step methods produce better results than the pure LC methods due to the effectiveness of candidate selection key to remove noisy comparisons. In particular, the Two-step OR method outperforms the other LC-based methods since it overcomes the lack of coverage of latitude and longitude predicates. Indeed, in the LC-based methods, the weight assigned to the geographical distance is very low due to the limited coverage of these predicates, whereas a high weight was assigned by the OR-based method.

	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**

Table 4.6: Results of different approaches to align events between Last.fm and Upcoming (50% training data)

Finally, Table 4.7 shows the results obtained by the Two-step OR method for different training splits. It is clear that this method achieves a good performance even for a small training set.

	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**

Table 4.7: Results of Two-step OR algorithm for event alignment with different splits of training data

In addition, we have also investigated the reconciliation between event directories and DBpedia. We note that these datasets encapsulate the descriptions of events, but different in terms of data model and data granularity. Indeed, the event directories provide a fine-grained information detailing a spatio-temporal dimension along with other properties. In contrast, DBpedia keeps a general level of description of very famous events without a granular precision about the event time, except for few of them. Considering this fact, we decided to create `rdfs:seeAlso` links between event directories and DBpedia, producing *N-to-*1 mapping instead of 1-*to*-1 mapping. To achieve this, we use SPARQL queries and label-based matching by setting a high threshold.

Evaluation of the agent reconciliation

Linking agents together plays an important role to bring valuable context such as artists' discography, fine detailed biography and illustrative photos. We reconcile agents derived

from event directories, and then with DBpedia. Since the agents' names exhibit the highest correlation and the total coverage, we consider each name token as a blocking key to fetch candidate solutions. In this context, the key challenge widely investigated in the literature is to resolve the naming conflicts. This problem appears when two different persons have the same name. Thus, we invoke additional information using the fairly correlated properties such as `dc:subject` and `dc:description`.

P_{source}	P_{target}	*Correlation*	*Coverage*
$label_s$	$label_t$	0.69	1
$subject_s$	$genre_t$	0.52	0.90
$description_s$	$comment_t$	0.35	0.98
$description_s$	$label_t$	0.19	0.90

Table 4.8: Correlation and Coverage rates between agent properties from Last.fm and DBpedia

Table 4.8 shows the correlation and the coverage rates measured on 100 agent pairs between Last.fm and DBpedia. In this experiment, we point out many correlated properties having the same meaning since DBpedia relies on different vocabularies (DBpedia ontology, FOAF). For instance, a person name in DBpedia is represented by three properties `rdfs:label`, `foaf:name` and `dbprop:name`. Hence, we manually select the most correlated properties. Using a ground truth of 2000 agent pairs from Last.fm and DBpedia, the Two-step OR method achieves the best performance with F-score equal to 0.98 (precision=0.99, recall=0.98).

Evaluation of the venue reconciliation

Venue reconciliation was particularly straightforward due to the consistent and complete description represented by a set of fields such as address, geo-coordinates, city, postal-code and country. First, we reconcile venues retrieved from event directories, and then we align them with external directories such as Foursquare [5], Here.com [6] and DBpedia. There is no a ground truth to evaluate this task, but we found that the identical instances checked on the fly are correctly matched. Moreover, a significant number of venues have been reconciled especially with the Foursquare directory.

Evaluation of the real-time reconciliation

To ensure the real-time processing, we create a scheduler that executes two successive tasks every 10-minutes:

1. The first task enables to fetch new photos in Flickr feeds (size of 20 items) and trigger accordingly the scraping requests to retrieve photos and events descriptions.

5. http://www.foursquare.com
6. http://www.here.com

2. The second task aligns the freshly stored data with various sources by invoking the reconciliation framework using REST requests.

To evaluate the real-time matching, we take a sample of data collected during 3 days and we compute two measures, namely: the storage interval which is the difference between the time data is uploaded in Flickr and the time data is stored in the triple store, and the reconciliation interval which is the difference between the time data is stored in the triple store and the time data is reconciled.

The response time of one scraping task related to one RSS feed ranges from few seconds to 3 minutes. This duration is mainly affected by the number of the event-related entities such as artists and attendees, where each one of them requires an API request. In Figure 4.4, we observe that the storage interval varies from 50 to 90 minutes attesting that our system contains the freshly uploaded photos in Flickr. We note that this variation is correlated with the delay between uploading photos and updating the Flickr RSS.

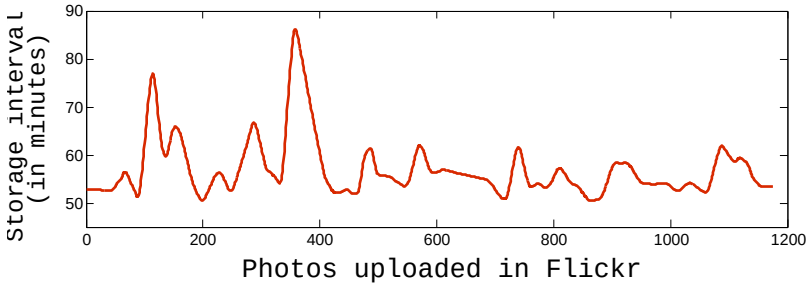

Figure 4.4: Evaluation of the storage interval

The response time of the reconciliation task ranges from few seconds to 6 minutes depending on the number of entities to be reconciled. Figure 4.5 highlights the short interval between the storage time and the matching time which approves the efficiency of our real-time reconciliation.

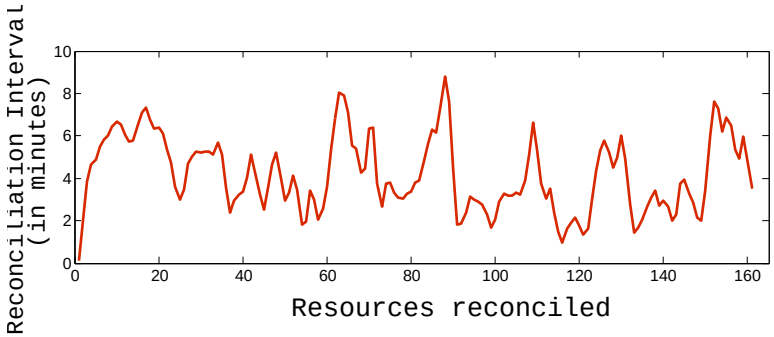

Figure 4.5: Evaluation of the reconciliation interval

4.2 Matching Semantic Events with Microposts

Social media has become an ubiquitous channel for users to share their thoughts and experiences. They host a huge amount of unstructured data imposing challenges to process and fully leverage the information. Part of this data may refer to real-world events (e.g., concerts, conferences) or to a particular piece of events (e.g., artist, talk, location, etc.). In this section, we overview related work on matching events and media, and we present our approach based on NLP techniques used to bridge the gap between structured and unstructured data.

4.2.1 Challenges and Related Work

With the exponential growth of online content, mining the relations between events and media has become a flourishing field in research. In particular, detecting events from Twitter microposts has been the subject of many recent efforts [140, 124, 8, 54, 121]. Indeed, Twitter is a valuable channel to gather real-time information for a variety of events. Such information may include participants' feedback or reflect what happened during events. However, mining event information from Twitter is a challenging research task due to the heterogeneity and to the sheer amount of data published. The short textual information and the noise inherent to microposts reveal to be serious challenges to efforts which aim to detect and track a specific topic or event. In addition, the real-time nature of Twitter demands advanced techniques to ensure scalable processing of continuous stream.

Mining the relations between events and media is often cast as a problem of event detection from social media. Several recent works [140, 124, 54] have proposed different approaches which are essentially based on clustering and classification techniques from machine learning or signal processing fields. In [140], the authors employ a wavelet analysis to detect unknown events considered as bursts of similar words within a temporal window. Other approaches [8, 124, 54] proposed to train a classifer based on Support Vector Machine (SVM) to detect which microposts that concern real-world events. Different from these approaches, our aim is to link events with media from reference reconciliation perspective. This task has been the subject of some research studies. One straightforward way is to simply retrieve tweets using the event hashtag if it is explicitly provided. In the case of missing hashtags, the work in [88] proposed an automated approach to extract popular events in a structured format from a social news website called Digg [7]. Relevant tweets are retrieved using full text search based on the event title, and filtered using a time interval.

In this thesis, our work goes beyond these approaches and aims at discovering links at the sub-event level of granularity. Indeed, one real-world happening can be part of a composite event. It is mainly known as an atomic event or sub-event. Typical examples are the scientific conferences which are generally composed of sub-events such as talks, tutorials, keynotes and so on. Moreover, it has been shown that the scientific community actively use

7. http://www.digg.com

Twitter as a valuable tool to communicate and share links and thoughts [37]. More precisely, the tweets which are related to conferences mostly contain useful materials such as slides, posters, videos, and also discussions about different talks, topics and people. However, only large events (e.g., main conference, workshops) have in general a hashtag associated with them, but most of sub-events (e.g., talks) are implicitly mentioned in tweets. This prevents users to easily track feedback about specific talks or topics and to clearly identify key points of discussion. In [121], the authors propose to convert tweets into RDF and enrich them by DBpedia concepts using Zemanta [8] keyword extraction API. Then, they align events with media using clustering and classification methods from machine learning. Motivated by the benefits of matching atomic events with media, we propose an automated approach leveraging the Natural Language Processing (NLP) and bridging the gap between structured and unstructured data. Our aim is to also ensure a real-time reconciliation instead of dealing with static datasets.

4.2.2 RDF Representation of Microposts

The first step in our approach is to represent tweets in a machine-readable format to be part of the Linked Open Data. This will enable the linkage of events with tweets using `lode:illustrate` property, thus forming a rich network of information. Structuring microposts is specifically the aim of Twarql [93], an open-source which encodes the microblog posts as RDF Data. Twarql highlights the benefits of RDF structuring in the analysis of microposts using expressive SPARQL queries. Still, there is a need to add meaning to the micropost itself. As machines can understand what an event is, they also require to make sense out of microposts. To solve this, one common solution is to extract meaningful metadata which is performed by the so-called Named Entity Recognition (NER) tools. This Information Extraction (IE) task seeks to identify and classify elements of text called "*Named Entities*" (NE). There is no consensus about what a named entity is. Overall, NER tools attempt to locate semantic information units that may refer to persons, organizations, locations, numerical and temporal expressions [99]. This information unit is represented by a named entity which has a label and associated with a specific class. In this field, the Semantic Web community have proposed new approaches for fine grained classification of named entities using popular ontologies such as DBpedia ontology. Besides, the disambiguation of named entities is made possible by linking them to URIs of real-word objects in Linked Open Data. Recently, Juric et al. [60] show that NE extraction together with topic modeling are relevant to contextualize debate transcripts. They make use of extracted context elements to achieve automatic link discovery between political debates and media.

We propose to use a NER tool that detects contextualized entities from microposts. This will obviously help machines to make abstraction from the noise in these short messages, and acquire a minimal understanding on what exactly the message entails. For example, given

8. http://www.zemanta.com

the following tweet: *"Kihara is attending Biophysical Society meeting at San Diego until Tuesday morning #bps12"*, the named entities extracted are *NE*= { *(Kihara, Person), (Biophysical Society, Organization), (San Diego, Place)* } using the extractor DataTXT-NEX [9] as depicted in Figure 4.6. We can observe that each NE is automatically classified into appropriate categories such as person, organization and place. These entities enrich the RDF description of the micropost using the property `dcterms:subject`. Figure 4.7 shows the RDF description of the previous example based on SIOC and LODE ontologies as well as the Dublin Core Terms vocabulary [10].

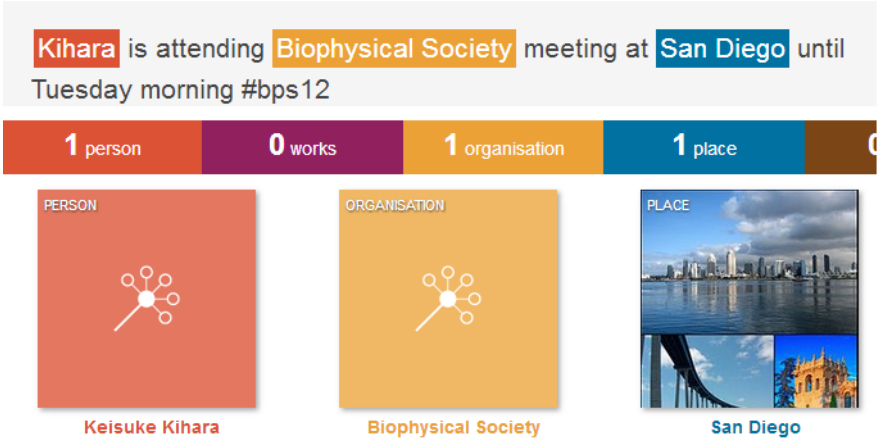

Figure 4.6: An example of named entities extracted from a micropost using dataTXT-NEX

```
@prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>.
@prefix sioc:<http://rdfs.org/sioc/ns#>.
@prefix lode:<http://linkedevents.org/ontology/>.
@prefix owl:<http://www.w3.org/2002/07/owl#>.
@prefix dcterms:<http://purl.org/dc/terms/>.
@prefix dbpedia-owl:<http://dbpedia.org/ontology/>.
<http://data.linkedevents.org/tweet/ab675d40-7f38-4fb0-93a1-1cf352f03ee5>
 a sioc:Post;
 sioc:id "173693229584232448";
 sioc:content "Kihara is attending Biophysical Society meeting at San Diego
 until Tuesday morning #bps12";
 sioc:hasCreator <http://twitter.com/kiharalab>;
 lode:illustrate <http://www.biophysics.org/2012meeting>;
 owl:sameAs      <http://twitter.com/kiharalab/status/173693229584232448> ;
 dcterms:date "2012-02-26T09:57:47";
 dcterms:subject [ a dbpedia-owl:Person ; rdfs:label "Kihara" ],
                 [ a dbpedia-owl:Location ; rdfs:label "San Diego" ],
                 [ a dbpedia-owl:Organization ; rdfs:label "Biophysical Society" ].
```

Figure 4.7: RDF/Turtle description of a micropost enriched with named entities

9. http://dandelion.eu/products/datatxt/nex
10. http://dublincore.org/documents/dcmi-terms

4.2.3 NER-based Matching Approach

Once converted to machine-readable format, microposts have to be associated with the corresponding events. The problem is how to detect a link between a micropost instance with a relevant event instance. This task refers to the instance matching problem in the Semantic Web, where our particular aim is to link data by *lode:illustrate* property. In this scenario, we consider that an event is fully described with rich structured data providing details about various elements such as sub-events, persons and topics. In order to enrich events with microposts, we capitalize on their structured decription in an ontological model. The idea is to exploit the overlap of concepts between the ontology describing the events and the taxonomy describing the named entities classification. More precisely, for each named entity *NE* having a label *NE-label* and a class *NE-class* in a micropost, we perform one of the following operations:

- The first operation is applied when there is a mapping between the named entity class *NE-class* (e.g., Person) and a class *C* from the event ontology (e.g., Author). In this case, the potentially relevant events are associated with a resource in which the type is mapped to *NE-class* and the label is similar to *NE-label*.
- The second operation is performed when no mapping exists between the named entities classification and the event ontology. Using the full text search, we retrieve the candidate events in which the associated literals contain *NE-label*.

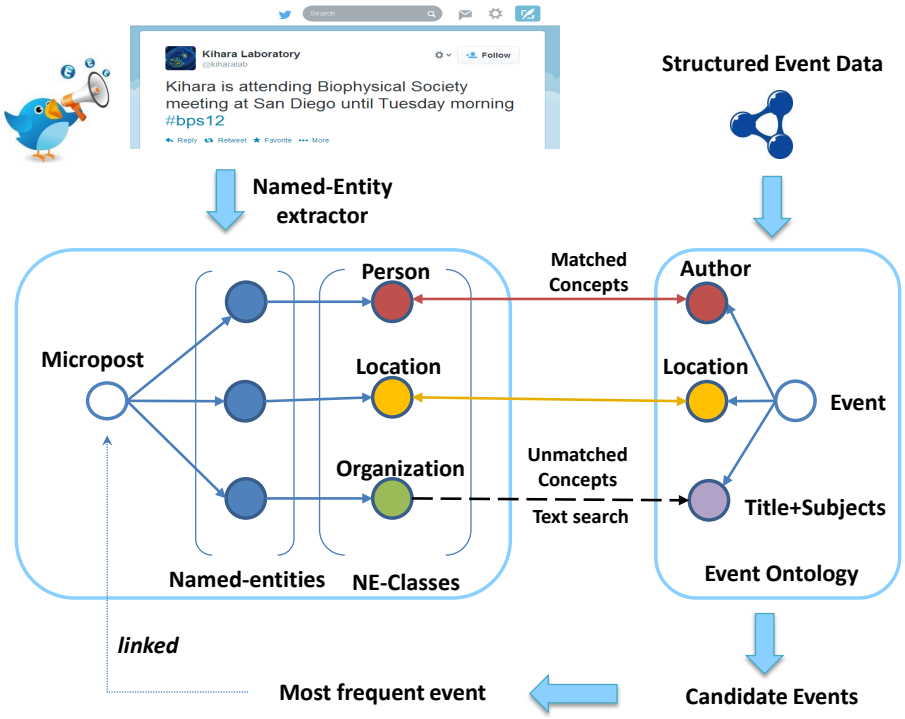

Figure 4.8: Overview of the alignment approach between microposts and events

Figure 4.8 illustrates our NER-based matching approach. For each micropost, a named entity is considered as a key to retrieve the list of candidate events using SPARQL queries. Then, we compute the frequency of occurrence of each event, and we link the most frequent one with the processed micropost. When the highest frequency concern multiple events, we select the event in which time is nearest to the micropost creation time. This approach can be applied in different domains. Still, it requires a mapping between the concepts of domain ontology and the taxonomy of named entities. This mapping can be achieved manually in reasonable time for small ontologies such as the case of LODE. Otherwise, it needs to be performed automatically by the use of ontology matching techniques [38].

4.2.4 Named Entity Recognition in Microposts

It can be drawn that our approach strongly depends on the performance of NER systems that should accurately annotate microposts. However, due to their informal and noisy nature, microposts provide new key challenges to existing NER systems which are mostly tailored for longer texts. When processing microposts, the NER system should be able to face the informal messages and to overcome the noisy linguistic features such as misspellings, grammatical errors and abbreviations. Moreover, the short texts often contain insufficient clues to efficiently contextualize an extracted term. This makes more difficult the classification and the disambiguation of a named entity which often need a background knowledge. All these issues drive our decision to carefully select a powerful tool that best fits our requirements. In particular, in order to achieve successful matching, we need to :

- Identify as much as possible relevant named entities in the micropost. We believe that this will mainly help increase the recall of our results.
- Ensure that the named entity is correctly classified in order to fully leverage the mapping between the event ontology and the NE taxonomy.

In research, NER is a problem that has been extensively approached on long and formal text such as newswire articles. This has led to several NE extractors proposed in the literature. Although they have a common purpose, these tools make use of different algorithms, dictionaries and training data. Their behavior may differ from one domain to another depending on the diversity of domains in which the system has been trained. As an effort to evaluate the strengths and weaknesses of these tools, they have been bundled in a unified framework called NERD (Named Entity Recognition and Disambiguation) [117]. The idea behind is to reassemble, within a same platform, the NE extractors that provide a Web API. These APIs enable developers to easily query NE extractors and compare their outputs. As these systems classify entities in different taxonomies, NERD proposed a unified ontology [11] that maps between them for comparison purpose. Instead of considering only one system among NERD extractors, we decided to use a combination of them with the aim to leverage the strength of each technology used. Our belief is that the more extractors are applied, the more named

11. http://nerd.eurecom.fr/ontology

entities are recognized and accurately classified. In addition, the probability that a named entity is correctly classified (at least one time) is higher when using many extractors rather than only one. This has been proved by a recent NERD evaluation in [117] which shows that NE extractors, taken individually, have a lower performance than the combined results. Blending the assets of each extractor enhance the named entity recognition in both newswire and microposts.

4.2.5 Use Case and Results: ISWC Conference

The Semantic Web community has set up the so-called Semantic Web Dog Food (SWDF) server [12] that exposes structured data about conferences as well as their related sub-events (e.g., talks, tutorials, sessions). As a use case to assess our NER-based matching approach, we take as an example the International Semantic Web Conference (ISWC) occurred in 2011. We first describe the ISWC 2011 dataset as provided by SWDF, and then we present the evaluation of our approach.

ISWC 2011 Dataset: Given all the authors, people who physically attended the conference or tried to follow it on social networks, we estimate that the ISWC 2011 has attracted more than 1,500 participants. The organizers published a lot of structured data about the conference including the list of accepted papers, their authors and institutions, the detailed program composed of sub-events with the exact timetable. This data is modeled using the SWC ontology [13], which is designed to describe academic events, and uses classes and properties from other ontologies such as FOAF to describe persons and SWRC to describe BibTeX elements of the publications [134]. The main conference is of type `swc:ConferenceEvent` and has a set of sub-events, namely `swc:WorkshopEvent`, `swc:TutorialEvent`, `swc:SessionEvent` and `swc:TalkEvent`. Table 4.9 shows some statistics about the ISWC 2011 dataset. The conference hosted 16 workshops, but only 8 of them provide complete description of presented publications. Some other useful information are also missing. For example, some publications are not attached to any event in the dataset, and no metadata exist about the keynotes speakers or about other important events such as the *Semantic Web Death Match*.

Main Event	Sub-Event	# Events	Papers	Authors
Conference Event	Workshop Event	16	75	185
	Tutorial Event	7	7	20
	Session Event	1	66	202
	Talk Event	93	93	275
	-	-	133	385
Total (distinct)	4	117	292	735

Table 4.9: Statistics about the ISWC 2011 dataset provided by SWDF

12. http://data.semanticweb.org/
13. http://data.semanticweb.org/ns/swc/ontology

```
@prefix swc: <http://data.semanticweb.org/ns/swc/ontology#>.
@prefix swrc: <http://swrc.ontoware.org/ontology#>.
@prefix event:<http://data.linkedevents.org/event/>.
@prefix paper:<http://data.semanticweb.org/conference/iswc/2011/paper/>.
@prefix ical:<http://www.w3.org/2002/12/cal/ical#>.

# Conference Event
event:conference-8179179367  a swc:ConferenceEvent;
   swc:isSuperEventOf  event:session-82730876;
   dc:title "10th International Semantic Web Conference".

# Session Event
event:session-82730876 a swc:SessionEvent;
   swc:isSuperEventOf  event:talk-388265905;
   ical:dtstart  "2011-10-25T14:00:00"^^xsd:dateTime;
   ical:dtend    "2011-10-25T16:00:00"^^xsd:dateTime;

# Talk Event
event:talk-388265905 a swc:TalkEvent;
   swc:hasRelatedDocument paper:industry13;
   dc:title "Practical applications of Semantic Wikis in commercial environments".

# Paper
paper:industry13 a swrc:InProceedings;
   dc:subject "realworld","semantic data integration","web 2.0","use cases",
   "Semantic enterprise wiki";
   dc:title "Practical applications of Semantic Wikis in commercial environments";
   swrc:author <http://data.semanticweb.org/person/daniel-hansch>.

# Person
<http://data.semanticweb.org/person/daniel-hansch> a foaf:Person;
   rdfs:label "Daniel Hansch";
   foaf:homepage <http://www.ontoprise.de>.
```

Figure 4.9: An RDF example describing some events in ISWC 2011 dataset

Figure 4.9 shows a sample of data where the main conference is the super-event of a session, which is in turn the super-event of a talk. The talk event concerns the presentation of one paper which is related to some metadata such as title and authors. We collected data from multiple social media sites in real time during the six days of the conference using the hashtags advertised by the organizers such as the main hashtag (*#iswc2011*) and the workshop hashtags (e.g., *#cold2011*, *#derive2011*). Table 4.10 shows some statistics about the different used sites along with the number of associated items and users. As expected, Twitter is by far the most used site: we have been able to collect 3,390 tweets shared by 519 users. A significant proportion of tweets contains hyperlinks that we have further analyzed. Hence, we extracted 384 different websites indexed by the so-called URL shorteners (e.g., Bitly) found in 1,464 tweets (43% of tweets). These links point to various Web resources such as blogs, slides and photos. For example, 25% of these resources consists of PDF documents published in the conference.

Media Service	Items	Users
Twitter	3390 tweets	519
pic.twitter	12 photos	6
yfrog	10 photos	9
Twitpic	10 photos	6
Flickr	47 photos	6
Google+	30 posts	26
Slideshare	25 slides	20

Table 4.10: Media services used during ISWC 2011 conference

A deeper analysis of collected tweets reveals the highlights of the conference or at least the parts which drew the most attention. Table 4.11 shows the top-five hashtags used during the ISWC 2011 conference.

Event/Concept	Hashtag	Frequency
ISWC 2011 main Conference	#iswc2011	2916
Linked Data Concept	#linkeddata	215
SemWeb Death Match 2011 Event	#deathmatch	213
SDOW 2011 Worksop	#sdow2011	172
COLD 2011 Workshop	#cold2011	170

Table 4.11: Top-five hashtags used in Twitter and related to ISWC 2011 conference

The most used hashtags are related to the main conference, some popular workshops and Semantic Web concepts. The second most tweeted event is the *Semantic Web Death Match*. Although it is important, this event was not available in the SWDF dataset. We have also analyzed the number of tweets per day as depicted in Figure 4.10. We observe a higher peak on Wednesday with 1,122 tweets from 248 users. The reason behind this peak is two-fold: the *Semantic Web Death Match* event during which 144 tweets were posted, and the keynote talk which was the subject of 136 tweets. Both events were not described in the SWDF corpus, while social network activities clearly state them as important moments of the conference.

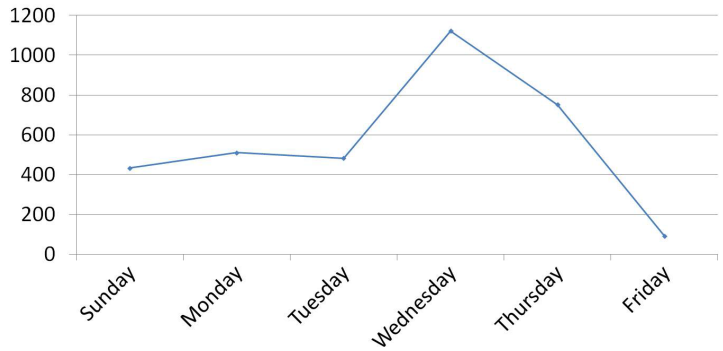

Figure 4.10: Number of tweets per day collected during the ISWC 2011

Matching ISWC sub-events with microposts

In order to apply our NER-based matching approach on ISWC 2011 dataset, we first analyze the "semantic" intersection between the NERD ontology and the SWC ontology used by SWDF. This intersection contains only two elements, namely Person and Organization. The Person class is used to represent the authors, while the Organization class is used to represent their affiliations. However, many events are associated with the same organization as many authors share the same affiliation. The author's affiliation is therefore not discriminant enough and it is likely to add irrelevant events when selecting candidates. Based on this observation, we decided to take into account solely the Person class.

On the other hand, as it can be drawn from the example of SWDF dataset in Figure 4.9, the most atomic event has as type `swc:TalkEvent` which describes a talk during which a paper is presented. Other atomic events (e.g., demo, poster) are also related to a large set of publications which are described through rich metadata such as title, authors and subjects. As a consequence, we resolved to reason directly over the publications instead of events. The relevant event is then retrieved by using its direct link with the relevant publication expressed by `swc:hasRelatedDocument` property. Algorithm 1 shows the pseudo-code of our NER-based reconciliation approach tuned for the ISWC 2011 dataset.

Algorithm 1 Real-time NER-based Reconciliation for ISWC 2011 dataset

```
tags ← set of tags used for lookup resource from a social media site
MS ← social media sites
while true do
  C ← fresh data from the conference provider
  mediaItemList ← retrieve the list of media items using tags
  for item ∈ mediaItemList do
    Intialize namedEntityList, publicationList
    namedEntityList ← extract named entities in the item
    for namedEntity NE ∈ unique(namedEntityList) do
      if class(NE) = Person then
        publicationList ← list of publications of the author NE from C
      else
        publicationList ← list of publications in which title contains NE from C
        publicationList ← list of publications in which topics contain NE from C
      end if
    end for
    relevantPublication ← most frequent publication in publicationList
    matchedEvent ← the event related to relevantPublication
  end for
end while
```

Evaluation: Performance of NE extractors

We extracted named entities using the NERD API. We observe that the large part of named entities were extracted by the following NE extractors: AlchemyAPI [14], Wikimeta [15], Extractiv [16], OpenCalais [17] and Zemanta [18]. In particular, Wikimeta recognized the highest number of named entities in this dataset. It has a strong ability to locate and to classify named entities. Overall, all extractors except for Zemanta succeed to achieve an accurate NE classification. Finally, we report in Table 4.12 the grouping results of the 5 main concepts: Person, Organization, Country, Time and Number. Wikimeta classifies correctly a higher number of Person and Organization than other tools. But, it fails to identify other classes, which is mainly due to the small used taxonomy.

	AlchemyAPI	Extractiv	OpenCalais	Wikimeta	Zemanta
Person	879	71	568	1340	138
Organization	54	-	47	2742	16
Country	13	4	13	-	34
Time -	5	-	-	-	
Number	-	62	-	-	-

Table 4.12: Number of axioms aligned to the NERD ontology for each extractor

Evaluation: Reconciliation algorithm

After removing retweets (i.e., tweets preceded by RT), we manually constructed a ground truth and we discovered that only 245 tweets (among 1710) directly concern events. The remaining tweets mostly discuss a general topic or specific concepts in the Semantic Web. In order to assess the added value of named entity extraction compared with simple keywords, we decided to perform Algorithm 1 where the named entities are replaced by keywords. More precisely, for each tweet, we retrieve the candidate events in which titles contain at least one of the extracted keywords. Then, we select the most frequent event. We have used the AlchemyAPI Keyword Extraction [19], a tool that extract topic keywords from texts. Three experiments have been performed using: (1) NE-based algorithm, (2) keyword-based algorithm, and (3) the combination of the output obtained from both NE-based and keyword-based algorithms. We show the precision and the recall in Table 4.13. These results show a relative good performance if we consider the lack of useful metadata in the ISWC 2011 dataset. As expected, the NE-based reconciliation is more precision-oriented than keyword-based reconciliation. To enhance performance, further questions can be investigated for

14. http://www.alchemyapi.com
15. http://www.wikimeta.com
16. http://extractiv.com
17. http://www.opencalais.com
18. http://www.zemanta.com
19. http://www.alchemyapi.com/api/keyword-extraction

future work such as: what optimal combination of NE extractors should be applied, and how to filter keywords and retain the most discriminative ones?

	Precision	Recall	F-score
Named-entity-based algorithm	61%	49%	54%
Keyword-based algorithm	40%	55%	46%
Hybrid approach (Named-entity + Keyword)	43%	64%	51%

Table 4.13: Precision-Recall of NER-based reconciliation

The evaluation has proved the importance of the class `Person` to retrieve relevant events (465 times), compared with other types such as `Organization` (295 times) and `Technology` (124 times). Moreover, our approach succeeded to detect some surprising true matching such as the examples provided in Table 4.14. This success is due to the accuracy of NERD to extract persons from microposts, and to the exploitation of the semantic overlap between the event ontology and the NE taxonomy.

Event	Tweet
Extending Functional Dependency to Detect Abnormal Data in RDF Graphs	Jeff Heflin and students seem to be doing a lot of stats, analysis of RDF data (e.g., quality assessment) these days
Learning Relational Bayesian Classifiers from RDF Data	Harris Lin: key limitation of learning methods is the necessity to have a direct access to RDF data

Table 4.14: Examples of true positive in NER-based reconciliation

On the other hand, we show three representative examples of false positive in Table 4.15. For example, the tweet in the first row was created during the closing ceremony to announce the best paper award, while it was matched with the talk event that refers to the presentation of this paper. Still, such matching can be seen as correct from the topic point of view. Moreover, the missing information in the ISWC 2011 dataset induced a confusion to discriminate between the correct and the wrong events as shown in the second example. This example was incorrectly aligned with a paper presentation having one author named *Chris Welty*, whereas it should be linked with the *Death Match* event which also involves the same person. However, the description of *Death Match* event is completely missing in the dataset. The same problem persists in the third example where the program has no knowledge about what has been discussed during the *Frank van Harmelen's* keynote. The tweet is linked to the event with which it shares common keywords.

False Event	Tweet	True Event
DBpedia SPARQL Benchmark – Performance Assessment with Real Queries on Real Data	DBpedia SPARQL Benchmark won the best-paper award http://t.co/nqU7HHgv	Closing Ceremony
Semantic Web Technology in Watson	Chris Welty the Semantic Web was a success, but the W3C standards were badly misunderstood	Death Match event
DC Proposal: Capturing Knowledge Evolution and Expertise in Community-driven Knowledge Curation Platforms	Terminological vs instance knowledge explains split of our community: owl reasoning vs. linked data #keynote	Frank van Harmelen keynote

Table 4.15: Examples of false positive in NER-based reconciliation

4.3 Conclusion

Social media host an ever increasing amount of knowledge, but spread and locked into multiple sites. Mining the overlap of these sites is a key asset to enhance the exploration of data within a single channel. To achieve this, we proposed some approaches with a focus on Semantic Web and NLP technologies as powerful means for linking data. More precisely, we have presented two approaches in this chapter in order to resolve the instance matching problem in different contexts. The former approach focus on how to link data in a fully structured space, characterized by the presence of different data types. The latter seeks to bridge the gap between structured and unstructured data leveraging the NER technologies.

Conclusion of Part I

In this part, we have presented the different steps required to build an event-based environment, harnessing the wealth of information provided by different websites. We used the Semantic Web technologies to bring together event-centric data, so that they become more discoverable and reusable. Overall, we have described how this data has been extracted, RDFized, interlinked and published following the Linked Data principles.

First, we have presented a framework that offers a simple-to-use and flexible tool to scrap events and related media using some popular social media sites. This has let to the construction of EventMedia, an RDF dataset published in the Linked Data cloud and continuously fed by new data.

Second, we have detailed the challenges that arise when reconciling data derived from heterogeneous and distributed sources. Evaluation results show how the event matching is sensitive to the temporal distance, and how an efficient string similarity improves the accuracy. Finally, we have tackled the problem of matching microposts with fine-grained and structured events, where the challenge is to face the noise of informal and short messages.

Part II

Exploring the Event Landscape: Applications, Recommendation and Community Detection

Overview of Part II

As Linked Data contains millions of RDF triples, consuming this knowledge can benefit various tasks such as enrichment, personalization and social analysis.

In Part II, we consume the Linked Data in event domain in order to create Semantic Web applications and to provide advanced personalization solutions.

In Chapter 5, we present some Semantic Web applications that support a friendly interface to browse events and help create an event with consistent details. We highlight the limitations of existing technologies designed to access and use RDF data in Web applications.

In Chapter 6, we propose a hybrid recommender system built on top of Semantic Web to make personalized suggestions of events. Such a system faces a number of challenges due to the inherent complex nature of events.

In Chapter 7, we propose an approach to detect overlapping semantic communities in event-based social network (EBSN). The idea is to detect communities that have high connectivity and share semantically similar topics. Community detection helps understand user behavior at a group level and improve personalization.

CHAPTER 5

Consuming Event-centric Linked Data

To date, Linked Data hosts billions of RDF triples based on various vocabularies and covering many domains such as government, health, media or more generally encyclopedic data. This wealth of information needs to be accessed, reused and visualized. Although recent efforts have endeavored to consume Linked Data, there is no clear insight into the level of maturity reached. In fact, more advanced technologies are needed to lower the barrier for the adoption of Semantic Web applications [91].

A conventional Web application is based on client-server communication and user interface technologies (e.g., HTML, Javascript, CSS) processed by Web browsers. Then, the introduction of graph based knowledge is the extension made by the Semantic Web application [98]. Such knowledge mostly makes use of RDF data model and includes some standards such as RDF Schema, OWL and RDFa (i.e., RDF annotation within HTML). Nevertheless, this extension has introduced new challenges in the development of Web applications [91, 47]. Overall, the difference lies in three layers: data storage, transaction processing and user interface. A greater attention is needed in order to create efficient technologies able to overcome the adoption bottleneck.

In this chapter, our aim is not to propose new solutions, but to exploit existing technologies and highlight their limitations. Ultimately, we wish to consume Linked Data and to develop new applications allowing end-users to browse, search and create events. We also propose an analysis that underlines the concrete benefits of Linked Data in some tasks such as event detection and user profiling.

5.1 EventMedia Application

EventMedia is a dataset that provides descriptions of events which are associated with media and enriched with Linked Data cloud. The back-end of our system consists of a live data crawler and an interlinking framework as has been previously described in Chapter 3. The front-end consists of a user friendly interface, designed to meet the user needs: relive experiences based on media and support decision making for attending upcoming events. The architecture of our system is depicted in Figure 5.1.

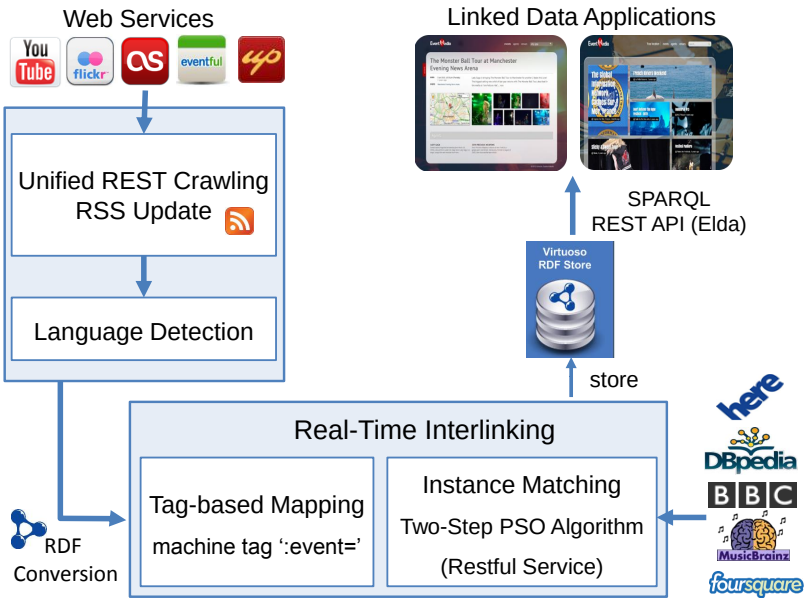

Figure 5.1: EventMedia System Architecture

5.1.1 UI Challenges

Along with the constant evolution of JavaScript and Web standards (e.g., HTML 5, CSS 3), developers become able to create a full-fledged application on the front-end side. This, so-called single page application, is supported by a large number of recent JavaScript frameworks allowing for faster and more interactive applications. Such frameworks facilitate building complex applications typically consuming data accessible via REST API and encoded in the popular JSON syntax. On the other side, Semantic Web applications are typically powered by an RDF triple store where data is published according to the Linked Data principles. This RDF data can be serialized in different formats including JSON. Still, it is challenging to represent RDF in an idiomatic JSON as commonly used in conventional Web applications. This problem arises due to the model mismatch between RDF and JSON (e.g., graphs vs trees, URIs vs shortnames, etc.). Consequently, dealing with RDF data and complex SPARQL queries is harder than dealing with raw data in current Web applications. There is a strong need for specialized libraries that comply with standard methods and best Web development practices. For a feasible adoption, Web developers should be able to use JSON as they normally would without considering to work within RDF data model. Another challenge is the structure variation and the large amount of data in EventMedia. Several descriptions of events may have missing properties such as involved artists and participants, and several links exist between similar entities. How to represent a large amount of data having relatively different representations is another question to be considered in the design of the user interface.

5.1.2 Elda: Epimorphics Linked Data API

The barrier caused by the technological change prompts researchers to propose solutions complying with existing Web technologies. In terms of data access, these proposals have been classified in three categories [50]. The first one is the endpoint enabled by a triple store to query RDF data with the expressive power of SPARQL query language. However, RDF with its different serializations and SPARQL are foreign to the most of Web developers. The second category is about a direct access to RDF resources over HTTP. This provide a native support that fully leverage Linked Data making use of the follow-your-nose approach. Such design is compatible with Linked Data principles, but supports very limited expressivity. Finally, the last category is based on Entity-Attribute-Value model (EAV) to map RDF over HTTP. In this API, an adapter layer maps the RDF triples to key-values pairs serializing the response in different formats like the idiomatic JSON using plain attributes names. An example of this adapter is the Linked Data API [115] which provides a familiar interface for the majority of Web developers. Although such an interface can increase the uptake of Linked Data in Web applications, the mismatch of data models (e.g., graph vs tree) might introduce some overhead, and the application might not be aware of some links in the data.

To easily make use of advanced JavaScript frameworks, we opted for the Linked Data API that provides a configurable way to access RDF data using simple RESTful URIs translated into SPARQL queries. Indeed, the API layer is deployed as a proxy in front of a SPARQL endpoint, and supports the provision of sophisticated querying without the need to write or parse SPARQL queries. In particular, we use Elda [1], a java implementation of the Linked Data API. Elda comes with some pre-built samples and documentation which allow building a specification to enable the connection between the back-end (data in the triple store) and the front-end (visualizations for the user). The API layer helps associate URIs with processing logic that extracts data from the SPARQL endpoint using one or more SPARQL queries, and then serializes the results using the format requested by the client. In particular, Elda provides a simplified XML and JSON representations of RDF data. It enables the creation of complex front-end applications in a relatively standard way without any consideration of RDF. This method greatly simplifies the development task, however, it fails to fully leverage the benefits of the Semantic Web such as the use of complex queries. Figure 5.2 depicts a sample of Elda specification file that defines two endpoints: an item endpoint to show one single event using its *ID*, and a list endpoint to show a collection of events using some filters (e.g., type, time, location, etc). The properties associated with each event are defined in the specification of the event viewer. Other viewers and endpoints do also exist such as for media, agent, location and user.

1. http://code.google.com/p/elda

```
# Define some properties of the API
<#EventMediaAPI> a api:API ;
        rdfs:label "EventMedia API"@en;
        api:maxPageSize "1000";
        api:defaultPageSize "10";
        api:endpoint <#event>,<#eventbyid>;
        api:sparqlEndpoint <http://eventmedia.eurecom.fr/sparql>;
        api:defaultViewer api:describeViewer.


# specification of the event viewer (which properties describe an event)
spec:eventViewer a api:Viewer ;
  api:name "event";
  api:property "title","description","space.lat","space.lon","inagent.label",
               "place.label",etc.


# Item Endpoint, showing a single event specified by its id
<#eventbyid> a api:ItemEndpoint;
                api:uriTemplate "/event/{id}";
                api:itemTemplate "http://data.linkedevents.org/event/{id}";
                api:defaultViewer spec:eventViewer.

# List Endpoint, showing multiple events selected by filters
<#event> a api:ListEndpoint ;
                api:uriTemplate "/event" ;
                api:defaultViewer spec:eventViewer;
                api:selector [
                        api:where "?item a lode:Event." ].

# All the properties are defined using a label name
rdf:type
  api:label "type".
rdfs:label
  api:label "label".
```

Figure 5.2: A sample of Linked Data API specification in EventMedia

5.1.3 EventMedia UI

This section illustrates the design of EventMedia user interface that allows discovering events through different contexts and visualizing associated media.

Visual Design

One design challenge is how to enable fluid faceted navigation of a vast event-based space, and how to create harmonious views of interconnected datasets. Users wish to discover events either through invitations and recommendations or by filtering available events according to their interests and constraints. We provide mechanisms to browse events by location or a period of time. Once an event is selected, related media are presented to convey the event experience along with background information such as category, agents, venues, attendance, etc. A typical example is illustrated in Figure 5.3 which shows a concert of Lady Gaga in 2010.

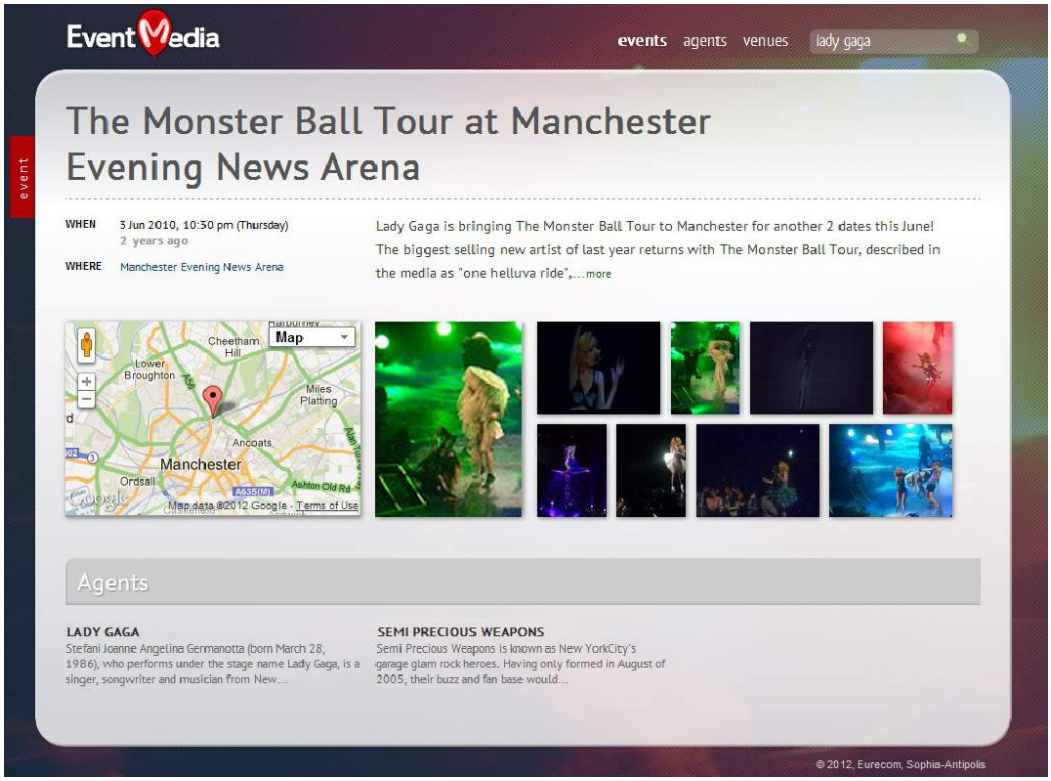

Figure 5.3: Interface illustrating a concert of *Lady Gaga* in 2010

Apart from the inspection of the event instance, other conceptual classes (e.g., venues, agents, users) have accessible views, so that the user can obtain more information about these instances and explore events related to them. In addition, we leverage the "owl:sameAs" links with external public datasets. For example, the venue view is enriched by using the links between EventMedia and Here.com (i.e., a location-based service developed by Nokia). This enrichment helps users get feedback about venues through a set of reviews and rating obtained from Here.com. It also provides a variety of nearby venues such as restaurants, shops and theaters. Finally, we enable users to filter data by their favorite language and we incorporate the option to rank events by popularity.

Technically, the front-end is based on the popular Backbone.js [2] JavaScript framework that facilitates the development of complex user interfaces. It provides an elegant REST integration which makes straightforward the use of Elda. The demonstration of EventMedia application is available at http://eventmedia.eurecom.fr along with a demo video [3].

Mobility Adaptation

Adapting EventMedia application for mobile devices is a valuable asset, which however significantly influences the design of the user interface. The "mobile-first philosophy" has led to highly modular design and encouraged a simple design with every unnecessary detail removed. To enable mobility support, our user interface is based on a grid structure, allowing organizing the content into the columns of various widths and recombining their position when necessary. On smallest screens, the grid modules have width set to 100%. When the screen width increases, depending on the available size and situation, more pleasing layout is generated with 50% or 33% of the page width. Figure 5.4 shows the same user interface in two different screen sizes.

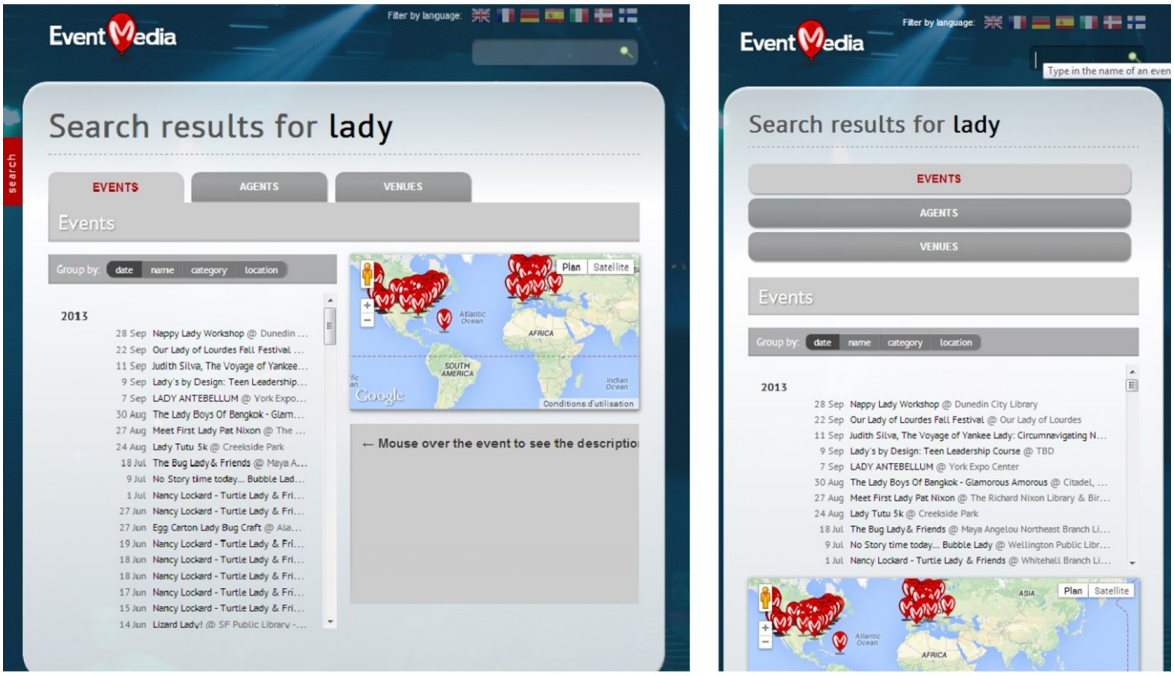

Figure 5.4: Same EventMedia Interface in two different screen sizes

2. http://backbonejs.org
3. http://eventmedia.eurecom.fr/demo.html

5.1.4 Discussion

To highlight the benefits of EventMedia user interface, we consider the following scenario: Alice wants to see what happened and who attended the "*Coldplay*" concert given on *August 8th, 2012* in *Chicago*. A first insight into the event directories underlines different descriptions of this concert as illustrated in Table 5.1. By collecting and linking this spread information, we provide a more homogeneous and complete description [4].

	description	time	place	category	ticket	artists	attendees	media
LastFm [5]		imprecise	√			√	√	√
Eventful [6]	√	√	√	√	√			
Upcoming [7]		√	wrong	√			√	

Table 5.1: Comparison between descriptions of Coldplay Concert in event-based services

EventMedia application as described in [65] won the first place of the Semantic Web Challenge 2012 hosted at the International Semantic Web Conference. It was considered as a good Semantic Web application in one keynote [8] given at the Extended Semantic Web Conference 2013. Our belief is that the key factor of this success lies in our design strategy focused on three important criteria, namely simplicity, flexibility and modularity:

- By simplicity, we refer to the interoperable and simple model that describes factual aspects of events, but which may come at the cost of more expressivity. This highlights one requirement in ontological engineering that is to consider not only the expressive power of the data model, but also the usability aspect. In fact, Web developers need to be able to access, filter and sort data using simple queries which strongly depend on the ontology designed.
- By flexibility, we mean the flexible data integration natively ensured by Semantic Web technologies and the flexible scraping framework allowing for the easy addition of new datasets.
- By modularity, we refer to the strategy followed in the user interface design. It enables us facing the high variability of data structure and fitting the constraints of mobile devices.

For the UI development, the use of Elda sheds light on some limitations to query and access data. At the time of writing, simple SPARQL operations are not supported such as *GROUP BY*, *HAVING* and *DISTINCT*. The same problem is raised when it comes to handle more complex queries including a sub-query or transitivity. We got around these limitations

4. http://eventmedia.eurecom.fr/#!event/a6ed6e81-fea5-4740-9e8f-2abf73273d5a
5. http://www.last.fm/event/3159427
6. http://eventful.com/E0-001-050047180-4
7. http://upcoming.yahoo.com/event/8634740
8. Keynote "What does it mean to be semantic?" at ESWC 2013, given by *Enrico Motta*, a professor at the Knowledge Media Institute (KMI), UK

by directly using SPARQL language when it is necessary. Another fact has been observed when developers used the EventMedia dataset. It concerns the representation of the temporal entity based on the Time Ontology [51]. This entity can be represented in two different ways: (1) an instant which is directly associated to the date time value, so that the path between an event and its temporal value is of length 2; (2) an interval which has a start and an end dates, and the path between an event and its temporal value is of length 3. Adopting this expressive representation in the LODE ontology was favored, instead of simplicity, to represent complex relationships to time (i.e., temporal intervals that do not coincide with date units) [129]. It appears that the price is paid when it comes to use and query data. Developers tend to neglect or rather avoid the second representation and they only retrieve events associated with an instant time. The use of *UNION* in SPARQL queries is unavoidable to retrieve all the events having different representations, sometimes generating complex queries or slowing down the response time. This proves the trade-off between expressivity and simplicity that needs to be considered by the ontology engineers.

5.2 Enhanced Facebook Event Application

Tens of thousands of events are created daily on Facebook, which motivated us to not only enrich EventMedia with Facebook, but also to enhance the native Facebook event application. In fact, when creating an event on Facebook, only some basic fields are provided such as title, description, time and location. These fields are very restrictive to describe an event and can be enhanced with other valuable features such as category and involved agents. Moreover, there is no mechanism to control the data redundancy. The same event could be created more than one time by different users which makes difficult to get a general overview. For example, when looking for the number of attendees or for a special attendee, one user has to check all the event pages on Facebook. Our goal is to improve this Facebook event application by proposing a new interface that presents a bridge between EventMedia and Facebook. This application should provide:

- More fields to create a rich event description by reusing information from Linked Data cloud such as artist description.
- Functionality to be able to upload illustrative media.
- Functionality to control the data redundancy and to prevent the creation of duplicate entities.

Compared with the classical Facebook interface, we integrate new widgets to facilitate the creation task as depicted in Figure 5.5. Among these widgets, there is a date picker, fields to describe locations (e.g., street, country) and a Google map for better visualization. We also enable the description of artists involved in the event and the uploading of media associated with it. To help users create new events, we propose an autocompletion functionality that queries the triple store and fetch the entities containing the typed word. When creating a new event, the application first stores the event in Facebook. Then, it generates the related

RDF/XML description that will be added or updated in EventMedia dataset depending on the user action (e.g., create, update). By querying the triple store, users can exploit the descriptions of existing events, agents and locations. However, control mechanisms are still needed to effectively prevent users from creating duplicates. The application is available online at https://apps.facebook.com/eventmedia.

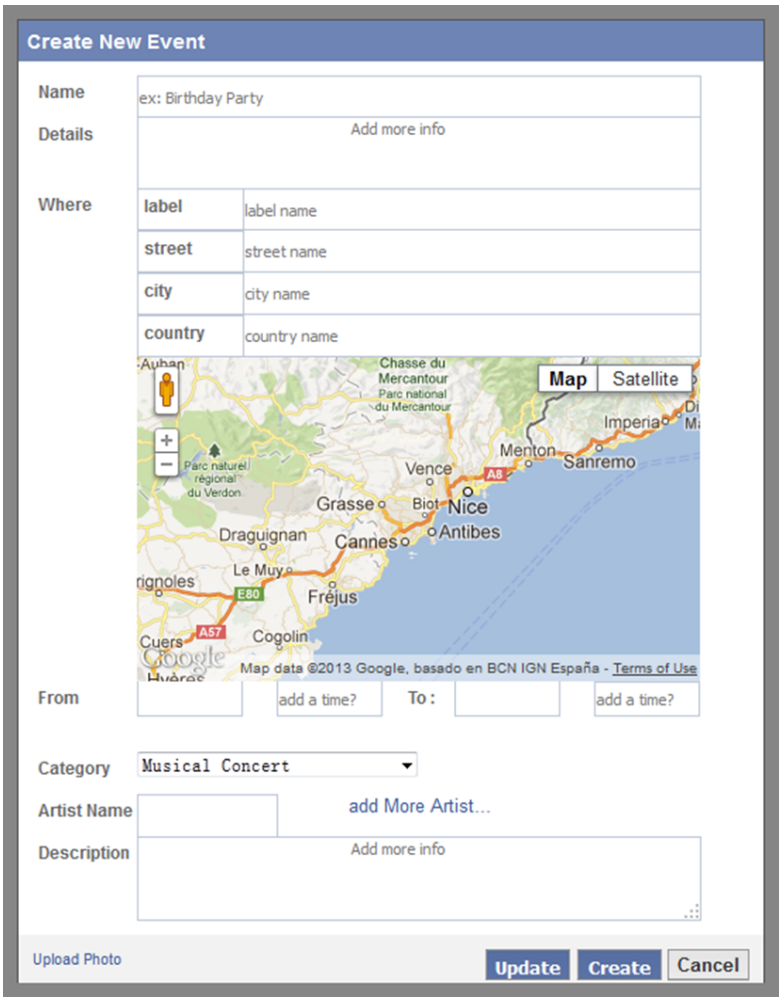

Figure 5.5: Enriched UI to create an event on Facebook

Technically, this application is based on Facebook JS SDK [9] and JS rdfQuery [10]. SPARQL queries are also used to manage the auto completion functionality. We agree that this application can be enhanced with more powerful functionalities and make it more user-friendly. The current application is a first prototype which still suffers from some issues. For example, the autocompletion shows duplicated names since the EventMedia dataset contains many entities having similar labels but originated from different sources.

9. https://developers.facebook.com/docs/javascript/
10. https://code.google.com/p/rdfquery

5.3 Confomaton: Conference Enhancer with Social Media

A scientific conference is a type of event where attendees have a tremendous activity on the Web. Participants tweet or post longer status messages, engage in discussions with comments, share slides and other media captured during the conference. They use popular social networks such as Twitter, Google+ and Facebook, and media platforms such as Flickr, YouTube and SlideShare. The information shared can be used to generate informative reports of what is happening, where (which specific room) and when (which time slot), and who are the active participants. We thus proposed a Semantic Web application called *Confomaton* [63] that exposes diverse media resources generated by users, that can potentially be linked with more structured metadata such as a detailed program of a scientific conference. Confomaton enables a better conference experience including visual conference summarization and explorative search. The name *Confomaton* is a word play based on the French term *Photomaton* (English photo booth) and *Conference*. Just like a Photomaton illustrates the scene inside of the booth, Confomaton illustrates an event such as a conference enriched with social media.

5.3.1 Confomaton Architecture

Confomaton is a Semantic Web application that produces and consumes Linked Data. It has a modular framework composed of four main modules as depicted in Figure 5.6: (1) a media collector working on a Node.js instance and monitoring numerous social networks and media providers; (2) an event collector based on our scraping framework (3) a reconciliation module in charge of linking the events with social media; (4) a graphical user interface powered by the Linked Data API.

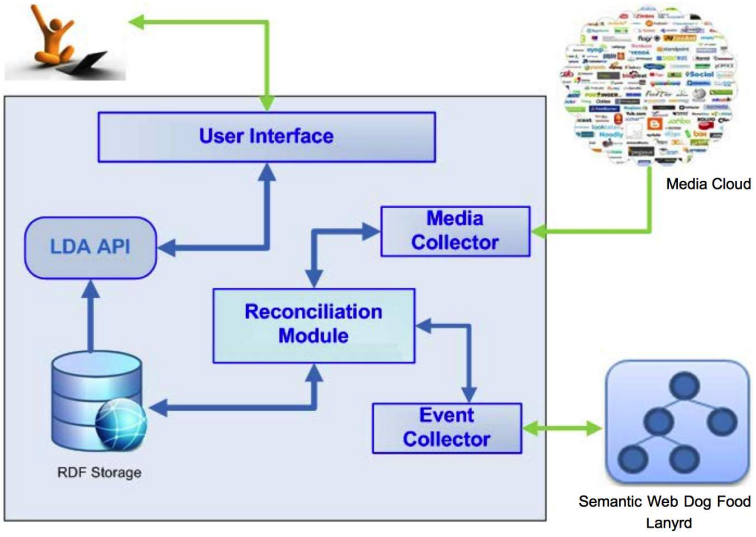

Figure 5.6: Confomaton System Architecture

The media collector is based on the framework proposed in [63] which retrieves various media items such as photos, videos and microposts from various social media sites. It supports 4 social networks (Google+, MySpace, Facebook, and Twitter) and 7 media platforms (Instagram, YouTube, Flickr, MobyPicture, img.ly, yfrog and Twitpic). Being agnostic of media providers, the framework delivers a unified output for all these sites. It takes as input a search term and then performs a parallel key-search using the APIs of the media providers. Figure 5.7 shows an example of metadata retrieved from Google+ and Flickr when searching by *#iswc2011* keyword.


```
{
 "GooglePlus": [
  {
   "mediaurl": "http://software.ac.uk/sites/default/files/images/content/Bonn.jpg",
   "storyurl": "https://plus.google.com/107504842282779733854/posts/6ucw1Udb5NT",
   "message": {...}
  }],
  "Flickr": [
  {
   "mediaurl": "http://farm7.staticflickr.com/6226/6290782640_e8a1ffdcc2_o.jpg",
   "storyurl": "http://www.flickr.com/photos/96628098@N00/6290782640/",
   "message": {..}
  }]
 }
```


Figure 5.7: Sample output of the Media Server searching by *#iswc2011* keyword

The event collector is based on our scraping framework and continuously retrieves up-to-date conference descriptions using Lanyrd feeds and Semantic Web Dog Food server (SWDF). Both sources are different in terms of data models and data granularity. The SWDF server hosts RDF archives prepared by a small set of people who are usually the conference organizers. It provides fine-grained information detailing the set of sub-events, such as sessions, talks along with the publications and their authors (as described in Section 4.2.5). The Lanyrd items are created and maintained by the wisdom of the crowd, but the descriptions are kept at a very general level: only some aspects related to the conference are described such as location, date, speakers, and attendees without any precision about sub-events and presented papers. For each retrieved event, we query the media collector to fetch media items in which the description contain the main tags of the conference. To discard noisy items, we retain the ones which are in a reasonable time window represented by the start/end dates of the conference. Retrieved media are directly linked with the main conference based on the main hashtag, and then we apply our NER-based reconciliation algorithm to link media with sub-events as described in Section 4.2.

5.3.2 Confomaton UI

The *Confomaton* user interface offers four perspectives characterizing an event and represented by tabs: (1) *"Where does the event take place?"*, (2) *"What is the event about?"*, (3) *"When does the event took place?"*, and finally (4) *"Who are the attendees of the event?"*. To access RDF data, we use the Elda implementation of the Linked Data API. The main demonstrator is available at `http://eventmedia.eurecom.fr/confomaton`.

On the left side of the main view, the user can select the main conference or one of its sub-events. On the center, the default view is a map which is centered on where the event took place (e.g., Bonn, Germany). The *What* tab is media-centered and allows to quickly see what illustrates a selected event (tweets, photos, slides). For the *When* tab, a timeline is provided in order to filter conferences according to a period of time. Finally, the *Who* tab aims at showing all the participants of the conference. Figure 5.8 shows two screenshots: one (on the left) describes up-to-date conferences located in different regions, and the second one (on the right) describes the ISWC 2011 conference and its sub-events.

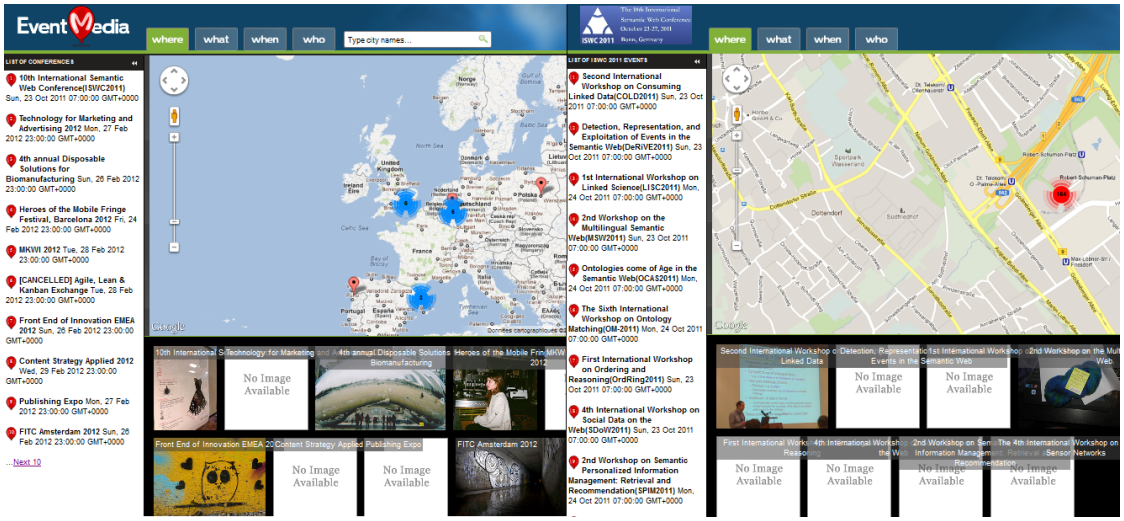

Figure 5.8: A showcase of *Confomaton* with Lanyrd (left) and Semantic Web Dog Food (right)

5.3.3 Discussion

Confomaton is a Web application that won the "Best Concept Award" in the "Linked Data-a-Thon" challenge of the International Semantic Web Conference (ISWC) 2011. It shows well the difficulty to use Semantic Web technologies in a real setting. The solution we proposed makes use of many social sites starting from scraping, reconciling and visualizing data. Nevertheless, the more sites are handled and the more issues one has to deal with, generally with the API provided by those sites (e.g., the number of requests of some APIs,

1500 requests per day for the Twitter API). Regarding the Linked Data API, at the time of writing, it was not possible to handle queries using selectors with *DISTINCT* and *GROUP BY* queries. We also faced the problem of the objective criteria to select a particular vocabulary for modeling the data. For instance, how to choose the right model for a hashtag (e.g., hashtag a `sioc:Topic`?). There is a need for providing more guidelines to help developers in deciding which vocabulary best fits their needs.

5.4 Behavioral Aspects and User Profiling

Apart from enriched views, we investigate the benefits of data reconciliation to conduct social analysis. In this section, we exploit the connection of event-centric data in order to provide insight into some behavioral aspects or to improve user profiling.

5.4.1 Behavioral Analysis using Linked Data

One advantage of linking events with media is to underline some aspects about photo sharing activity. Analyzing the spatial-temporal dimension is, for example, useful for event detection problems. Thus, we investigate how users share media according to the event time and location. Figure 5.9 shows the long tail trend to upload photos right after the event started, and nearby the venue in which it took place. This lets us suggest that one potential feature in event detection is to identify the peaks of users' activities within a narrow spatial-temporal window.

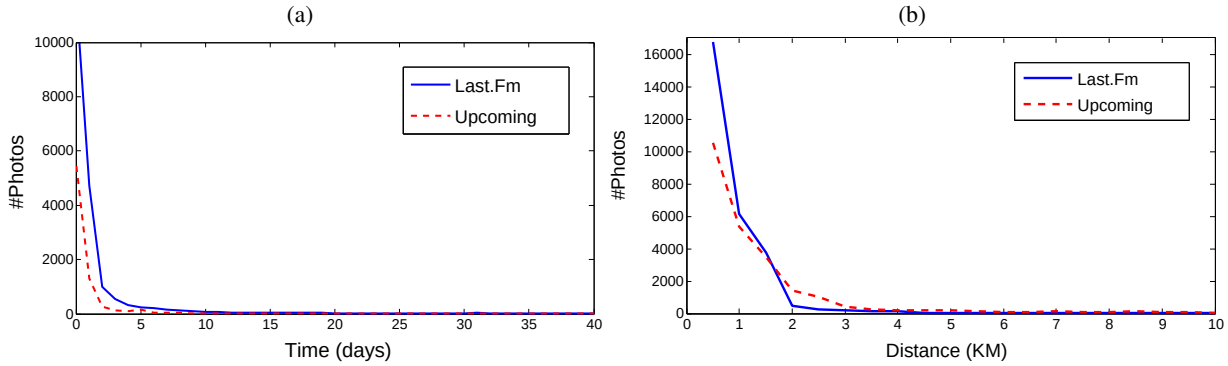

Figure 5.9: The trend to share photos in the temporal-spatial dimension

In addition, we investigate how people upload photos according to the attendance rate. Figure 5.10 highlights a strong positive correlation between the attendance rate and the amount of shared media. Moreover, we discover that most of active users are in general located in "United States" and "United Kingdom" as depicted in Figure 5.11. Such analysis is made possible thanks to the rich description of events (e.g., attendance, location) and their relationships with media.

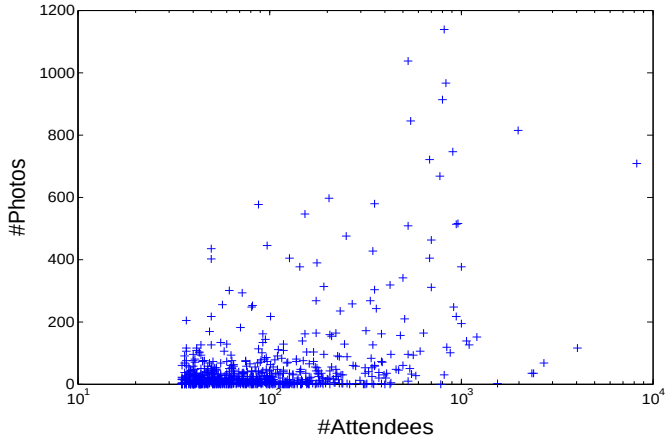

Figure 5.10: Correlation between the attendance rate and the amount of shared media

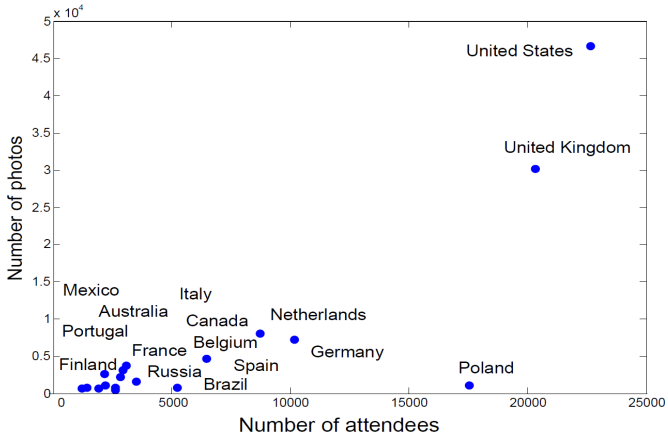

Figure 5.11: The trend to share photos in different countries

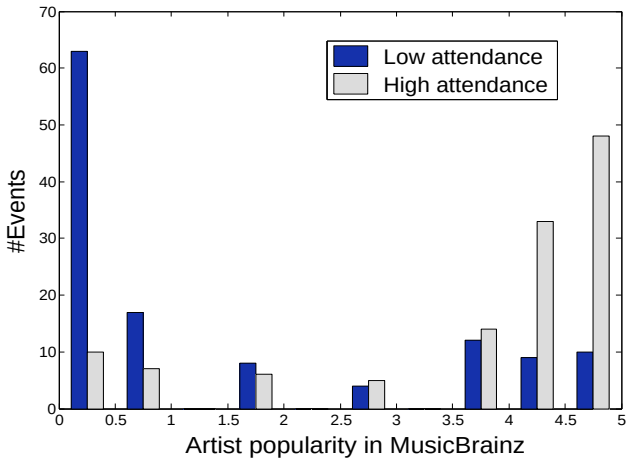

Figure 5.12: Correlation between attendance rate and artist popularity

In this section, we also address the following question: what are the events that prompt people to participate? Taking into account the high attendance rate, we distinguish between two kinds of events. The former contains the events featuring a significant number of artists, while the latter surprisingly includes those which are associated with few artists. To gain a deep insight, we invoke additional information about artists' popularity using the MusicBrainz open dataset. Results are illustrated in Figure 5.12. We clearly observe that the more artists are popular, the higher is the event attendance. Again, we endorse the benefits of the background knowledge coming from open datasets to understand these real facts.

5.4.2 User Profiling using Linked Data

User profiling is one cornerstone in personalization systems. However, the problem is that users are often reluctant to explicitly describe their interests. To solve this, one common solution is to rely on past user consumption of Web resources, considered as key elements to reflect the user interests. For example, a user, who attended an event, might have an interest in the related topics or artists. Assuming this fact, we attempt to create a user profile by relying on one directory, and then by exploiting the connection of this directory with the Web of Data. We particularly use the Last.fm directory since it hosts a large number of active users and provides functional API methods. We first constructed a user interest model considering solely the Last.fm annotations, and then we enriched them with DBpedia subjects (e.g., genres). As a modeling method, we have been inspired by the approach proposed in [142] where the profile is reflected by the annotations of items with which the user has interacted. Technically, this approach quantifies the user interests based on the Latent Dirichlet Allocation (LDA) [15], a topic modeling technique based on the co-occurrence of terms. In our scenario, these items can be represented by the artists involved in the events in which the user participated. Our approach is as follows: for each artist i, LDA generates a T-dimensional vector of topic proportions $\Theta_i = [\theta_i^1, \theta_i^2, ...\theta_i^T]$, where T is the number of topics. Then, we compute the variance of each topic dimension t over all the artists A: $\Theta^t = [\theta_1^t, \theta_2^t, ...\theta_A^t]$. The user interest scores over T topics are represented by the vector $[\Theta^1, \Theta^2 ... \Theta^T]$. For this experiment, we only retain 39 users (among 400 users) that explicitly expressed their interests by means of tag-count pairs on their Last.fm home pages. For each user, several steps are performed:

1. The interesting tags associated with a "topical" tag and with a high count are retrieved from the Last.fm user profile, thus building a ground truth.
2. We collect the tags of appropriate artists from Last.fm (using `dc:subject`) and DBpedia (using `dbpedia-owl:genre`) on which we apply the LDA-based approach. Then, we retain the tags of topics associated with high interest scores.
3. User interests are modeled based on three sources: (1) Last.fm tags; (2) DBpedia subjects; (3) the combination of Last.fm and DBpedia. Lastly, we compute the Cosine similarity between each generated model and the ground truth.

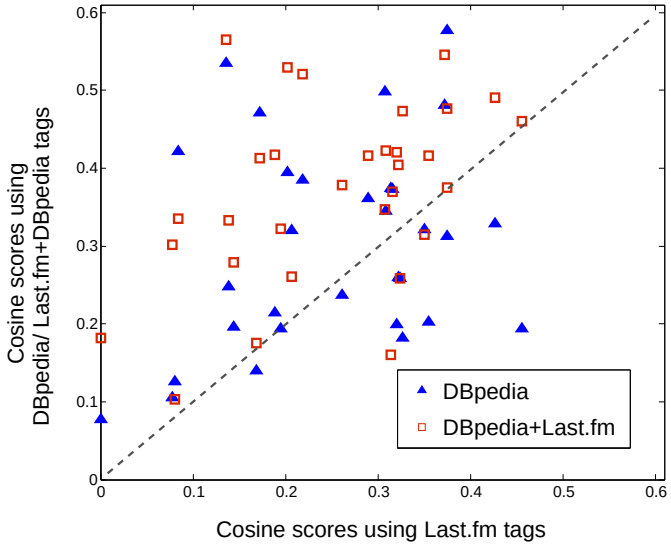

Figure 5.13: Comparison of user profiling based on Last.fm and DBpedia

Figure 5.13 depicts a scatter plot of Cosine similarity scores. We clearly observe that the combination of Last.fm tags with DBpedia improves the modeling of user interests. This is another application that emphasizes the benefits of Linked Data as means for introducing more coherent and qualitative data.

5.5 Conclusion

The development of Semantic Web applications has recently drawn more attention in the research field. With the steadily growth of Linked Data, rich structured information is made available in various domains. To consume this knowledge, efforts aim to reduce the gap between Semantic Web technologies and conventional Web applications. In this chapter, we have proposed some Web applications that consume event-centric Linked Data, and we have used existing solutions to access RDF data in conventional Web applications. In particular, dealing with the user interface was a challenging task given the expectations a user has with regards to the responsiveness of the application. This criteria is particularly conditioned on the querying performance of the triple store which is under investigation by the Semantic Web community. It also depends on the design strategy where the faceted navigation and simple queries are generally favored. Overall, our experience sheds lights on the strong dependency between the simplicity of the data model and its usability in Web applications. We have also highlighted some limitations to efficiently query RDF data using the Linked Data API. Finally, we have described some use cases showing the concrete advantages of using Linked Data for behavioral analysis and user profiling.

CHAPTER 6

Hybrid Event Recommendation

Recommendation in online services has gained momentum during the recent past years as a key factor to deliver personalized content. Reducing the information overload and assisting customers to make decision become part of primary concerns in the e-service area. To this aim, recommender systems attempt to provide efficient filters that decode the user interests, and optimize accordingly the information perceived. To help these systems predict items of interest, various clues are available ranging from user profile, explicit ratings, to past activities and social interactions. For more details, Appendix B.3 describes two popular recommendation techniques, namely the content-based recommendation and the collaborative filtering.

6.1 Challenges and Related Work

Integrating a recommender system in event-based services is a key advantage to attract people attending events and to promote face-to-face social interactions. Indeed, the event recommendation can draw on different features such as the user preferences (ratings, likes, etc.), the attended events (visited places, involved artists) or even the social co-participation. Broadly speaking, the decision making upon attending events depends on some restrictions such as time, location, category, popularity and friends' presence. However, the existing techniques (e.g., collaborative filtering and content-based methods) cannot cope at all with the complex inherent nature of such a decision. In addition, a recommender system is often application-specific, that is, to be tuned according to the item context (e.g., type, reasons to select an item, etc.). Another challenge in our work is that events often involve different topics (e.g., different genres in one musical concert). As a result, the user profile constructed based on the attended events may contain a wide variety of topics. This leads to topically diverse profile that may conceal the effective user interests.

In the research area of recommender systems, many approaches have been proposed to recommend movies, but few are the studies that deal with event recommendation. Events are particularly hard to recommend due to their short life time and the system often suffers from high sparsity of rating data. Some works have been proposed to overcome these issues and improve the recommendation accuracy. Cornelis et al. [26] built a hybrid approach within a fuzzy relational framework that reflects the uncertain information in the domain. The rationale behind is to recommend future events similar to those like-minded users have

liked in the past. However, this framework was not evaluated and there is no clear insight about its performance. Minkov et al. [96] followed the same rationale and proposed a low rank collaborative method to predict the rating of future events. They highlight the performance of the collaborative filtering over the content-based system. Still, their approach was more tailored to recommend scientific talks in the same building and there is no consideration of the geographical constraint. Some other systems have been developed such as "Pittcult" [78] and "Eventer" [61] that position the user within a social network and leverage the trustworthiness between users. Such a feature is valuable for recommendation, but it is not available in many systems. Finally, a user centered evaluation [35] showed that the straightforward combination of CF and CB recommendations outperforms both individual algorithms on almost qualitative metrics such as accuracy, novelty, diversity, satisfaction and trust. Other interesting related works are the recent studies that harness the power of Linked Data in recommender systems. For example, Di Noia et al. [32] use the Linked Data as the only background knowledge to recommend movies. They highlight the performance of the system that exploits ontology-based data representation compared with the keyword-based representation. Still, there is no deep exploitation of the latent similarity that may exist between movie attributes (e.g., two similar actors).

To tackle the challenges of event recommendation, we propose a hybrid system based on Semantic Web technologies [70]. Our belief is that a structured representation presents one solution to cope with the complexity of event-specific characteristics. This modeling will ensure a more straightforward way to explore and reason over the data. It makes possible to ask complex queries, for example, to retrieve events involving the same artist within a specific geographical area. In addition, the semantic model empowers the enrichment of event descriptions with additional information from Linked Data. Such enrichment can provide valuable inputs for the content-based recommender system as has been proved in [32]. As a second step, we propose to quantify the user interests based on topic modeling technique. The objective is to detect the user propensity towards specific topics. It will be integrated in the recommender system in order to control the impact caused by the diversity of a user profile. Finally, we exploit the collaborative participation assuming that the social information about "which friend will attend an event" plays an important role in decision making. In this work, we mainly investigate the extent to which the data enrichment, the social information and the user interests modeling can improve the system performance.

6.2 Content-based Recommendation using Linked Data

The principle of content-based (CB) recommendation is to suggest new items similar to those a user liked in the past. The similarity between items is computed based on the descriptive features of the item using a distance measure such as Cosine similarity, Pearson correlation and Latent Semantic Analysis [76]. The most common representation of the item

is the keyword-based model, in which attributes are represented by weighted vectors of keywords usually computed by TF-IDF scheme (term frequency/inverse frequency). To build such a profile from unstructured data, feature extraction techniques are needed to shift the item description from the original representation to a structured form suitable for next processing (e.g., keyword vectors). This task becomes straightforward by the use of Semantic Web technologies. CB recommender systems can greatly benefit from the ease of ontology-enabled feature extraction, and the availability of Linked Data covering different domains to enrich the item profile. In the following, we explain how to compute the items similarity in Linked Data.

6.2.1 Items Similarity in Linked Data

In order to compute the similarity between items in Linked Data, we resolved to apply the approach proposed by Di Noia et al [32]. The key idea is that semantically similar items from RDF graph are the subject of two RDF triples having the same property and the same object (where a triple=<subject,property,object>). The intuition behind is that: *if two subjects are in the same relation to the same object, this is evidence that they may be similar subjects*. Technically, the approach is based on an adaptation of the classic Vector Space Model (VSM) [125], a well-known technique in Information Retrieval (IR). In this model, similarity between documents and queries is computed using their representative t-dimensional weighted vectors of discriminating terms. The application of VSM in RDF graph projects the Linked Data to 3-dimensional tensor where each slice represents an adjacency matrix corresponding to one property in the ontology. Indeed, the Linked Data network can be defined as a graph $G = (V,E)$ where V is a set of resources and E is the set of properties between resources in V. For each property p in the set E, the related adjacency matrix presents the linkage between the subjects (on the rows) and the objects (on the columns) from V via p. Then, a non null weight is assigned to each entry $X_{i,j,p}$ in the tensor for each existing triple $< i^{th}$ subject, p^{th} property, i^{th} object$>$. Figure 6.1 shows an example of tensor slices related to some properties, namely: `lode:atPlace`, `lode:involvedAgent` and `dc:subject`.

Assuming that the properties are semantically independent, we would be able to compute the similarity between events according to each property separately. The representation of an event e_i according to the property p is a t-dimensional vector indexing the terms/objects related to e_i via p. The TF-IDF weight of each object o is:

$$w_{o,i,p} = f_{o,i,p} \cdot log\left(\frac{N}{m_{o,p}}\right) \tag{6.1}$$

where $f_{o,i,p} = 1$ if a link exists between the node e_i and the object o via the property p, otherwise $f_{o,i,p} = 0$. N is the total number of events in the dataset, $m_{o,p}$ is the number of events linked to the object o via the property p. Then, the similarity between two events e_i and e_j according to the property p is computed using Cosine distance between their representative

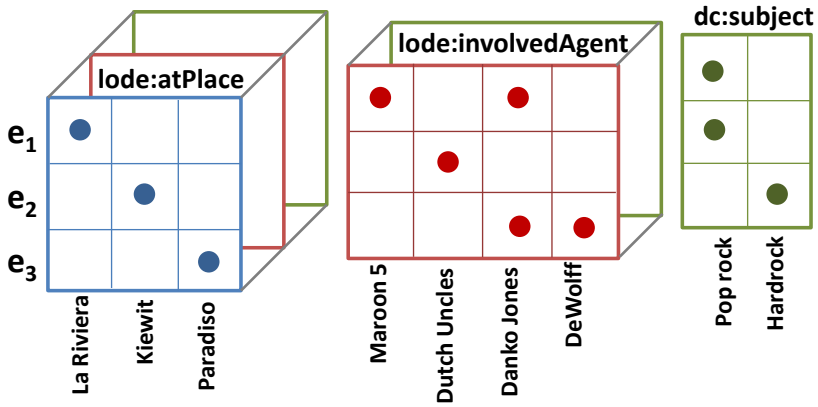

Figure 6.1: Tensor slices of some event properties (place, agent and subject)

vectors as following:

$$sim^{p}(e_i, e_j) = \frac{\sum_{r=1}^{t} w_{r,i,p} \cdot w_{r,j,p}}{\sqrt{\sum_{r=1}^{t} w_{r,i,p}^2} \cdot \sqrt{\sum_{r=1}^{t} w_{r,j,p}^2}} \tag{6.2}$$

This approach can be applied to detect similarity between subjects or objects of RDF triples. It has been successfully used to recommend movies and to improve the quality of content-based system [32]. However, it is still limited when the adjacency matrix is very sparse such as the case of matrices associated with the properties `lode:atPlace` and `lode:involvedAgent`. In fact, such predicates are characterized by the diversity of their object values, thus considered as discriminant properties. For instance, the t-dimensional vector related to `lode:atPlace` property has only one non-zero weight since an event is typically held at only one venue.

6.2.2 Similarity-based Interpolation

In order to mitigate the sparsity of the adjacency matrix, we propose to interpolate fictitious values based on the similarity of objects. Thus, we initially introduce a discriminability metric (i.e., discriminant power) to detect which properties are associated with highly sparse matrices. The discriminability metric is defined as follows:

$$Discriminability(p) = \frac{|\{o \mid t =< s, p, o > \in G\}|}{|\ t =< s, p, o > \in G\ |} \tag{6.3}$$

where G is the RDF graph, t is the triple representing the link between the subject s and the object o via the property p. This formula quantifies the discriminability by the number of different object values on the target property. For instance, from a set of 1700 events (related to 10,323 agents, 627 places and 5,758 subjects), we found a discriminability score of 0.64 for the `lode:involvedAgent` and 0.45 for the `lode:atPlace`, while it is only equal to 0.10 for the `dc:subject` predicate. Furthermore, similar events are not necessary occurred at the same location or featuring the same performers. In order to reduce the discriminability

impact, we interpolate fictitious weights in the adjacency matrix based on the similarity between objects as depicted in Figure 6.2.

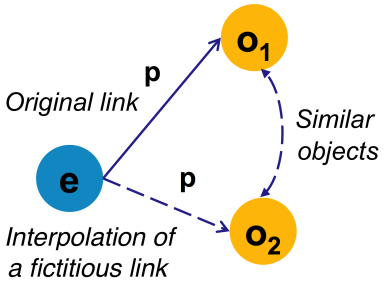

Figure 6.2: Similarity-based Interpolation

More precisely, if an object o_k is similar to another object o_h, and if both $f_{o_h,i,p} = 1$ and $f_{o_k,i,p} = 0$, then $f_{o_k,i,p} = sim(o_k, o_h)$. Note that $f_{o_k,i,p}$ reflects the strength of the fictitious link which associates the event e_i with the object o_k via the property p. If the object o_k is similar to more than one object originally linked to the event e_i, the weight $f_{o_k,i,p}$ will be equal to the highest similarity score. Thus, for each object o_k, the equation 6.1 becomes:

$$w_{o_k,i,p} = \max_{o_h \in H} sim(o_k, o_h) \cdot log\left(\frac{N}{m_{o_k,p}}\right) \tag{6.4}$$

where H is the set of objects originally linked to the event e_i. The intuition behind this formula is that: *if two subjects are in same relations to similar objects, this is evidence that they may be similar subjects*. We do not pay attention to how similarity between objects is computed. In fact, this measure depends on the nature of the object itself and there exist several existing techniques that can be used. In our case, we exploit the similarity scores between agents (i.e., artists) provided by third party services such as Last.fm, and we compute the normalized geographical distance between venues.

6.3 Event Recommendation

Different from a classic item, events occur at a specific place and during a period of time to become worthless for recommendation. Moreover, while a classic item (e.g., movie, book) continuously receives useful feedback, an event has few rating due to its transiency. In our dataset, these ratings are represented by the binary user-event attendance matrix which has a sparsity rate equal to 98% (i.e., a set of users attend a very limited number of events). As a solution, one can address event recommendation using CB recommender system that exploits the matching of event attributes with the user profile. This perfectly complies with the constraints considered when it comes to decide whether or not to attend an event. Metadata such as distance, time, topics and artists are important and influential factors in such a decision.

Still, the CB recommendation might suggest items with a limited diversity and overlook the social information regarding the question "which friend is going?". To reduce this gap, we propose to enhance its performance by enriching the content using Linked Data, and by improving the detection of the user interests. Then, we incorporate the social information using Collaborative Filtering (CF) method, thus producing a hybrid recommendation.

6.3.1 Content-based Recommendation

The CB recommender system suggests future events similar to those a user has attended in the past. We assume that there is a sufficient number of past attended events in the user profile to avoid the *cold-start* problem [1], which is out of the scope of the present work. In order to predict the participation of the user u to the event e_i, we combine the similarity values between events as following:

$$rank_{cb}(u, e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \; sim^p(e_i, e_j)}{\mid P \mid \; \cdot \mid E_u \mid} \tag{6.5}$$

where E_u is the set of past events attended by the user u, P is the set of properties shared between two events e_i and e_j, and α_p is the weight that reflects the contribution of the property p in the recommendation. The properties selected to compute the similarity between events are those which are related to the location, subjects (tags) and involved agents (artists). In contrast, the temporal information is not considered in this work and left for future study. Our belief is that temporality could be harnessed to index the recent events in the user profile, thus reducing the computation. Still, there is a need to deeply investigate the impact of the user profile reduction on the system performance.

Geographic Closeness

In recent research study [111], it has been shown that users generally tend to attend nearby entertainment events. This fact makes the location a valuable feature in event recommendation. In our approach, we need to measure the similarity between events according to the property `lode:atPlace`. Thus, we normalize the distance between two locations using a specific threshold θ which needs to be determined. As the user home is missing in our data, we measure the distance between attended events for each user as depicted in Figure 6.3. Note that the attendance rate becomes extremely low from $\theta = 80$ Km. We consider that this value is the normalization threshold from which the similarity between events is equal to zero according to the property `lode:atPlace`.

1. The problem to produce good recommendations for new users where nothing is known about their preferences

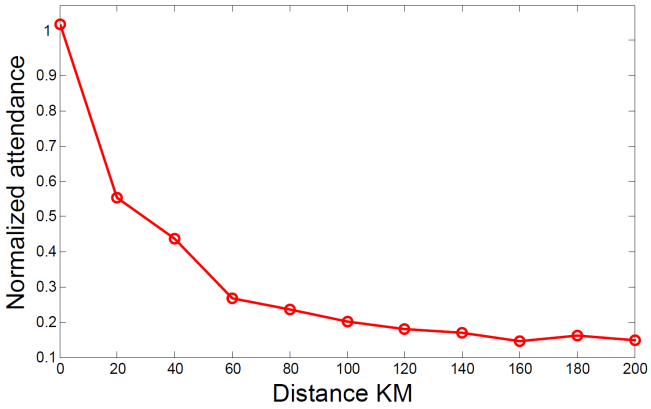

Figure 6.3: Normalized average attendance per distance

Enrichment with Linked Data

One method to enrich the item profile from Linked Data is to consume background information from DBpedia. The key advantage of DBpedia is the availability of semantically rich data in various domains. Using the mapping between EventMedia and DBpedia, we enrich the topics of events using the DBpedia topics (e.g., genres) related to involved artists. More precisely, we retrieve the categories associated with the property `dcterms:subject` of artists by simply querying the DBpedia SPARQL endpoint [2]. The reason behind our interest in DBpedia is that topics are accurately labeled and classified.

6.3.2 User Interest Modeling

One fundamental goal in the recommender system is to suggest new items that best fit the user interests. In our case, this is particularly difficult to achieve due to the presence of topically diverse events. In fact, the real-world social events can be classified into large set of categories ranging from large festivals and conferences to small concerts and social gatherings. When attending an event, the user might be interested in a specific show or artist or might have broad interests. In consequence, relying on event similarity according to the `dc:subject` property can be influenced by the topical diversity of tags related to events in the user profile. To alleviate this impact, we leverage the Latent Dirichlet Allocation (LDA) [15] for detecting the relevant user interests as previously described in Section 5.4.2.

Figure 6.4 illustrates the pipeline of the user interests modeling. For each event e_i having a set of tags, LDA generates a T-dimensional vector of topic proportions $\Theta_i = [\theta_i^1, \theta_i^2, ...\theta_i^T]$, where T is the number of topics and Θ_i reflects the semantic categories of the event. Then, we compute the variance in each topic dimension t over all the events E attended by a user $\Theta^t = [\theta_1^t, \theta_2^t, ...\theta_E^t]$. The diversity score of each corresponding user is the mean of the variances of all the topics dimensions (mean of $\Theta^1, \Theta^2 ... \Theta^T$).

2. http://dbpedia.org/sparql

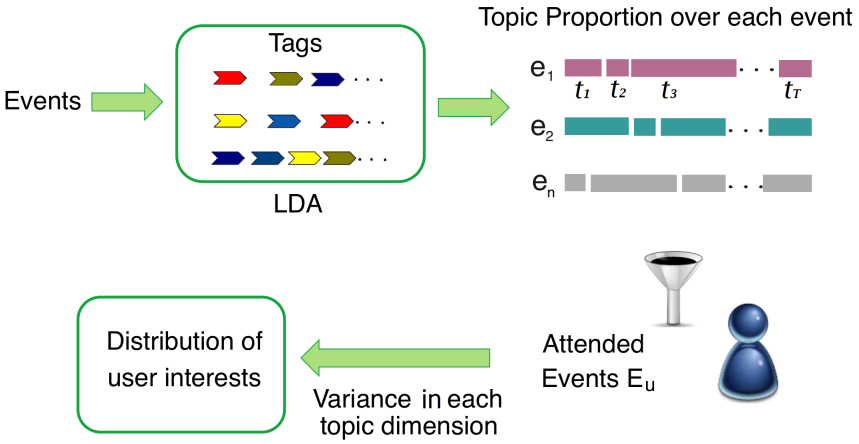

Figure 6.4: The pipeline of user Interests modeling

This approach as introduced by Wu et al. [142] has been originally designed to study the diverseness of individual tastes. But, we think that it is also helpful to detect user's propensity from a topically diverse profile. Indeed, events can be divided into two classes: those related to very few topics or those related to many topics. We consider that events in the first class are those which really exhibit the user interests. Using the variance, we are able to detect high proportions within topic dimension given that this dimension is likely to also contain low proportions (i.e., events are not regularly distributed over the topics). As an example, Figure 6.5 shows the normalized diversity scores obtained from a sample of 1,000 Last.fm users. In Figure 6.4(a), it is shown that most of diversity scores range from 0.3 to 0.5 indicating that users have relatively high interests in specific topics. The diversity scores near to 1 represent users having strong interests in very few topics such as the case of the user plotted in Figure 6.4(b). This user has a strong bias specifically towards the topic 9. Finally, the diversity scores close to zero generally represent the users associated with few attended events (i.e., the cold-start problem).

To take into account the effective user interests in a recommender system, we give emphasis to the events which are more likely to correspond to the user interests. We assign different weights β to the events included in the peaks of interest, and to those which are out of these peaks. These weights are then estimated using training methods. The content-based recommendation is extended as following:

$$rank_{cb++}(u,e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \, \beta_p \, sim^p(e_i,e_j)}{|\,P\,| \cdot |\,E_u\,|} \tag{6.6}$$

where $\beta_p = 1$ if the property p is different from `dc:subject`, otherwise the $\beta_{subject}$ is an estimated value depending on whether the event e_j corresponds to the user interest or not.

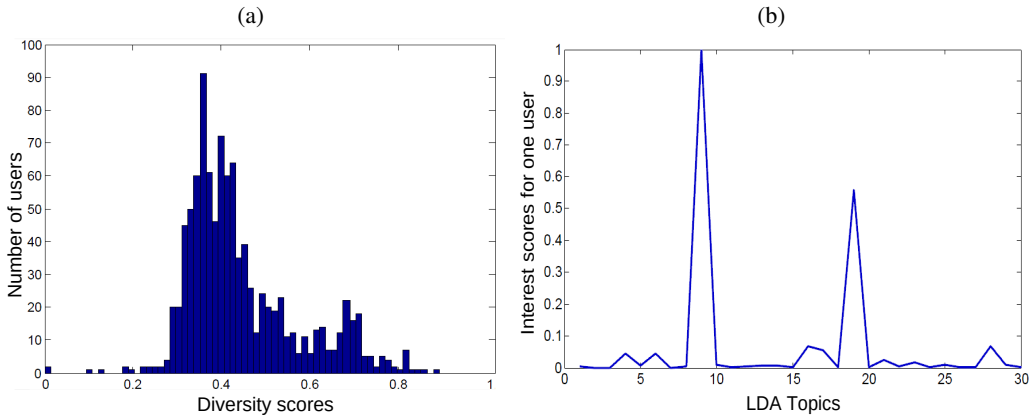

Figure 6.5: Distribution of topical diversity scores with T = 30: (a) for all the users; (b) for one specific user.

6.3.3 Collaborative Filtering

A form of social interactions is the collaborative participation such as co-authoring a paper or co-attending an event. In [87], Liu et al. highlight the existence of an offline social network built from the co-attendance of social events. Accordingly, we consider that two users involved in the same event can potentially have a stronger tie than other users. Our assumption is that the more events in which users involve, the stronger is their tie. Thus, the co-attendance can be a clue to provide information at first glance about which "friends" will attend an event. Moreover, our dataset contains the users' RSVP that express their intent to join social events, which can be exploited to predict unknown intents. However, unlike the traditional user-based collaborative filtering (CF), we decide to consider not only the similarity between users, but also the contribution of a group of friends. We define the following formula as the prediction that a user u_i will attend an event e based on the RSVP of his/her co-attendees (i.e., users who have attended past events with the user u_i):

$$rank_{cf}(u_i, e) = \frac{\sum_{j \in C} a_{i,j}}{|C|} \cdot \frac{|E_i \cap (\cup_{j \in C} E_j)|}{|E_i|} \tag{6.7}$$

where C is the set of co-attendees who will attend the event e, E_i is the set of attended events by the user u_i, and $a_{i,j}$ is the fraction of common events between the users u_i and u_j by the cardinality of E_j. Note that the weight $a_{i,j}$ reflects whether the most of events which are attended by the user u_j are also attended by the user u_i. The rationale behind this formula is two-fold: (1) in the first part, we consider the contribution of each co-attendee individually; (2) in the second part, we consider the co-attendees as a group of friends, and we assume that the more events they attended together with the user u_i, the more strongly is their relationship.

6.3.4 Hybrid Recommendation

To combine the predictions of both CB and CF recommender systems, we propose a weighted hybridization using a linear combination of predicted rank. Taking into account the user diversity and combining the equations (6.6) and (6.7), we propose the following function:

$$rank(u,e) = rank_{cb++}(u,e) + \alpha_{cf}\, rank_{cf}(u,e) \tag{6.8}$$

where α_{cf} is the weight of CF method estimated in conjunction with the weights of CB features using optimization functions for training the system.

6.4 Experiments and Evaluation

In this section, we carry out a set of experiments measuring the precision and recall metrics to assess the contribution of each step in our approach, and to evaluate the performance of our system compared with existing approaches.

6.4.1 Real-world Dataset

We use the EventMedia dataset and particularly the Last.fm directory which contains a large number of active users. Using SPARQL, we collected 2,436 events, 481 active users whose the attendance rates are within [15, 50], generating 12,729 distinct consumption (i.e., user-event pairs). This set of events are related to 14,748 distinct artists, 897 locations and 4265 tags (music domain). For the evaluation, we use a test set containing the most recent 30% of the consumption and a training test with the remaining 70% consumption. Then, we measure two metrics used in top-N recommendation task: Precision is the ratio of correctly recommended items and the length of the recommendation N; Recall is the ratio of correctly recommended items and the total number of future consumption. Precision and Recall are computed at different N values.

6.4.2 Learning Rank Weights

To learn the weights of our prediction function, we first test the linear regression with gradient descent that minimizes the least-squares cost function. Then, we use two evolutionary computation methods, namely the Genetic Algorithm (GA) and the Particle Swarm Optimization (PSO) motivated by their success in a wide range of tasks (details in Appendix B.2).

To apply GA in our approach, a chromosome is represented by a vector of the coefficients that need to be estimated. Each chromosome is then evaluated using a fitness function. This function aims to minimize the prediction error and thus maximize the precision of results. Table 6.1 shows the GA setting parameters.

Population size	Iterations	crossover	mutation
30	80	0.9	0.01

Table 6.1: Setting of GA parameters for event recommendation

As for PSO, a particle is represented by a vector of weights and the fitness function aims at maximizing the precision. Table 6.2 shows the PSO setting parameters.

Population size	Iterations	c_1	c_2	inertia
30	80	1.494	1.494	0.729

Table 6.2: Setting of PSO parameters for event recommendation

6.4.3 Experiments

First, we show in Table 6.3 the sparsity rates of adjacency matrices according to each property. We can see the efficiency of our method to discover latent similarity between events especially for discriminant properties. This highlight the importance of the similarity-based interpolation and the enrichment using Linked Data. Unlike the keyword-based recommender systems, the interpolation is straightforward in our system thanks to the ontology-based data structuring.

Task	location	agent	subject
(1)	0.9942	0.9174	0.3175
(2)	0.6854	0.7392	0.2843

Table 6.3: Sparsity rates of the adjacency matrices before (1) and after (2) the similarity-based interpolation (for location and agent) and data enrichment with DBpedia (for subject)

Second, we assess the performance of the training methods to learn the coefficients α in the hybrid recommendation function (Equation 6.8). Note that for this experiment, we do not include the user interests model and we set the $\beta_{subject}$ equal to 1. This experiment aims to rather compare the performance of optimization methods. Figure 6.6 shows the Precision and Recall curves. It is obvious that setting all coefficients equal to 1 achieves the worst performance because there is no adaptive optimization. It is also shown that precision optimization methods (GA and PSO) yield considerably better results compared with error (RMSE) minimization method based in linear regression. This has been also proved in recent work [27] showing that methodologies based on error metrics do not necessarily improve the accuracy of top-N recommendation task. One given explanation is that the RMSE-oriented methods rely only on known ratings to train the system and do not consider the unrated items. Finally, Figure 6.6 highlights the better performance of PSO compared with GA algorithm. We observed a faster convergence to the optimal solution in PSO compared with GA which

needs more iterations. This is due to the inherent behavior of PSO where the evolution is only guided by the best particle. In contrast, the GA evolution is guided by a group of solutions in which even weak candidates continue to survive after some iterations. In the following, we use the PSO algorithm to train the system.

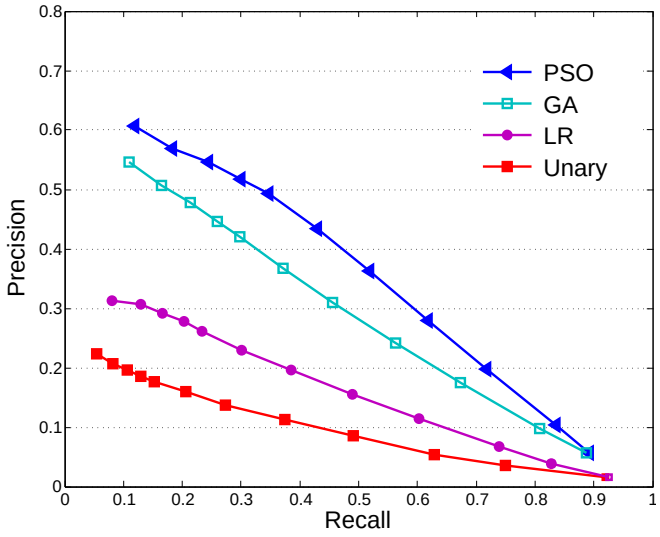

Figure 6.6: Recall and Precision using different approaches to estimate the vector α

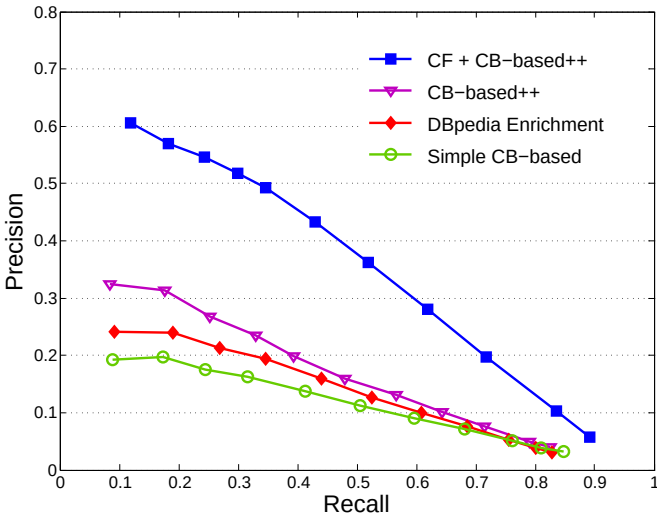

Figure 6.7: Evolution of the recommendation accuracy by incorporating the DBpedia enrichment, user diversity (CB-based++) and collaborative filtering (CF)

To gain insight into the influence of the different steps in our approach, we examine the evolution of the system performance by incorporating in each experiment (by order) the enrichment with DBpedia, the user interest modeling and the collaborative filtering. Results are illustrated in Figure 6.7. We can observe that enriching data with DBpedia slightly improves

both precision and recall. Indeed, introducing more coherent and qualitative data is one solution to reduce the noise that can be found in the collective knowledge of crowd tagging (e.g., Last.fm tags). Then, the user interest modeling also enhances the system performance. For this experiment, we fix the coefficients α obtained with PSO. Then, we train the system to compute the coefficient $\beta_{subject}$ which depends on the peaks of the user interests. As a result, we obtain $\beta_{subject}$ equal to 0.4 when the event is not included in an interest peak, and $\beta_{subject}$ equal to 1.6 (4 times more) otherwise. This proves the importance to clearly discern the user interests when the user profile contains diverse topics. Finally, combining these results with the collaborative filtering notably increases the recommendation accuracy. Our belief is such an improvement is perfectly tangible with the use of a real-world dataset. According to the user centered study presented by Fialho et al. [40], social information such as people and friends who are attending an event has strong priority and influence on decision making.

Lastly, we assess the extent to which a hybrid event recommendation outperforms the existing collaborative filtering based on matrix factorization to detect latent factors from the user-item matrix. We compare our system with the traditional user-based CF and the Probability based Extended Profile Filtering (UBExtended) proposed by Pessemier et al. [110] to recommend events. This method employs a cascade of two user-based CF systems aiming to recommend the most consumed (i.e., popular) events. The rationale behind is that the probability to consume an event is proportional to the current popularity of the event (i.e., has attracted many users). The comparison results are depicted in Figure 6.8. It is shown that the UBExtended method outperforms the user-based CF algorithm. Still, the hybrid recommendation exhibits the best results in terms of precision and recall. This is due to the benefits of hybridization as has been proved in other research studies [119].

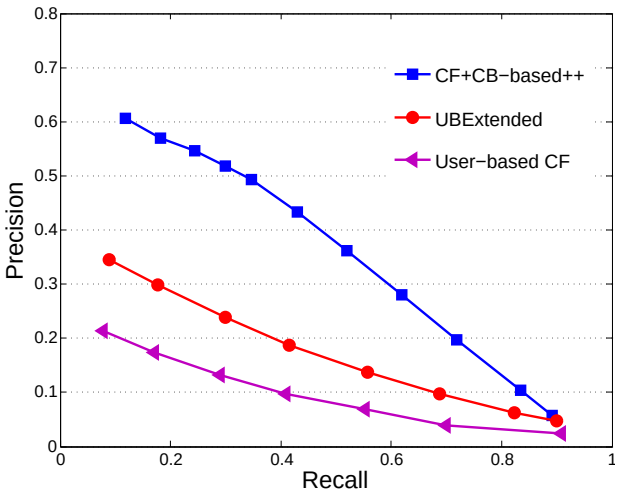

Figure 6.8: Comparison of hybrid event recommendation with pure CF algorithms

6.5 Conclusion

In this chapter, we have presented an approach for event recommendation combining both CB and CF advantages, and using Linked Data to enrich the event profile. In addition, we have proposed an approach to model the user interests and to overcome the topical diversity in the user profile. The evaluation particularly highlights the importance of social information and the user diversity model to enhance the system performance. In the future, we plan to take into account other significant features such as event popularity and temporal indexing of recent consumption.

CHAPTER 7

Overlapping Semantic Community Detection in Event-based Social Network

Community detection has recently received a great attention as a major topic for analyzing social networks. It aims to uncover the substructures within a network revealing which users are likely to have common interests, occupations and social properties. The information about the underlying communities can be of a great benefit for many tasks such as people recommendation, information diffusion and personalization. For instance, a personalization system can be based on user's community to gain more knowledge about his/her behavior [128, 109]. It has been also proved that the substructures within a network provide new powerful means of recommendation and collaborative filtering [72, 109].

Today's people use event and media websites to interact together either online by sharing microposts and photos or offline by attending events. Thus, many social connections can be formed and strengthened during social events, thus forming an event-based social network. It is ideal to analyze this network along with the information about users and events in order to discover semantically coherent communities. A person is naturally interested in many events which may be associated with multiple topics. It is thus more reasonable to divide users into overlapping semantic groups instead of disjoint ones. A semantic community as defined in [21] is "a set of nodes which show a common interest in a given topic and are organized according to a particular topology". Each community has a specific interest on general topic, for example politics or education. An efficient community detection algorithm should cluster individuals who are closely connected and share common interests. In this chapter, we propose an approach based on Semantic Modularity Maximization (SMM) to detect overlapping semantic communities in event-based social network.

7.1 Challenges and Related Work

Broadly speaking, community detection is dividing the vertices into groups such that there is a higher density of links within groups than between them [23]. It has attracted attention in recent years leading to several interesting studies. Most of existing methods focus on network topology and analyze the links between users. They attempt to detect disjoint

communities by optimizing different link-based objectives. The most popular of these methods aim to maximize the quality metric known as modularity Q introduced by Newman and Girvan in [103]. This metric is high when there are dense connections (edges) between nodes within communities but spare connections between nodes in different communities. Some of modularity maximization methods are based on greedy agglomeration [23] and spectral clustering [105]. These works exploit the structural properties and the linkage patterns within a network and they have been successfully used in some applications. However, they generally detect communities in which users have different interests as no consideration of the semantic dimension was made. It is difficult to interpret the nature of users' relationships grouped within such communities [28, 146]. Merging therefore the semantic information with the linkage structure is essential to produce meaningful and interpretable communities.

As efforts to detect semantic communities, some studies have exploited the topic modeling techniques such as pLSA [52], LDA [15] and AT (Author-Topic Model) [133]. For example, the work in [82] makes an analogy between the LDA document-topic-word and the user-topic-websites. The idea behind is that users sharing similar online access pattern tend to belong to the same semantic group. This method primarily relies on the link information in a social graph, and it is only efficient when regular interaction patterns can be detected. Another technique called Community-User-Topic (CUT) [147] extends the LDA model to detect communities using the semantics of content. As a result, communities are represented as random mixtures over users who are associated with a distribution of topics. This method does not consider the link information assuming that community members only share common topics. Obviously, both methods can not be applied in real-world social networks where users' memberships are conditioned on their social relationships as well as their shared interests [146].

Recently, some works start to investigate the combination of both content and link information. For example, the generative Bayesian model (Topic User Recipient Community Model) presented in [122] combines discussed topics, interaction patterns and network topology to detect semantic communities. In [146], Zhao et al. have proposed an approach based on a modified k-means algorithm (EWKM-Entropy Weighting K-means) to partition social objects (e.g., mails, events, etc.) into semantic clusters. Each semantic cluster contains members who interacted with similar social objects. Then, a modularity maximization method is applied in each semantic cluster, which is in turn divided into clusters considered as semantic communities. In our work, we made analogy between these social objects and events and we extensively compare our algorithm with this approach called as EWKM-based method in the rest of this chapter.

On the other side, community detection in event-based social network has been the subject of some research works. For example, Liu et al. [87] proposed an approach based on an extended Fiedler method to consider both the online and offline interactions. This method seems efficient to detect cohesive communities, but it is still a link-based method and no

attention was paid to semantic dimension. In [83], the Event-based COmmunity DEtection (ECODE) algorithm enriches the event-to-event network with virtual links based on participants' semantic similarity computed using the users profiles. Virtual links aim to enhance connectivity between events sharing users having semantically similar interests. ECODE computes the similarity between events based on their shared physical and virtual links, and then clusters them using a hierarchical method. The community of users associated with each event cluster is generated using an assignment function. In the same context, Wang et al. [138], proposed a community detection approach in location-based social network (LBSN). Their approach exploits different features such as user social similarity and venue-user similarity, and uses an edge-centric co-clustering which simultaneously discovers overlapping groups of venues and that of users. To sum up, these different studies provide important insight into detecting communities in event-based social network. However, in such networks, none of these works aim to maximize both connectivity strength and semantic purity.

7.2 EBSN: Event-based Social Network

Websites such as Lanyrd, Last.fm, Flickr and Twitter host an ever increasing amount of event-centric knowledge maintained by rich social interactions. These interactions form two types of event-based social networks (EBSN). The former is represented by the typical online activities such as sharing media and exchanging thoughts about events. The latter captures the face-to-face social interactions reflecting the offline co-participation in the same events. In other words, EBSN is a heterogeneous social network underlying the co-existence of both online and offline social links [87]. Meanwhile, the information about these social interactions are spread over multiple websites. For example, people tend to mostly use media platforms to share photos about events, whereas they express their intent to attend events in online event directories. Exploiting the overlap of these distributed websites is a key advantage to analyze the social networks. In this section, we describe how to construct an event-based social network using offline and online interactions and we highlight some of their interesting properties.

7.2.1 EBSN Definition

Based on user activities in social media, we define the following EBSNs making difference between online and offline networks. Different from the definition given by Liu et al. [87] based on friendship links in social networks, we consider that the online EBSN is constructed by solely capturing the online interactions such as sharing microposts and photos about the same events. Similarly, the offline EBSN is constructed by considering the physical co-participation in the same events. In particular, we exploit three EBSNs in EventMedia:

- **Last.fm EBSNs.** In Last.fm, there are two networks: the online EBSN is built based on the activity of co-commenting events, whereas the offline EBSN is based on the RSVP provided by users.
- **Flickr Online EBSN.** We exploit the activity of co-sharing photos related to the same events (provided by Last.fm) and we build an online media EBSN.
- **Twitter Online EBSN** Similarly, we exploit the co-tweeting activity about the same conferences (provided by Lanyrd) to build another online media EBSN.

7.2.2 Spatial Aspect of Social Interactions

In the following, we investigate how far from their homes people interact within the offline and online EBSNs. We compare the geographical distance between the user's home and the locations of events. As the user's home location is not explicitly provided by Last.fm, we infer it using the average of most frequent positions of attended events. Results are depicted in Figure 7.1 based on a random set of events and their associated users.

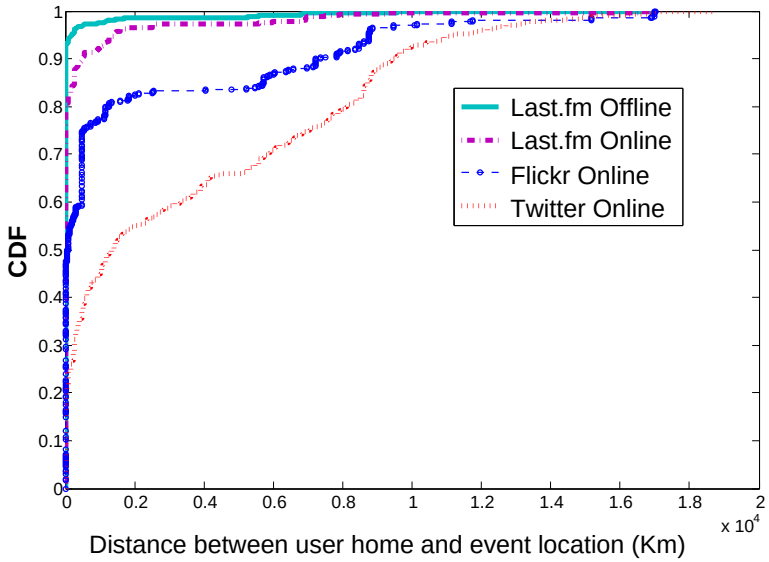

Figure 7.1: Locality of user activities in offline and online EBSNs

We observe that 95% of users' activities in offline network are within 100 km from their homes. This rate slightly decreases in online Last.fm EBSN indicating that, even on the Web, people tend to interact about nearby events. This aspect has already been proved in an existing study [87] showing that users' activities in EBSNs are much more location constrained compared with location-based social network. In contrast, the online interactions in media-based EBSNs seem to be less conditioned on event location. The reason behind can be two-fold: (1) the nature of sharing activity which is more present in media platforms than event directories, and the users are generally non-uniformly spread; (2) the type of events

(conference) indicating that people tend to travel far from their home for business purpose rather than for entertainment activity (musical concert). Based on these findings, we decided to perform community detection using conferences from different cities in Lanyrd, whereas we only focus on a specific geographical location in Last.fm and Flickr.

7.2.3 User Participation

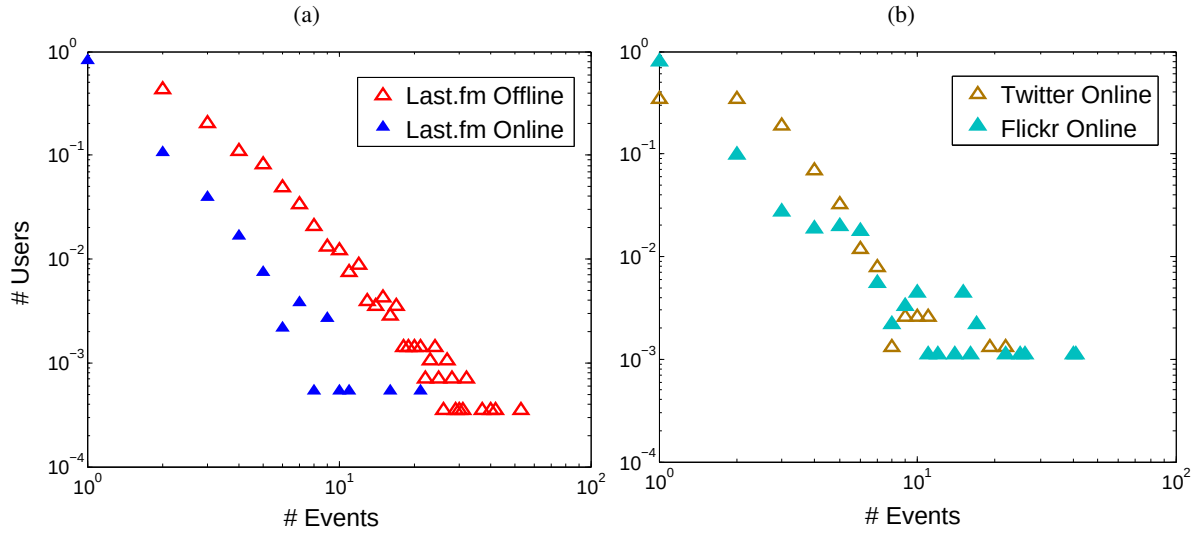

Figure 7.2: Number of participants per event in (a) Last.fm offline and online EBSNs and (b) Flickr and Twitter online EBSNs

To gain insight into some EBSN properties, we study the user participation behavior. As shown in Figure 7.2, the results resemble a power-law distribution indicating that most of users are associated with few events. Similar results have been highlighted in other works studying the event attendance behavior [87, 46]. In particular, there are 81% of users who are associated with only one event in Last.fm online EBSN, and 76% of users sharing photos of only one event in Flickr EBSN. This fact can be also drawn in Table 7.1 if we compare the number of shared media with the number of users. During the evaluation, we will show the influence of the user participation distribution on community detection.

7.3 SMM-based Community Detection

In this section, we first describe our graph model, and then we present our approach based on Semantic Modularity Maximization (SMM) proposed to detect overlapping semantic communities.

7.3.1 Graph Modeling

Taking into account the users, the events and their related attributes, we consider the fourth-tuple graph $G =< U, S, T, E >$ for both online and offline EBSN where U is the set of users, S is the set of social events which are in turn associated with a set of tags T, and finally E is the set of undirected edges. E contains two kinds of links $E = E_{US} \cup E_{UU}$:

- E_{US} is formalized as $E_{US} = \{(u,s)|u \in U, s \in S\}$ and denotes the links between users and events.
- E_{UU} is the set of links between users (i.e., a link represents the co-participation in same social event), formalized as $E_{UU} = \{(u_i,u_j)|u_i \in U, u_j \in U\}$.

In this graph, each user can be represented as a binary vector of related events, and each event can be represented as a binary vector of related users. Similar way is applied using the event-tag relationship.

7.3.2 The SMM Approach

SMM stands for Semantic Modularity Maximization. In our SMM approach, we first compute the similarity between events based on both the link and semantic information. Then, we employ a hierarchical clustering algorithm that groups events into semantic clusters. This clustering aims to maximize a newly defined quality function called "semantic modularity". Finally, a link-based function determines the effective user attachment to each event cluster, thus generating overlapping communities of users.

7.3.2.1 Similarity Computation

In EBSN, the overlapping communities sharing common interests can be detected by clustering similar users together. However, user-based clustering needs high computational time due to the large number of users. One solution is to employ an event-based clustering from which communities of users are formed based on event-user link. Still, the event similarity should reflect both the linkage and semantic properties. To solve this, we use the notion of *Homophily* which is observed in many social networks [92]. Homophily refers to the tendency of persons to be associated with other persons that share similar characteristics. In other words, users involved in the same events have a higher likelihood to share similar interests and get connected. Similarly, tags associated with the same events are more likely to be semantically similar. This implies that similar events are sharing both like-minded users and semantically similar tags. Thus, we cluster events based on their similarity both in the user space and in the semantic space. In the event-user network, events can be represented as a vector of users, and users can also be represented as a vector of events. To reduce the dimension of the event-user matrix, we decided to use one popular technique in dimensionality reduction called Singular Value Decomposition (SVD). The idea is to represent events in a latent user space using an orthogonal basis. Given a matrix A, the SVD is the product

$U\Sigma V^T$ where U and V are the left and right singular vectors and Σ is the diagonal matrix of singular values. An event vector $\tilde{e}_i$ in the latent user space can be represented as follows:

$$\begin{aligned} & e_i(u_1,u_2,...,u_n) = \{U\Sigma\}_i V^T \\ \Leftrightarrow \quad & e_i(u_1,u_2,...,u_n) = \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k)V^T \\ \Leftrightarrow \quad & \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k) = e_i(u_1,u_2,...,u_n)V \\ \Leftrightarrow \quad & \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k) = AV \end{aligned}$$

In order to detect similar events that share like-minded users, we leverage the spectral co-clustering proved in [31] and indicating that only the top singular vectors, except of the first one, contain partition information. The algorithm first normalizes the event-user matrix as follows:

$$A_n = D_1^{-1/2} A D_2^{-1/2} \tag{7.1}$$

where the entries of the diagonal matrices D_1 and D_2 are respectively the event degrees and the user degrees (i.e., a degree is the number of connections the node has to other nodes). Then, applying SVD on A_n gives $A_n = U_n \Sigma_n V_n^T$. Only the top singular vectors (except of the first one) are selected from $V_n = (v_1, v_3, ..., v_n)$ to form the matrix $V_n' = (v_2, v_3, ..., v_m)$ where $m \ll n$. Finally, the event vector in the normalized user latent space can be written as follows:

$$\tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k) = A_n \cdot V_n' \tag{7.2}$$

Similarly, we represent events in the latent semantic space applying this method on the event-tag matrix. Recent experiments in text corpus suggests that the dimension m of V_n' depends on the corpus size and it was set between 50 and 1000 [75]. Indeed, small value of m is advantageous to remove noisy information. In our case, we set m equal to 200 for the user space and equal to 50 for the semantic space since there are more users than tags in our dataset. Afterwards, we use Cosine distance to compute the events similarity S_u in the latent user space and S_t in the latent semantic space. Finally, we combine the similarities as follows:

$$S_{sim} = \alpha\, S_u + (1-\alpha)\, S_t \tag{7.3}$$

where α is the parameter that controls the balance between user-oriented similarity and semantic-oriented similarity. In this approach, the pair-wise computation using Cosine distance can be reduced by selecting candidate solutions that only index the potentially similar events. Intuitively, these solutions are the events that share in common a minimum number of tags or users with the original event. Variants techniques can be used such as the Locality Sensitive Hashing (LSH) [41] or its variants (e.g., MultiProbe LSH [89]) which are popular high-dimensional similarity search methods. In ECODE algorithm [83], it has been shown

that the candidate selection was efficient to save a significant amount of computational time without affecting the communities detected. In the present work, the candidate selection was not considered since we use relatively small datasets in our experiments.

7.3.2.2 SMM-based Algorithm

To group similar events, a hierarchical agglomerative clustering is employed. It begins by assigning each data item to its own individual cluster. The two most similar clusters are merged together into a single cluster. This step is repeated until all the items are grouped into a single cluster, thus forming a hierarchy. As outlined in Algorithm 2, the most similar events s_i and s_j are clustered together forming a new event s_{new}. Then, we compute the similarities between s_{new} and the rest of events. This process is iterated until there is no significant increase of the quality function. This approach is based on hierarchical clustering which is advantageous compared with other methods such as k-means since the predefined number of clusters is not required.

Algorithm 2 SMM-based Algorithm

S: set of social events $s_1, s_2 ... s_i$
N_T: number of topics
S_{sim}: event similarity matrix
while Community Size>T **and** *SemQ* function increases **do**
  Merge the most similar events s_i and s_j into a new event s_{new}
  for each event $s_k \in$ S **do**
    $S_{sim}(s_{new}, s_k)$ = average($S_{sim}(s_{new}, s_i)$+$S_{sim}(s_{new}, s_j)$)
  end for
  Compute *SemQ*
end while

As our objective is to produce semantic clusters, we define a semantic oriented quality function which has the same rationale as Newman's modularity Q. This novel function is called semantic modularity (*SemQ*), and it aims to maximize the intra-similarities and minimize the inter-similarities of communities in the semantic space. Our SMM-based approach aims to guarantee semantically coherent topics in each cluster by maximizing semantic similarity within clusters, but minimizing it between clusters. To formalize the semantic modularity, we use the events similarity S_t computed in the latent semantic space and we compute the intra-similarities (IntraSem in Equation 7.4) and inter-similarities (InterSem in Equation 7.5) where C is the set of discovered clusters:

$$IntraSem = \frac{1}{|C|} \sum_{C_k \in C} \left(\frac{}{\sum\limits_{\substack{i,j \in C_k \\ j>i,\ S_t(i,j)\neq 0}} S_t(i,j)} \right) \tag{7.4}$$

$$InterSem = \frac{1}{|C|} \sum_{C_k \in C} \left(\sqrt{\sum_{\substack{i \in C_k,\ j \in C_l \\ l>k,\ S_t(i,j)\neq 0}} S_t(i,j)^2} \right) \tag{7.5}$$

Finally, the semantic modularity *SemQ* is defined as follows:

$$SemQ = IntraSem - InterSem \tag{7.6}$$

Note that the maximal value of *SemQ* stops the clustering process. In meanwhile, each detected cluster keeps in mind a minimal knowledge about the link information held by the event similarity in the user space.

7.3.2.3 User Assignment

The last step of our approach is to generate clusters of users from the detected clusters of events based on the event-user links. As the user may participate in many events, we generate overlapping semantic communities. However, a user may be weakly involved in one semantic cluster that not really reflects his/her interests. To address this problem, we propose to discover the effective user's memberships by computing his/her assignment scores. If the user u_i is a member of the community C_j, the assignment function is defined as follows:

$$A(u_i, C_j) = \frac{D_{C_j}(u_i)}{D(u_i)} \tag{7.7}$$

where $D_{C_j}(u_i)$ is the degree of the user u_i within the community C_j (i.e., the number of links of the user u_i with other members in the community C_j), and $D(u_i)$ is the global u_i's degree in the network. The user's membership to one community is determined if the related assignment score is higher than the average of assignment scores over all communities.

7.4 Experiments and Results

This section presents the evaluation of our SMM-based community detection approach applied on real-world datasets. We first describe these datasets followed by the description of the performance metrics and the obtained results.

7.4.1 Experimental Datasets

For experiments, we use the following datasets accessible online [1], and we show their network statistics in Table 7.1.

1. http://www.eurecom.fr/~khrouf/ebsn

EBSN	Users	Events	Tags	Edges	Density	ClustCoeff
Last.fm Offline	2847	915	272	95897	0.0237	0.1144
Last.fm Online	1729	470	248	9936	0.0067	0.398
Flickr Online	868	375	221	7071	0.0188	0.2624
Twitter Online	768	275	166	14237	0.0483	0.4852

Table 7.1: Some network statistics about the experimental datasets

Muscial Event (Last.fm and Flickr):

We have previously demonstrated that a very high fraction of social interactions for entertainment purpose exist between geographically close friends. Hence, we focus our analysis on events taking place in one city, and we select the capital "London" as it exhibits a significant number of users and events compared with other cities in EventMedia. Operationally, we query the EventMedia SPARQL endpoint to retrieve data, and we crawl additional metadata using the REST API of Last.fm and Flickr. Then, we pre-process the dataset as follows: First, we retrieve the events occurred in 2012 and 2013, and associated with media. Then, we remove the tags which are associated with very low frequency (less than 5) in order to reduce the semantic noise. Second, we remove the singletons of event-user pairs where an event has only one participant and this participant is involved in only this event. Finally, we obtain the following EBSNs: (1) an offline Last.fm EBSN containing 915 events, 2847 users and 272 tags; (2) an online Last.fm EBSN containing 470 events, 1729 users and 248 tags; (3) an online Flickr EBSN containing 375 events, 868 users and 221 tags. Note that the removal of singletons event-user pairs has significantly reduced the size of the online Last.fm and Flickr EBSNs indicating that users' activities in those networks are more sporadic and mostly present individual behaviors.

Conference (Lanyrd and Twitter):

Similarly, we use SPARQL queries to retrieve data from EventMedia, and Twitter API to retrieve additional information such as user's home location. As no tags were associated with the conferences, we extract them from the descriptions using tokenization and we remove stop-words. However, this method produced very noisy tags as some conferences are vaguely described (e.g., *The World is Changing, Is Your Company on Board?*). We also attempt to automatically process the related tweets. Still, many tags do not really reflect what is the conference about due to the presence of several noisy messages (e.g., personal status updates, opinions, etc.). To solve this, we manually label the conferences descriptions by selecting the most representative keywords. Due to this manual effort, we only keep the interesting conferences which are related with very active users. Finally, we obtain an online EBSN which contains 275 events, 768 Twitter users and 166 tags.

7.4.2 Topic Modeling

In order to assess the semantic purity in each cluster, we first need to detect the set of topics in each dataset. Thus, we decided to employ the popular topic modeling technique LDA [15] where we consider the events as documents. The use of LDA has led to coherent topics when dealing with Lanyrd conferences, but slightly ambiguous topics in case of Last.fm events. This is due to our manual labeling of conference descriptions where we carefully select qualitative tags. In contrast, Last.fm contains crowdsourced tags generated without moderator oversight and known to be less accurate. Moreover, the musical concerts may feature many artists related to different topics (i.e., genres), making more difficult to detect co-occurrences in LDA. The conferences, on the other hand, often target only one general topic (e.g., Semantic Web). To solve the topic modeling in Last.fm, we resolved to exploit the DBpedia classification of musical genres that may help detect the fuzzy similarity between them (e.g., death metal and deathcore). More precisely, we leverage the existing SKOS taxonomy [2] in DBpedia using the generalization relations such as `skos:broader` and `skos:narrower`. For each event tag, we retrieve the DBpedia genre in which the property `dcterms:subject` is related to this tag. Then, depending on the depth of this genre in the taxonomy, we select a more general or specific genre. The idea is to ensure effective topic or genre distribution with reasonable depth granularity. Tables 7.2 and 7.3 show few examples of topics detected respectively in Last.fm and Lanyrd. Note that we obtained 24 topics in Last.fm consisting of high-level musical genres, and 30 topics in Lanyrd where the optimal number of topics in LDA is determined based on the approach proposed by Griffiths et al. [43]. Finally, Figure 7.3 shows that many conferences have at most two topics, while this number slightly increases for musical events.

Topic	Example of Last.fm Tags
Heavy metal	metal alternative, progressive metal...
Pop	synthpop, powerpop, pop punk...
Electronic	indietronica, synthpop, folktronica...
Rock	hard rock, alternative rock, glam rock...

Table 7.2: Example of topics detected in Last.fm

Topic	Example of Lanyrd Tags
Education	learning, education, teaching, technology
programming	programming, language, python, library
Innovation	creativity, technology, business, future
Application	mobile, application, Web

Table 7.3: Example of topics detected in Lanyrd

2. http://dbpedia.org/page/Category:Musical_subgenres_by_genre

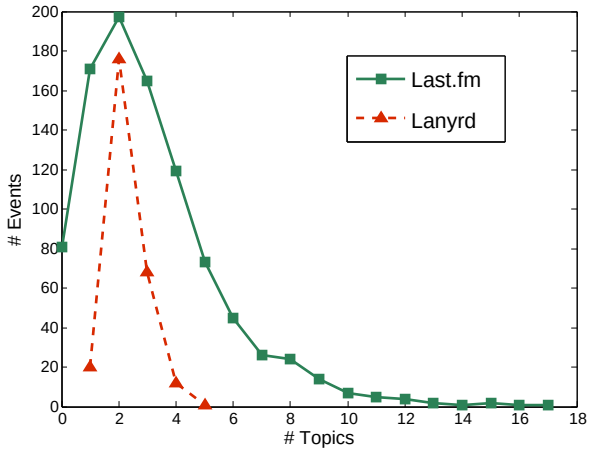

Figure 7.3: Histogram of the number of topics per event

7.4.3 Performance Metrics

To evaluate our approach, the performance metric should take into account the combination of both link and semantic information. We adopt the $PurQ_\beta$ metric introduced by Zhao et al. [146]. It has been inspired by the F-score measure that considers both the precision and recall metrics. $PurQ_\beta$ considers both the semantic purity and the Newman's modularity. First, we introduce the function that measures the semantic purity in each cluster as follows:

$$Purity_i = max_j \left(\frac{n_{ij}}{n_i} \right) \tag{7.8}$$

where n_{ij} is the number of tags belonging to the topic j and the cluster i, and n_i is the number of tags in the cluster i. The final score of the $Purity$ is the overall average of all the purity scores of communities. Yet, we observed during experiments that the measure of $Purity$ does not effectively detect the presence of clusters having low semantic purity. Hence, we decided to also examine the F_{purity} which is the fraction of clusters having $Purity_i$ higher than the average score $Purity$. Finally, the metric $PurQ_\beta$ combines the link and semantic metrics as follows:

$$PurQ_\beta = \frac{(1+\beta^2)(Purity \cdot Q)}{\beta^2 Purity + Q} \tag{7.9}$$

where Q is the Newman modularity [103] used to evaluate the goodness of a partition, ensuring that there are more edges within communities than between them. Then, the parameter β is used to adjust the weight of $Purity$ and Q. $\beta = 0.5$ means that $PurQ_\beta$ puts more emphasis on $Purity$ than Q. In contrast, $\beta = 2$ puts more emphasis on Q. The general behavior of communities is when the semantic $Purity$ increases, the modularity Q decreases, and vice versa.

7.4.4 Evaluation

We first evaluate how the coefficient α in Equation 7.3 affects the performance of our approach. Figure 7.4 shows the evolution of the semantic *Purity* and the modularity Q when α increases. It can be seen that the modularity increases if a high weight is assigned to event similarity in the user space. However, the semantic purity and the modularity do not evolve at the same scale. While Q slightly increases, *Purity* drastically decreases. Thus, good values of $PurQ_\beta$ can be obtained when $\alpha \in$[0,0.5].

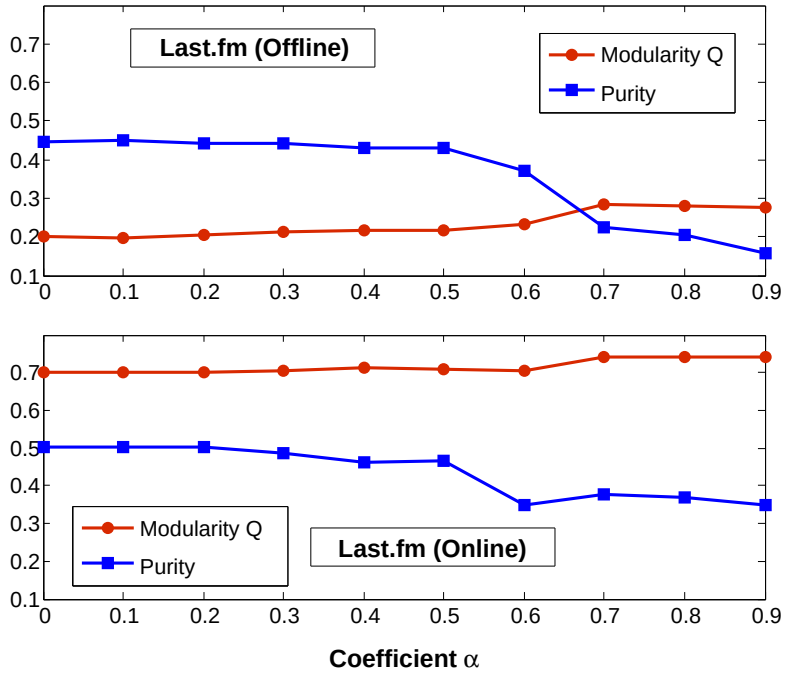

Figure 7.4: Evolution of the modularity Q and the semantic *Purity* with the parameter α

Then, we compare our SMM-based approach with some related work described in Section 7.1: (1) The popular modularity maximization approach based on greedy agglomeration (Greedy Q) where only the link information is considered [104]; (2) The Edge co-clustering approach (or EdgeCluster) proposed in [138] and applied on location-based social network. For this approach, we consider as features the user similarity in the event space and in the semantic space. Based on these features, Edge co-clustering uses k-means to cluster similar "user-event" edges. This method has been evaluated only on two small datasets as it requires a very large computation time; (3) The ECODE algorithm which introduces the concept of content-based virtual links in the event-to-event network, clusters together similar events sharing high physical and virtual links, and uses an assignment function to produce communities; (4) The EWKM-based method based on two-step clustering: k-means clustering of similar events, and modularity maximization clustering in each event cluster. The comparison results are depicted in Figure 7.5.

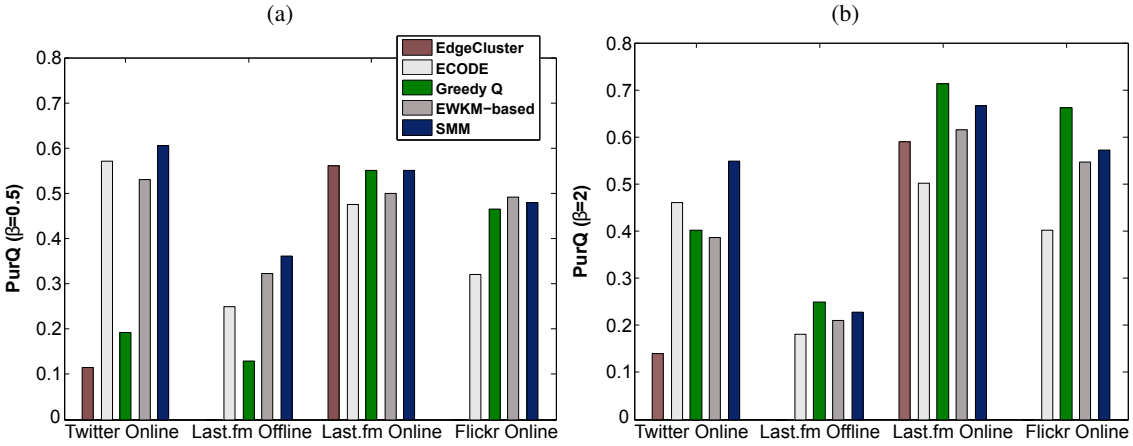

Figure 7.5: The performance comparison with $\beta = 0.5$ and $\beta = 2$ for different datasets

All the evaluated methods have nearly similar performance in Last.fm Online EBSN particularly when $\beta = 0.5$. Indeed, the communities detected within this network have very small sizes (e.g., average size equal to 15 for the Greedy Q) due to the extremely sporadic interactions. This is also explained by the low density link and the user participation behavior where 92% of users are associated at most with only two events. Hence, the link information was sufficient to obtain a good semantic purity. This aspect slightly decreases in Flickr dataset where 78% of users are associated with at most two events. Indeed, the Greedy Q method apparently achieves a good semantic purity. However, the fraction F_{purity} is only equal to 0.6, a fair value compared with the EWKM-based method and the SMM approach where F_{purity} are respectively equal to 0.89 and 0.91. In Last.fm Offline and Twitter EBSNs, the Greedy Q method has a poor performance when $\beta = 0.5$. This can be explained by the high density network compared with other datasets. Moreover, the identified communities were very large. For example, this methods produces a community having 474 members among 2847 users in Last.fm offline EBSN. This indicates that users within this network are densely connected which could explain the low modularity Q values produced by different approaches.

Evaluating the semantic-based methods, we note a better performance for ECODE in Twitter EBSN than in the other datasets. This is due to the addition of virtual links to the event-to-event network based on the semantic similarity between users. However, the user profile in Last.fm is much more semantically diverse than in Lanyrd which leads to ambiguous similar scores. In reality, the user may be interested in many musical concerts having different topics, whereas he has more restrictive "scientific" interests that mostly fit his/her expertise domain. We also observe a poor performance of the Edge co-clustering algorithm in Twitter EBSN because it is sensitive to the number of clusters that needs to be accurately determined. Finally, the SMM approach achieves the best performance both when $\beta = 0.5$ and $\beta = 2$. Note that there is similar behavior between our method and the EWKM-based

method. For instance, the average size of communities in Last.fm Offline EBSN is equal to 0.33 for EWKM-based method, and 0.29 for our approach. However, the EWKM-based method is based on k-means clustering which is sensitive to the initial distribution of centroids, thus producing different results in each run. This problem disappears in our approach thanks to the use of hierarchical clustering. From the computation point of view, we observe that all these methods have nearly the same computational time except of the Edge co-clustering. Finally, low semantic purity values were observed in Last.fm Offline EBSN compared with Twitter EBSN. The reason behind this observation is that the musical concerts in Last.fm are attached to much more semantically diverse tags than the conferences in Lanyrd. In the following, we select the EWKM-based method to further evaluate our SMM approach.

Coductance Comparison

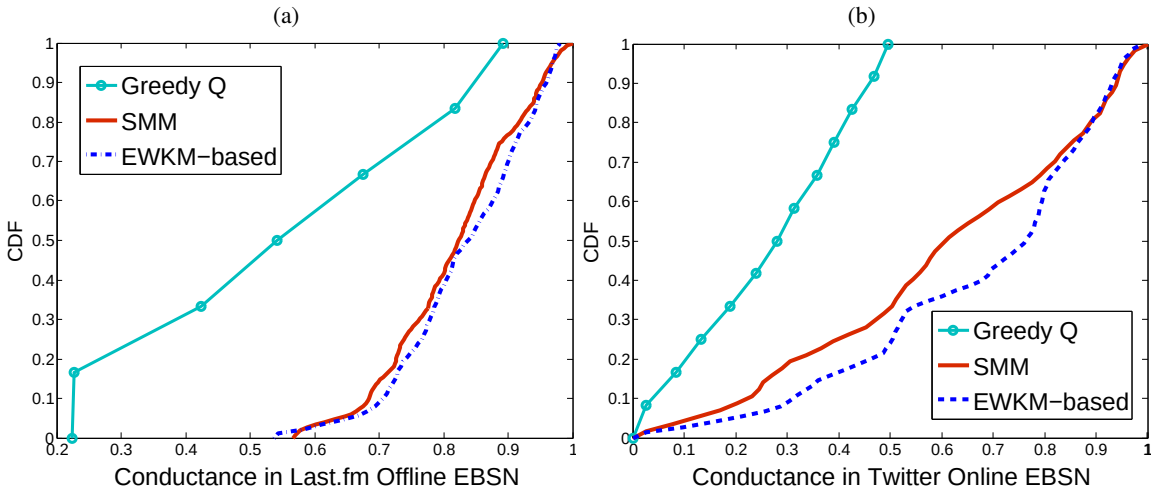

Figure 7.6: Conductance comparison in (a) Last.fm Offline and (b) Twitter Online EBSNs

It is difficult to construct a ground truth that represents the real communities within a network. Hence, we evaluate our community detection approach using the *Conductance* metric [80]. Conductance is a popular quality function assessing whether the detected communities are densely linked but weakly attached to the rest of the network. Note that this metric evaluates the performance from the link-based perspective, where lower conductance means better community partition. Figure 7.6 shows the cumulative distribution (CDF) of the conductance respectively in Twitter EBSN and Last.fm Offline EBSN. It is obvious that the Greedy Q method has high conductance, as it produces very large communities using solely the link information. The SMM approach produces lower conductance values than EWKM-based approach especially in Twitter Online EBSN. We believe that the better performance in Twitter Online EBSN is due to its clustering coefficient which is larger than that

of Last.fm Offline.

User Profile Comparison

To evaluate the performance from the semantic-based perspective, one way is to compare the users' profiles within each community. Hence, we retrieve the users' tags from each website and we only keep the frequent ones, thus creating users profiles represented as vectors of tags. Cosine distance is then applied to compute the similarity between users' profiles within the same communities. We consider that two users are similar when they have a Cosine distance above 0.3, a quite reasonable value considering the noisy tags. Figures 7.7 shows the CDF of the fraction of similar users within the same communities. It can be seen that our SMM approach clustered more "semantically" similar users than the EWKM-based method.

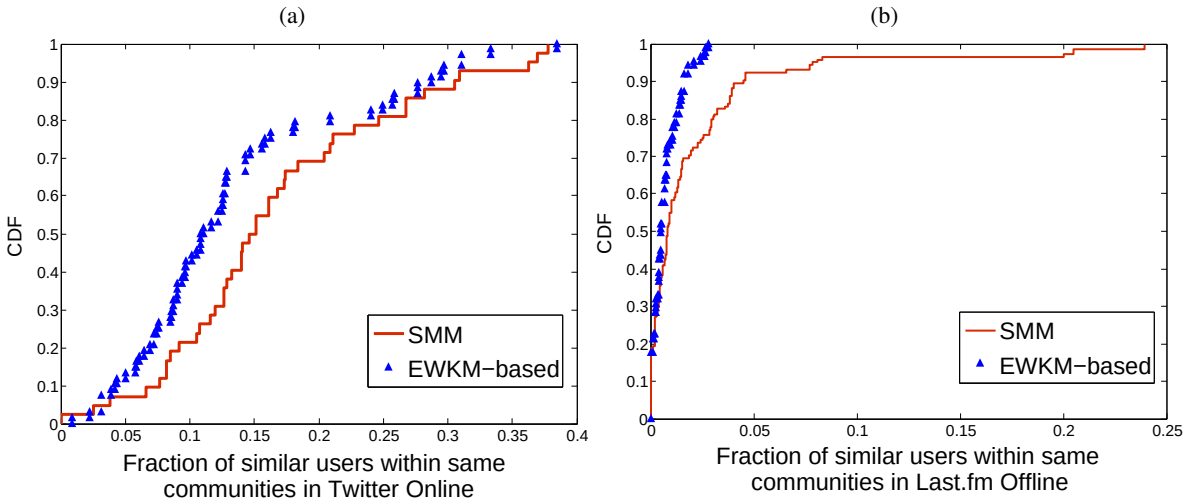

Figure 7.7: Comparison of user profiles in (a) Twitter Online EBSN and (b) Last.fm Offline EBSN

We also examine the fraction of "friends" within each community. The friendship information was extracted using the online social networks that exist in Last.fm and Twitter. Results are shown in Table 7.4. We can see that a large fraction of friends were clustered in the same community by Greedy Q in Last.fm Offline EBSN compared with the other methods. This is also justified by the very high average size of communities detected which is equal to 474.5. Moreover, we can note that conference participants having similar semantic interests are more likely to be friends than the case of musical concert participants.

Method	Twitter Online	Last.fm Offline
Greedy Q	0.72	0.69
EWKM-based	0.70	0.23
SMM	0.73	0.29

Table 7.4: Average fraction of friends within the same communities

Communities Overlap

Lastly, Figure 7.8 shows a tag cloud representing a sample of the most overlapping communities in Twitter EBSN. The link thickness exhibits the overlapping degree. It can be drawn that the main topic of these communities is the Web domain which is the interest of many users who share different kinds of expertise.

Figure 7.8: A sample of some overlapping communities in Twitter Online EBSN

In Twitter EBSN, the SMM approach detects 65 communities while the EWKM-based method produces 92 communities. Analyzing both community structures, it has been found that our approach discovers fewer but more cohesive semantic communities. We evaluate the cohesiveness using the popular Silhouette coefficient [120]. For instance, we have detected only one community about the topic "*user experience*" with a cohesion equal to 0.1. In contrast, 4 communities have been detected about this topic by the EWKM-based method including 2 singletons (i.e., community having one user) and having a cohesion equal to -0.3. This finding underlines the performance of the SMM approach to group together strongly linked users into cohesive communities sharing semantically similar interests.

7.5 Conclusion

In this chapter, we have proposed a new approach to detect overlapping semantic communities in event-based social network. Taking into account both the link and semantic information, we have clustered events by maximizing a newly defined metric called Semantic Modularity. Then, the user membership to each cluster was determined by a link-based function. A comparison with existing studies shows the efficiency of our SMM-based approach to detect meaningful and cohesive communities. The evaluation has highlighted how people interact differently in offline and online EBSN and how these interactions depend on the event category (e.g., conference, concert, etc.). For future work, we plan to combine both the offline and the online worlds to solve community detection in a heterogeneous network.

Conclusion of Part II

This part has been devoted to put in use the Linked Data in event domain. Of a particular interest is the applications that handle better event visualization and discovery, and the personalization techniques.

We have developed Semantic Web applications to support creating and browsing events in a user friendly interface. Overall, consuming Linked Data is advantageous to deliver enriched views of events, and to uncover interesting behavioral facts. Still, it was challenging to use conventional Web technologies on top of RDF data, a fact which reminds the trade-off between simplicity and expressivity of the data model.

We have exploited Semantic Web technologies to build a hybrid recommender system for events. Ontology-enabled feature extraction has been proved to be helpful for reducing the data sparsity in a recommender system. We have highlighted the benefits of Linked Data to improve performance by providing enriched data.

Finally, we have proposed an approach to detect overlapping semantic communities in event-based social network. We have considered both the link and semantic information to form cohesive groups having semantically similar interests. Linking events with media was particularly useful to construct EBSNs from media services. We have showed how people interact differently from one site to another.

CHAPTER 8

Conclusions and Future Perspectives

In this chapter, we summarize the major achievements of this thesis and we give an outlook on future perspectives.

8.1 Achievements

An ever increasing amount of information spread on the Web is centered on the notion of "event". Currently, most of companies that provide calendar of events such as Eventful, Last.fm and Lanyrd are using Web 2.0. They provide an environment where users can view and create an event, and locate events through keyword-based search and ranked results. Such design as unconnected data silos is, however, different from the conception to which aim the "Web of events". There is no support to handle the natural relationships that link events at different levels (e.g., similarity) or to link events with their experiential attributes such as discussions and captured media. Indeed, associating events with background knowledge and media, and linking events together may change the way people or systems exploit data. This thesis thoroughly describes the different steps aiming to realize the vision of the Web of events having as a foundation the Web 2.0 and harnessing the Semantic Web technologies. The work presented put a focus on building the Web of events and on reusing it in Web applications and personalization techniques. The contributions made are:

- **Data Structuring:** A milestone towards the Web of events is to semantically model what an "event" is. Due to its inherently multidimensional nature, we surveyed some different definitions, and we retained the one which represents the factual properties. This definition is based on the *Ws* questions: *What*, *When*, *Where* and *Who*. To formalize the event definition, we opted for the LODE ontology as an interoperable model realized without any particular interpretation or perspective. This complies with our strategy to retrieve and model any type of event. On the other hand, the modeling of media was simply achieved by the reuse of popular ontologies in the domain.

- **Data Aggregation:** Many Web directories contain event-centric data including calendar of events or captured media. Aggregating this data and exploring the explicit overlap between these directories are parts of the building process. Thus, we developed a framework that collects events and media, and exploits explicit metadata (e.g.,

machine tags, hashtag) to link them. We followed one design requirement which is the flexibility. The objective is to be able to flexibly add more event and media directories in the future.

- **Data Reconciliation:** We particularly addressed two different tasks having in common the challenge of data heterogeneity. The former creates identity link between two same real-world instances and the latter aligns events with microposts at the sub-event level of granularity. For the first task, we surveyed existing automatic instance matching tools. Yet, none of them is able to overcome the heterogeneity found between event directories. As a solution, we proposed a domain-independent matching approach taking into account various data types. As for the second task, we proposed a Named Entity-based approach to bridge the gap between the unstructured content of microposts and the structured descriptions of events. The idea is to exploit the mapping between the taxonomy of named entities and the concepts of the event ontology.

- **Application and Analysis:** We developed some Semantic Web applications enabling new mechanisms to browse and discover events or to create events in a controllable way. Our experience highlighted some limitations to use RDF data with conventional Web technologies. We also showed the importance to design a simple data model at the expense of expressivity. Lastly, we explored the benefits of Linked Data to uncover behavioral aspects and to improve the user profiling.

- **Event Recommendation:** We designed a hybrid recommender system in order to suggest personalized events. It is built on top of the Semantic Web and combines content-based recommendation and collaborative filtering. It is shown that ontology-enabled feature extraction and enrichment with Linked Data significantly improve the performance. In addition, a user may be involved in many events, but interested in specific topics. Thus, we proposed a method to alleviate the impact of the topical diversity that may characterize a user profile. Results underlined the importance of the social information and the user interests modeling in event recommendation.

- **Community Detection in EBSN:** We presented an approach to detect overlapping semantic communities in event-based social network relying on structural and semantic features. The links between events and media were used to construct event-based networks from media directories. The approach proposed aims to maximize a novel metric called semantic modularity. The evaluation results prove the performance of our approach, and shed light on the difference between the online and offline networks in terms of users' interactions which can be dense or sporadic.

In summary, these contributions pave the way to build the Web of events as part of Linked Data. The main idea is to bring together event-centric data into a unified structured knowl-

edge with the flexibility and depth afforded by the Semantic Web technologies. As rich data made available in the Linked Data cloud, one can expect efficient supports to browse, search and visualize rich data. The work presented in this thesis goes beyond this fact and further demonstrates the utility of Linked Data in other tasks such as personalization, user modeling and behavioral analysis. Although focused on events, some proposed approaches could be easily propagated to other domains such as movie recommendation or community detection in social networks.

8.2 Perspectives

The work in this thesis specifically targets the event domain to build and leverage a meaningful knowledge base. Still, it could be extended by the following future directions:

- **Enrichment:** One enhancement is to enrich EventMedia dataset with other popular social media websites such as Facebook and Eventbrite [1]. As such, we can increase the overlap in terms of coverage and exploit the assets of each website. Indeed, at the time of writing, the integration of Facebook was under development. Enrichment could also concern data modeling by incorporating useful vocabularies such as the Tickets ontology [48] and the Allen's interval temporal algebra [3].

- **Relationships of Events:** Events sharing spatial-temporal context or having in common a specific topic or participants may have a relationship between them. This reveals a key aspect in the Web of events which is to represent the natural relationships at different levels such as referential, structural and causal. While we only dealt with the identity link, the other links remain unexplored. This opens the door for future work exploring more meaningful connections. Moreover, the existing approaches mostly address a specific domain (e.g., historical [25]) and focus on specific event attributes (e.g., time [57]). There is a need for a formal specification that takes into account all the event attributes and proves its efficiency to be applied in different domains (e.g., social, political).

- **Temporal Dynamics:** Temporal dynamics is an important aspect that recently drives the way to design computing applications. The growth of online activities has led to new challenges about how to handle streaming data, instead of static files, which needs more efficiency and scalability. Moreover, data may shift over time, a fact that may impact many tasks such as reconciliation or recommendation. In instance matching, solving at the same time the high heterogeneity and temporal dynamics is a quite challenging problem. Our strategy put focus on the heterogeneity problem and still based on supervised learning using labeled data. In order to face future changes, it can be

1. http://www.eventbrite.com

sought to automatically generate a ground truth or to fully rely on an unsupervised method. Temporal dynamics has also an impact on event recommendation. Unlike a classic product, an event is ephemeral, and as such, the list of events continues to grow in the user profile and may become unmanageable. To solve this, one simple approach is to discard irrelevant items outside a temporal window [137]. This raises the question about the effective window size and its impact on the system performance.

- **Scalable Recommender System:** To overcome the information overload, we proposed a hybrid recommender system based on the classical Vector Space Model (VSM). Although efficient to provide personalized events, our approach has a serious drawback of scalability since the time complexity is linear to the number of events (i.e., documents in VSM). Considering this limitation, several optimization techniques found in the literature could be integrated to speed up the computation. One technique is to reduce complexity in VSM by pruning unnecessary similarity comparisons. This can be ensured by the high-dimensional similarity search techniques such as the popular indexing method named Locality Sensitive Hashing (LSH) [41]. Another solution worth to be investigated is the multi-relational learning using tensor factorization which can be applied in Linked Data. This is particularly the goal of Rescal-ALS, a scalable tool that represents entities in a latent space enabling efficient information propagation via the dependency structure [106].

- **Community-based Recommendation:** Exploiting community detection for recommendation has been the subject of numerous research studies. It is also an indirect way to assess the quality of the identified communities. Indeed, it has been shown that taking advantage from a collective behavior of users is one solution to alleviate the cold-start problem [123, 13] or to diversify recommendation [39]. In this perspective, our recommender system could be improved by the integration of our community detection approach applied on event-based social network. Another similar direction is to build a signed network from user interactions as has been proposed by Maniu et al. [90], which can be used to build a trust-aware recommender system.

Part III

Appendix

APPENDIX A

List of Publications

A.1 Journals

- Khrouf, Houda; Troncy, Raphaël "De la modélisation sémantique des événements vers l'enrichissement et la recommandation". Revue d'Intelligence Artificielle, Numéro spécial Ingénierie des connaissances, 28(2-3), pp. 321-347, 2014.
- Khrouf, Houda; Milicic, Vuk; Troncy, Raphaël "Mining events connections on the social Web: Real-time instance matching and data analysis in EventMedia". Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 24:3–10, 2014.
- Khrouf, Houda; Troncy, Raphaël "EventMedia: a LOD Dataset of Events Illustrated with Media". Semantic Web Journal, Special Issue on Linked Dataset descriptions, pp. 1570–0844, 2012.

A.2 Conferences and Workshops

- Khrouf, Houda; Troncy, Raphaël "Hybrid event recommendation using linked data and user diversity". In Proceedings of the 7^{th} ACM Conference on Recommender systems, Hong Kong, China, pp. 185-192, 2013. *(acceptance rate: 24%)*
- Buschbeck, Sven; Troncy, Raphaël; Jameson, Anthony; Khrouf, Houda; Spirescu, Adrian; Suominen, Osma; Schneeberger, Tanja; Hyvönen, Eero "Parallel faceted browsing". In Proceedings of the ACM Conference on Human Factors in Computing Systems, Paris, France, pp. 3023-3026, 2013.
- Khrouf, Houda; Milicic, Vuk; Troncy, Rapahël "EventMedia live: Exploring events connections in real-time to enhance content". In Proceedings of the Semantic Web Challenge at the 11^{th} International Semantic Web Conference (ISWC), Boston, USA, 2012. **1^{st} Prize Winner of the Semantic Web Challenge**.
- Buschbeck, Sven; Jameson, Anthony; Troncy, Raphaël; Khrouf, Houda; Suominen, Osma; Spirescu, Adrian "A demonstrator for parallel faceted browsing". In Proceedings of the EKAW 2012 Workshop on Intelligent Exploration of Semantic Data Workshop, Galway, Ireland, 2012. **Winner of the IESD Challenge**.

- Khrouf, Houda; Atemezing, Ghislain; Rizzo, Giuseppe; Troncy, Raphaël; Steiner, Thomas "Aggregating social media for enhancing conference experiences". In Proceedings of the 1^{st} International Workshop on Real-Time Analysis and Mining of Social Streams (RAMSS), Dublin, Ireland, pp. 34-37, 2012.
- Khrouf, Houda; Atemezing, Ghislain; Steiner, Thomas; Rizzo, Giuseppe; Troncy, Raphaël "Confomaton: A conference enhancer with social media from the cloud". In Proceedings of The Semantic Web: ESWC 2012 Satellite Events, Heraklion, Crete, pp. 463-467, 2012.
- Khrouf, Houda; Troncy, Raphaël "EventMedia: visualizing events and associated media". In Demo Session at the 10^{th} International Semantic Web Conference (ISWC), Bonn, Germany, 2011.
- Khrouf, Houda; Troncy, Raphaël "EventMedia Live: reconciliating events descriptions in the Web of data". In Proceedings of the 6^{th} International Workshop on Ontology Matching, 2011, Bonn, Germany, pp. 250-251, 2011.
- Khrouf, Houda; Troncy, Raphaël "Réconcilier les événements dans le Web de données". In Proceedings of the 22^{nd} Journées Francophones d'Ingénierie des Connaissances, Chambéry, France, pp. 726-738, 2011.

A.3 Archived Technical Reports

- Troncy, Raphaël; Khrouf, Houda; Shaw, Ryan; Hardman, Lynda "Specification of an event model for representing personal events". ALIAS Deliverable D4.1, 2011, http://deliverables.aal-europe.eu/call-2/alias/d4-1-specification-of-an-event-model-for-representing-personal-events.
- Troncy, Raphaël; Khrouf, Houda; Atemezing, Ghislain; Fialho, Andrsé; Hardman, Lynda "Module for knowledge enrichment of event descriptions". ALIAS Deliverable D4.3, 2011, http://deliverables.aal-europe.eu/call-2/alias/d4-3-module-for-knowledge-enrichment-of-event-descriptions.
- Khrouf, Houda; Troncy, Raphaël; Milicic, Vuk "Module for personalized discovery of new contacts on line". ALIAS Deliverable D4.4, 2013, http://deliverables.aal-europe.eu/call-2/alias/d4-4-module-for-personalized-discovery-of-news-contacts-on-line.
- Khrouf, Houda; Troncy, Raphaël "Module for retrieval of opinionated content". ALIAS Deliverable D4.5, 2013, http://deliverables.aal-europe.eu/call-2/alias/d4-5-module-for-retrieval-of-opinionated-content.
- Khrouf, Houda; Troncy, Raphaël "Module for topics recommendation". ALIAS Deliverable D4.7, 2013, http://deliverables.aal-europe.eu/call-2/alias/d4-7-module-for-topics-recommendation.

- Scharffe, François; Fan, Zhengjie; Ferrara, Alfio; Khrouf, Houda; Nikolov, Andriy "Methods for automated dataset interlinking". Datalift Deliverable D4.1, 2011, http://hal.inria.fr/hal-00793435.
- Euzenat, Jérôme; Abadie, Nathalie; Bucher, Bénédicte; Fan, Zhengjie; Khrouf, Houda; Luger, Michael; Scharffe, François; Troncy, Raphaël "Dataset interlinking module". Datalift Deliverable D4.2, 2011, http://hal.inria.fr/hal-00793433.

APPENDIX B

Extended Background

In this chapter, we provide an extended background about the basic concepts and different techniques used throughout this thesis.

B.1 String Similarity

There are three main families of string similarity functions, namely token-based functions, character-based functions and hybrid functions. In this appendix, we overview the most popular functions in each family. The following described formulas compare two strings s and t which are associated with two token sets $S = s_1, s_2, ..., s_n$ and $T = t_1, t_2, ..., t_m$, respectively. For the computation, we use the Similarity Metric Library available online [1].

B.1.1 Token-based Functions

The first family of the string similarity is the token-based functions which consider a string as a set of tokens. Intuitively, tokens also called "bag of words" are substrings generated by a tokenization function (e.g., typically by a whitespace) applied on the original string. Making use of token-based functions is advantageous to overcome a change in the ordering or a swap of words. For example, the similarity between *Mahatma Gandhi* and *Gandhi Mahatma* will be maximal as both strings share the same tokens. However, the main drawback of such functions is to penalize approximate tokens having few spelling variations. That is, the comparison score of *brother* and *brothers* is zero.

One popular function is the Jaccard similarity [55] which is the ratio of the intersection size and the union size of two token sets:

$$Jaccard(S,T) = \frac{|S \cap T|}{|S \cup T|}$$

Q-gram is another function which splits a string into small overlapping (i.e., common characters) units of size q. To obtain such units with the first and last characters of a string, we introduce a padding character (e.g., #). For example, the 3-grams of *Gandhi* is the set *(##G,#Ga, Gan, and, ndh, dhi, hi#, i##)*. Then, Jaccard function is typically used based on these tokens to compute the similarity score.

1. http://sourceforge.net/projects/simmetrics

The drawbacks of Jaccard is that it is very sensitive to spelling errors and it significantly penalizes the unmatched tokens. In contrast, q-gram is less sensitive to spelling errors or to unmatched tokens. This comparison is illustrated in Table B.1.

String s	String t	Jaccard	3-gram
Johnny Depp	Johny Dep	0	0.75
sir Johnny Depp	Mr Johnny Depp	0.5	0.78

Table B.1: Comparison between Jaccard and 3-gram functions

Cosine distance is another typical token-based function used in Information Retrieval for high dimensional data. Given two n-dimensional vectors X and Y containing the weights of tokens in S and T, the Cosine distance is defined as the cosine angle between these two vectors:

$$Cosine(X,Y) = \frac{|X \cdot Y|}{\|X\| \cdot \|Y\|} = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}}$$

To generate the weights vectors X and Y, the TF-IDF Cosine is commonly used where each token has a weight according to Term Frequency-Inverse Document Frequency scheme. This scheme is composed of two measures: term frequency (*tf*) and inverse document frequency (*idf*). The intuition behind the term frequency is that the more often a token appears in a given string, the higher is its contribution to the similarity. In contrast, the inverse document frequency assigns higher weights to rare tokens in all the corpus (all the strings or documents). For each token s_i in the token set S, the IF-IDF score is:

$$tf\text{-}idf_{i,s} = tf_{i,s} \cdot log\left(\frac{D}{D(t_i)}\right)$$

where $tf_{i,s}$ is the term frequency of s_i in the string s, D is the number of the strings in the corpus, $D(i)$ is the number of strings that contain the token s_i in the corpus. This weight increases proportionally to the number of times a token appears in the document, but is offset by the frequency of the token in the corpus. The TF-IDF Cosine is useful to compute the similarity between two text documents in which attributes exhibit word frequency.

B.1.2 Character-based Functions

The second family is the character-based functions also called edit-based similarity. Unlike the token-based functions, a string is considered as an ordered sequence of characters instead of a set of tokens. They allow different "edit operations" necessary to transform one string to another such as deletion, insertion, substitution and transposition of characters. The use of these functions is mainly performed on short strings to overcome spelling errors. However, their performance drastically decreases when changing the order of tokens.

One popular function is the Levenshtein distance [81] that allows three edit operations which are the deletion, insertion and substitution. The score is equal to the minimum number of operations required to transform s to t. For example, to transform *Maria* to *Mario*, we need to replace "a" by "o", which gives a similarity score equal to 1. The normalized score is equal to 0.8. One drawback of Levenshtein is that it is not adapted for some variations such as abbreviations (e.g., *Gandhi Mahatma* and *Gandhi M*) or extra prefix (e.g., *Sir Gandhi* and *Gandhi*).

A similar metric is the Jaro distance [58] which allows character transpositions and based on the number and the order of common characters. Two characters are considered to be common if they are equal and if the distance between their positions i and j within the two strings does not exceed H, where $H = 0.5 \times min(|s|,|t|)$. Given a set of common characters σ, a transposition occurs if the i^{th} common character of s is different from the i^{th} common character of t. Let θ is half the number of transpositions, Jaro is computed as:

$$Jaro(s,t) = \frac{1}{3} \times \left(\frac{|\sigma|}{|s|} + \frac{|\sigma|}{|t|} + \frac{|\sigma| - \theta}{|\sigma|} \right)$$

Jaro distance performs well when there is few spelling variations. However, as common characters have to occur in a specific distance, variations such as a long prefix in one string yields a low similarity. For example, The Jaro similarity between $s = Doctor\ John\ Smith$ and $t = John\ Smith$ equals to only 0.46. A variant of Jaro distance, called Jaro-Winkler similarity [141] uses the length of the longest common prefix to emphasize matches in the first p characters of the two strings. For example the Jaro similarity between *John S* and *John Smith* is 0.86, while the Jaro-Winkler score is 0.94.

B.1.3 Hybrid Functions

To overcome the limitations of character and token based functions, the metrics in the third family combines both of them, and referred as hybrid functions.

Extended Jaccard similarity is a hybrid function proposed to include not only the equals tokens, but also the similar ones in the the original Jaccard function [139, 5]. Let *TokenSim* be a string similarity metric that compares two tokens s_i and t_j, and θ is the related threshold, the set of shared similar tokens between s and t is defined as:

$$Shared(s,t) = \{(s_i,t_j) | s_i \in S \wedge t_j \in T : TokenSim(s_i,t_j) \geqslant \theta\}$$

The set of unique or unmatched tokens in s is defined as:

$$Unique(s) = \{s_i | s_i \in S \wedge t_j \in T \wedge (s_i,t_j) \notin Shared\}$$

Similarly, we define the set $Unique(t)$ for the string t. This has been extended by a function that gives weights w to matched and unmatched tokens, which are combined using an

aggregation function Ag. The hybrid Jaccard is defined as:

$$matched = Ag_{(s_i,t_j)\in Shared(s,t)}\ w(s_i,t_j)$$

$$unmatched = Ag_{(s_i)\in Unique(s)}\ w(s_i) + Ag_{(t_j)\in Unique(t)}\ w(t_j)$$

$$HybridJaccard(s,t) = \frac{matched}{matched + unmatched}$$

Note that different weights could be given for the tokens in $Shared(s,t)$, $Unique(s)$, and $Unique(t)$. For instance, let s= *Mindy Smith* and t=*Minndy Smith Festival*, the hybrid Jaccard generates the following sets:

$$Shared(s,t) = \{(Mindy, Minndy), (Smith, Smith)\}$$

$$Unique(s) = \varnothing$$

$$Unique(t) = \{Festival\}$$

Assuming that the weights of matched tokens is their normalized Levenshtein similarity, and the weights of unmatched tokens is equal to 1. If the aggregate function Ag simply sums the weights, the hybrid Jaccard is:

$$HybridJaccard(s,t) = \frac{0.83+1}{0.83+1+0+1} = 0.64$$

Note that the score remains low due to the influence of unmatched tokens. Our Token-wise metric proposed in Section 4.1.2 follows the same rationale, but gives more importance to similar tokens. Moreover, the weight of unmatched tokens takes into account the fact that the two token sets have different sizes. In this example, the weight for unmatched tokens is equal to $\frac{2}{3} = 0.66$. We obtain higher score than hybrid Jaccard when using Token-wise:

$$Token\text{-}Wise(s,t) = \frac{2\times(0.83+1)}{2\times(0.83+1)+0.66\times(0+1)} = 0.84$$

Another hybrid function is the Monge-Elkan similarity [97] that matches every token s_i in T with the token t_j in T having the maximum similarity using *TokenSim* metric. Monge-Elkan is defined as:

$$MongeElkan(s,t) = \frac{1}{|S|}\sum_{i=1}^{|S|}\max_{j=1}^{|T|} TokenSim(s_i,t_j)$$

Given the previous example (s = *Mindy Smith* and t = *Minndy Smith Festival*), and using Levenshtein as *TokenSim*, the Monge-Elkan score is:

$$MongeElkan(s,t) = \frac{0.83+1}{2} = 0.91$$

Monge-Elkan is sensitive to the size of the first string. For instance, if t is the first string which is of length 3, the Monge-Elkan score decreases to 0.61.

The last hybrid function is called SoftTFIDF [24] which extends the Cosine similarity, following the same rationale as hybrid Jaccard. Let $CLOSE(\theta, S, T)$ be the set of words $s_i \in S$ such that there is $t_j \in T$ where $TokenSim(s_i, t_j) > \theta$, and $maxsim(s_i, t_j) = max(\{TokenSim(s_i, t_j) | t_j \in T\})$. The SoftTFIDF is defined as:

$$SoftTFIDF(s,t) = \sum_{s_i \in CLOSE(\theta,S,T)} \left(\frac{tf\text{-}idf_{s_i}}{\|X\|} \cdot \frac{tf\text{-}idf_{t_j}}{\|Y\|} \times maxsim(s_i, t_j) \right)$$

where X and Y are the vector representations of s and t containing the $tf\text{-}idf$ scores of related tokens, respectively. Given the previous example ($s =$ *Mindy Smith* and $t =$*Minndy Smith Festival*), and unit weights for all the tokens (no corpus considered), the SoftTFIDF gives:

$$SoftTFIDF(s,t) = \frac{1}{\sqrt{2}} \times \frac{1}{\sqrt{3}} \times 0.83 + \frac{1}{\sqrt{2}} \times \frac{1}{\sqrt{3}} \times 1 = 0.75$$

B.2 Optimization Techniques

In this appendix, we also overview some technical aspects used in this thesis. More precisely, we describe two artificial intelligence techniques namely the Genetic Algorithms and the Particle Swarm Optimization widely used in optimization problems.

B.2.1 Genetic Algorithm (GA)

Genetic Algorithm is a stochastic method inspired by the mechanism of natural evolution and genetic inheritance [143]. GA is one of the most popular evolutionary algorithms widely used for solving optimization problems in many areas such as machine learning and image processing. The idea behind is that the best solution can be found by combining the "good" parts of other solutions. In GA, a population is a set of *chromosomes* (candidate solutions) and each chromosome denotes a set of *genes*. The content of each gene is called "allele". A key component in GA is the setting of the fitness criterion which accurately evaluates the quality of candidate solutions. First, a population of chromosomes are randomly generated and evaluated using the fitness function. The chromosomes having higher fitness values than others are stochastically selected, recombined and mutated to produce a new population for the next generation. To achieve this, GA has a set of key operators, namely selection, crossover and mutation. The selection operator is used to select chromosomes called parents to create the descendants of the next generation. The selection usually favored fitter parents, and there exist some selection techniques in the literature. One technique is the *Stochastic Universal Sampling (SUS)* developed by Baker [6] and used in this thesis. Given a line where each chromosome occupies a segment proportional to the chromosome's fitness, *SUS* uses N equally spaced pointers placed over the line where N is the number of selections required.

Once parents for new population are chosen, genetic operators are applied such as crossover and mutation. Crossover refers to the recombination of parents to form a child. In particular, we used the scattered crossover which creates a random binary mask, then selecting the genes of parents based on this mask. In order to force the algorithm exploring new areas in the search space, mutation is performed which alters at least one gene in a chromosome according to a predefined probability. Mutation rarely occurs in nature, which can justify the typical value 0.01 generally used as a mutation probability. Finally, the algorithm stops iterating when the optimal solution is produced or a maximal number of iterations is reached.

B.2.2 Particle Swarm Optimization (PSO)

PSO is a population-based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling. It is similar to evolutionary algorithms and it was introduced in 1995 by Kennedy and Eberhat [62]. Compared with GA, PSO is easy to implement with few parameters to adjust, and each individual benefits from its history whereas no such a mechanism exists in GA. PSO has been successfully applied to solving a wide range of optimization problems in different fields such as robotics, image, neural network and information retrieval. It simulates a group of birds searching for food in a bounded area, where the best position is the one containing the highest density of food. At the beginning, all the birds start searching for food randomly. Each bird knows two positions: its own position (i.e., history) found with the most of food and the best position from the whole swarm. The birds will be guided by these two positions in the search process until optimal convergence.

Technically, the PSO algorithm initializes a population of random solutions called particles, and searches for the optimal solution of a fitness function by updating generations. In each generation, each particle accelerates in the direction of its own personal best solution found so far, as well as in the direction of the global best position discovered so far by any of the particles in the swarm. This means that if a particle discovers a promising new solution, all the other particles will move closer to it, exploring the region more thoroughly in the search process. Each particle i in the swarm has the following attributes: a current position x_i, a current velocity v_i, and a personal best position p_i in the search space, and the global best position p_{gbest} among all the p_i. In each iteration, the velocity and the position of each particle is updated as follows:

$$v_i(t+1) = w.v_i(t) + c_1 r_1 (p_i - x_i(t)) + c_2 r_2 (p_{gbest} - x_i(t))$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where c_1 is the acceleration coefficient for each particle to move to its personal best position, c_2 is the acceleration coefficient to move to the global best position, r_1 and r_2 are random

numbers uniformly distributed within [0,1], and w is the inertia weight which controls the contribution of the particle's previous velocity to its current velocity. The velocity and acceleration are responsible for changing the position of the particle to explore the space of all possible solutions, instead of using existing solutions to reproduce. The personal and the global best positions are the optima of a predefined fitness function, respectively in each iteration and for all past iterations. In this thesis, to adjust some PSO parameters, we followed the setting recommended by Eberhart and Shi [36].

B.3 Recommender Systems

Broadly speaking, the recommender systems are based on two popular strategies: the content-based filtering and the collaborative filtering. In the following, we overview the basic concepts about those techniques.

B.3.1 Content-based Recommendation

The content-based recommendation exploit the attributes characterizing an item or a user. They analyze the content information collected explicitly or implicitly to construct a user or an item profile. The matching between both profiles can be quantified using a variety of similarity distances such as Cosine similarity, Pearson correlation and Latent Semantic Analysis [30]. This kind of matching is also applied to discover people sharing similar interests. It is closely related to detecting documents of similar content in information retrieval field. A known successful realization of content-based filtering is the Music Genome Project which is used for the Internet radio service Pandora.com. In this project, a trained music system ranks each song based on hundreds of distinct musical characteristics. These attributes or genes capture not only a song's musical identity but also many significant qualities which are relevant to understanding listeners' musical preferences [73]. Another interesting study proposed by Chen at al. [22] compare different recommender algorithms in the IBM's enterprise social networking service called "Beehive". The authors underline that the pure content matching is the most effective to recommend unknown friends and diverse items. However, the content-based recommendation has the drawback to not take into account the information in preference similarity across individuals.

B.3.2 Collaborative Filtering Recommendation

The second recommendation strategy is based on collaborative filtering(CF), a technique that does not need an explicit content profiling and purely rely on past user behavior [49]. It has been widely applied in many well-known services such as Amazon, Faceboook, LinkedIn, MySpace and Last.fm. The basis is to analyze the relationships between users and inter-dependencies among items to identify new user-item associations. In other

words, the system makes automatic predictions (filtering) about the user interests based on the preferences of like-minded and similar users (collaborating). The intuition behind is that if a person A has the same preference as a person B on a specific item, A is more likely to have B's preference on another item, as the example illustrated in Figure B.1.

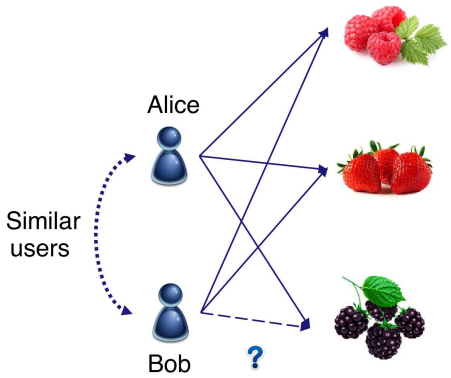

Figure B.1: User-based collaborative filtering: Alice has a crush on berry fruits, Bob also likes two of them. The recommender system understands that Alice and Bob have similar tastes, and Bob is recommended the Blackberry

There exists two primary categories of collaborative filtering which are memory-based and model-based approaches. The memory-based systems compute the similarity between users or, alternatively, between items based on users preferences, thus detecting the neighbors of a given user or item. Indeed, the unknown rating value of the active user u for an item m is an aggregation of the ratings of users similar to u for the same item m, or an aggregation of the ratings of the user u to similar items of m. The model-based systems, on the other hand, use data mining and machine learning algorithms to estimate or learn a model from observed ratings to make predictions. A typical example is the latent factor model that discovers unobserved factors from ratings patterns. The underlying assumption is that there is a set of common hidden factors which explain a set of observations in co-occurrence data. More precisely, the similarity between users and items is simultaneously induced by some hidden lower-dimensional structure in the data. Recently, several matrix factorization methods [73] have been proposed as a successful realization of latent factor model. The users and items are simultaneously represented as unknown feature vectors within a user-item matrix. These feature vectors are learnt using low-rank approximations, so that they approximate the known preference ratings with respect to some loss measure. Despite the important success of collaborative filtering, it still suffers from three serious limitations: the sparsity problem where there are few ratings about items, the cold-start problem where items have no ratings, and the scalability problem where a large amount of users and items have to be analyzed.

ANNEXE C

Résumé en Français

C.1 Introduction

Récemment, de nouveaux services en ligne ont permis aux utilisateurs de facilement publier, filtrer et organiser de vastes répertoires de références dans leurs domaines d'intérêt. Le Web social a connu, par exemple, la création de plusieurs référentiels d'événements et de média favorisant l'accès rapide à l'information et la socialisation virtuelle. Cette évolution a changé la manière dont les gens organisent et communiquent autour d'événements en utilisant de plus en plus les dispositifs numériques. Par conséquent, des milliers d'événements sont publiés régulièrement sur internet sous forme de calendriers électroniques et sont illustrés par des contenus multimédias. Pour mieux comprendre ces nouvelles tendances, une étude exploratoire auprès d'un échantillon de participants a été réalisée. Le but est de découvrir comment les gens utilisent les référentiels d'événements, les réseaux sociaux et les sites de partage multimédia pour chercher et partager des données événementielles [40]. Les résultats démontrent que l'information disponible est souvent incomplète, erronée et enfermée dans une multitude de sites Web. Les participants ont reconnu leur besoin d'accéder à plusieurs sources pour collecter toutes les informations disponibles sur un événement et ainsi construire une vue complète. Ils préconisent la nécessité d'une source unique pour explorer le contenu, non pas en créant une nouvelle source de données, mais en centralisant les données existantes pour assurer une couverture plus large. En effet, l'usage de plusieurs moyens de partage est devenu si important que les informations pertinentes sont noyées dans une gigantesque masse de données. Comment organiser et gérer cette énorme quantité d'informations et comment améliorer la qualité de données sont des défis majeurs dans plusieurs domaines de recherche portant sur la technologie de l'information. En particulier, le domaine du Web sémantique a connu l'émergence de nouveaux concepts permettant de donner du sens aux données et les rendre exploitables par des machines. Ainsi, un nouveau paradigme est apparu, connu sous le nom du "Web de données" [1] (*Linked Open Data en anglais*). L'idée est d'étendre le Web des documents afin de créer un réseau de données structurées, connectées, publiées en ligne et facilement réutilisables. En effet, le Web de données a comme double objectif *i)* de publier des objets représentés en RDF et identifiés par des URIs ; *ii)* et d'interconnecter ces objets entre eux.

A l'ère du déluge informationnel, les chercheurs s'investissent de plus en plus dans le

1. http://linkeddata.org/

mouvement du Web sémantique. Ils étudient les technologies et les standards qui permettent de collecter, structurer et interconnecter les données à grande échelle. Leur efforts ont contribué à l'évolution rapide du Web de données qui a pu atteindre une masse considérable de données disponibles, reliées et librement exploitables. En septembre 2010, il était composé de plus de 25 milliards de triplets RDF couvrant divers domaines tels que les données encyclopédiques, médiatiques, gouvernementales, géographiques et statistiques. Cependant, la présence des données événementielles reste très limitée. Ainsi, l'introduction d'une nouvelle source sémantique d'événements provenant des médias sociaux a l'avantage de fournir du contenu structuré facilement exploitable par les applications multimédia. Cela nécessite l'intégration à large échelle de plusieurs sources de données hétérogènes, unifiant ainsi l'information dans un environnement homogène. Comme les référentiels d'événements et de média sont en constante évolution, il est important de construire une architecture qui soit suffisamment flexible afin de pouvoir ajouter et interconnecter facilement de nouvelles sources. 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. Nous exploitons ces technologies tout au long de la thèse pour répondre aux problèmes majeurs d'intégration de données et de peronnalisation du contenu.

C.2 Contexte de la thèse

Dans ce travail, nous employons des techniques d'intégration des données afin de surmonter la diversité des sources événementielles distribuées. En effet, les données provenant de sources multiples sont souvent hétérogènes, que ce soit au niveau syntaxique ou sémantique. Le même objet peut être représenté, nommé ou stocké différemment d'une source à une autre. L'hétérogénéité syntaxique concerne les formats utilisés pour stocker les données tels que XML, relationnel, objet, etc. Dans le Web sémantique, la résolution de ce problème est assurée à travers l'utilisation de RDF qui offre un langage commun pour décrire formellement des ressources selon un modèle de graphe. En revanche, l'hétérogénéité sémantique demeure quant à elle la plus difficile à résoudre. Elle se produit lorsqu'il existe des conflits de représentation qui peuvent survenir au niveau des schémas et des données. Dans cette thèse, nous nous focalisons sur le problème d'hétérogénéité de données qui apparaît lorsque les informations sont incomplètes où certaines propriétés ne sont pas renseignées, lorsque les données contiennent des erreurs et lorsqu'elles sont décrites différemment. Par exemple, le prénom d'une personne peut être décrit en entier ou en abrégé. Ceci est un problème majeur pour l'intégration de données où il est important de décider si deux descriptions provenant de sources différentes réfèrent ou non à la même entité du monde réel (par exemple, un même événement ou un même lieu). Il peut surgir dans plusieurs applications comme lors du nettoyage d'une base de données contenant des redondances ou lors de la fusion de plusieurs

sources de données. Il s'agit donc de définir une méthode qui permet de réconcilier les instances décrits relativement à travers le même schéma en utilisant des mesures de similarité sémantique pour déterminer les instances qui réfèrent à la même entité du monde réel. En particulier, nous étudions la réconciliation des données événementielles provenant du Web dans le but de construire une plateforme homogène qui facilite la navigation et l'exploration d'événements. Nous étudions également l'enrichissement sémantique d'événements structurés avec des médias qui sont souvent non structurés par leur nature.

L'intégration de plusieurs sources de données événementielles permet de donner l'impression à l'utilisateur de naviguer dans un système homogène. Cependant, des milliers d'événements sont partagés chaque jour générant ainsi une abondance importante d'information et dispersant l'attention d'utilisateurs. Pour mitiger ce problème, il convient donc de comprendre le comportement d'utilisateurs afin d'optimiser l'information perçue en intégrant des techniques avancées comme la recommandation ou le ciblage précis d'un public donné. Dans ce travail, nous cherchons tout d'abord à construire un système de recommandation d'événements capable de surmonter la nature transitoire des données événementielles et prendre en compte la dimension sociale. Dans un tel système, les utilisateurs sont associés à un nombre très limité d'objets rendant difficile la compréhension de leurs préférences. Cette limitation peut être compensée par l'analyse des interactions sociales entre les usagers [26]. Outre la recommandation, il existe une autre stratégie novatrice de personnalisation axée sur la collectivité. Cette stratégie suppose que la compréhension d'un utilisateur ne peut se limiter à lui seul, mais doit inclure également son environnement [109]. Le système a donc besoin de détecter les groupes d'individus qui partagent les mêmes intérêts, appelés communautés. La détection de communautés peut servir dans plusieurs applications telles que le ciblage publicitaire, la recommandation ou l'analyse de l'influence sociale. Dans la recherche, la majorité des approches proposées étudient uniquement les relations entre les individus menant à des communautés parfois incompréhensibles, par exemple une communauté où les individus sont fortement liés mais ne partagent pas les mêmes intérêts. Par conséquent, il est important de prendre en compte les préférences d'utilisateurs en analysant le contexte sémantique d'événements auxquels ils ont participés. L'objectif est de détecter des communautés partitionnées, recouvrantes et qui portent sur une thématique compréhensible.

Cette thèse est structurée autour de contributions dans le domaine d'application de la gestion des événements. Dans la section C.3, nous décrivons notre modélisation sémantique d'événements, ainsi que la construction d'un jeu de données appelé EventMedia fournissant des descriptions d'événements et de médias les illustrant. Dans la section C.4, nous proposons une approche pour réconcilier les instances événementielles dans le but d'améliorer la qualité et la complétude des données. Puis, dans la section C.5, nous enrichissons les événements par des micro-messages tout en cherchant à combler le fossé entre les données structurées et non structurées. Nous présentons ensuite deux mécanismes de personnalisation pour pouvoir décoder les intérêts d'utilisateurs. En particulier, nous mettons en valeur, dans

la section C.6, l'intérêt du Web sémantique lors de la conception d'un système de recommandation. Puis, nous proposons dans la section C.7 une approche qui permet la détection de communautés sémantiques et recouvrantes. Enfin, la section C.8 présente la conclusion de la thèse ainsi que quelques directives pour les travaux futurs.

C.3 Collecte et sémantisation des données événmentielles

Dans le contexte actuel, nous opérons dans un environnement marqué par la production croissante de données événementielles et l'émergence continue de nouveaux médias sociaux. Pour faire face à cette évolution, un système de collecte de données devra garantir une intégration flexible permettant l'introduction de nouvelles sources de données avec le moins d'efforts possibles. Dans cette section, nous présentons notre système de collecte de données, ainsi que la modélisation sémantique basée sur des ontologies existantes. Notre collecte a mené à la construction d'un jeu de données appelé EventMedia qui a fait son apparition dans le Web de données.

C.3.1 Collecte et agrégation des données

Les services Web sont de plus en plus présents de nos jours permettant aux développeurs d'accéder aux données, principalement à travers des requêtes REST. Ils permettent à des organisations de définir et publier un ensemble de fonctions logicielles formant ainsi des interfaces de programmation (APIs). Afin de collecter les données provenant de plusieurs sites, il est donc nécessaire d'examiner les APIs associés, et gérer la différence entre ces APIs en termes de politique d'utilisation, des méthodes REST et des schémas de réponse. Dans le but de pallier cette hétérogénéité, il convient de concevoir une API pivot qui combine les différentes APIs en exploitant leurs points communs. Nous avons donc défini une nouvelle API pivot que nous utilisons dans notre système de collecte de données décrit dans la Figure C.1

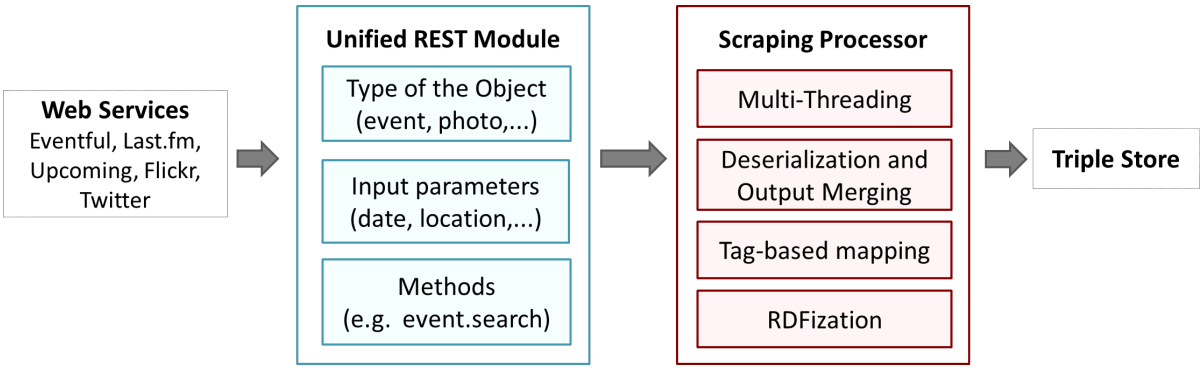

FIGURE C.1 – Le système proposé pour la collecte des données

Notre système de collecte de données est composé de deux modules. Le premier module définit une nouvelle API pivot qui contient les méthodes “sémantiquement” communes entre les APIs sources. Par exemple, la méthode “event.search” prend comme paramètres d’entrée les noms de sources cibles et des opérateurs de filtrage. L’appel de cette méthode déclenche l’appel des méthodes correspondantes dans les sources cibles. La correspondance entre cette méthode et les méthodes sources sont décrites dans des fichiers de configuration. En effet, chaque API source est associé à un fichier descripteur sérialisé en JSON. Ce fichier contient les paramètres globaux comme la clé et l’URL racine, ainsi qu’une liste des objets requêtes. Chaque objet requête concerne un objet particulier (p. ex. événement, lieu, personne) et représente la correspondance entre les paramètres de la méthode source et ceux de la méthode pivot. Un exemple d’un objet requête est décrit dans le listing C.1. Cette stratégie est destinée à faciliter l’introduction d’une nouvelle API source à travers la configuration de son fichier descripteur.

```
"Query": [
          {
            "Type": "search.events",
            "Method": "{0}geo.getevents&api_key={1}",
            "Inputs": [
                {
                    "Name": "Location",
                    "Format": "&location={0}",
                    "Required" : "true"
                },
                {
                    "Name": "LocationRadius",
                    "Format": "&lat={0}&long={1}&distance={2}",
                    "Required" : "true"
                },
                {
                    "Name": "PageNumber",
                    "Format": "&page={0}"
                }

            ]
          }
  ]
```

Listing C.1 – Exemple d’un objet requête pour la collecte d’événements

Le deuxième module assure quant à lui la gestion des requêtes REST et le traitement des réponses reçues. Il utilise le multithreading pour exécuter plusieurs requêtes en parallèle tout en respectant les quota des APIs sources. Les réponses reçues de différentes sources sont exportées dans un schéma commun décrivant un ensembles d’objets, à savoir l’événement, le lieu, les participants et les médias. La description de ces objets sont convertis en RDF en utilisant des ontologies existantes comme décrit dans la section suivante.

C.3.2 Modélisation sémantique

A l'issue des travaux collaboratifs, certains vocabulaires sont devenus très populaires facilitant l'interconnexion de données. Ainsi, le vocabulaire `Dublin Core` [2] est utilisé pour attacher un titre ou une description à une ressource, le vocabulaire `FOAF` [3] pour décrire une personne ou un groupe, et le vocabulaire `WGS84` [4] pour représenter les coordonnées des lieux géographiques. Quant à la notion d'événement, une multitude d'ontologies ont été développées, mais dans des contextes et buts différents. En effet, 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. L'ontologie CIDOC-CRM a été proposée, par exemple, afin de décrire des contenus multimédias relatifs au patrimoine culturel. Elle vise donc les événements historiques au sens large (par exemple guerre, naissance) ou liés aux objets décrits (par exemple établissement d'une bibliothèque). Une étude comparative entre les différents modèles d'événements a été menée par Shaw et al. [129] en mettant l'accent sur l'expressivité et le choix de modélisation. Ce travail propose une ontologie appelé LODE qui tire les meilleurs parties des différents modèles existants. L'idée est de représenter une réalité consensuelle qui n'est pas associée à une perspective ou une interprétation particulière. De ce fait, l'ontologie LODE permet la description interopérable des aspects factuels d'un événement représentés en terme des quatre "Ws" comme suivant :

1. *What* : qu'est-ce qui s'est passé.
2. *Where* : où l'événement s'est-il passé.
3. *When* : quand l'événement s'est-il passé.
4. *Who* : qui etait impliqué.

La Figure C.2 illustre comment l'événement identifié par *ID = 3163952* sur Last.fm est décrit avec l'ontologie LODE. Plus précisément, elle indique qu'un événement de type `Concert` a eu lieu le `21 mai 2012 à 12h45` au `Cinéma Paramount`, avec le groupe de rock `Snow Patrol` et l'un des participants est appelé `earthcapricor`. Cet événement existe aussi dans un autre site Web appelé Upcoming sous l'identifiant *ID = 3163952*. On crée donc une relation d'identité *owl :sameAs* entre les deux événements.

En résumant le modèle global de données, la description d'un événement comprend la catégorie, un texte descriptif, la date sous forme d'un instant ou d'un intervalle de temps, le lieu, les artistes et les participants impliqués. La description du lieu est détaillée à travers les différents champs d'une adresse (par exemple rue, ville, code postal, pays). La description d'un artiste comprend quant à elle le nom, la biographie, des tags descriptifs et une photo. Enfin, chaque participant est attaché à un pseudo, un nom et un avatar

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

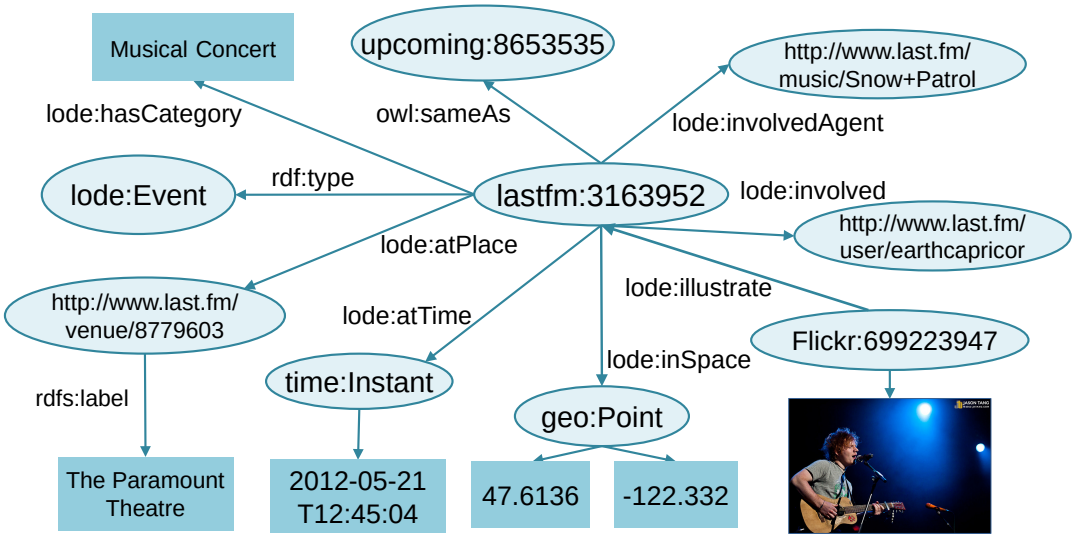

FIGURE C.2 – L'événement *concert de Snow Patrol* décrit selon l'ontologie LODE

C.3.3 EventMedia : un jeu de données événementiel

La collecte et la sémantisation de données a mené à la construction d'un jeu de données événementiel appelé "EventMedia". Il se compose de descriptions d'événements publiés sur quatre annuaires populaires sur le Web, à savoir :

1. Last.fm : c'est un vaste annuaire de musique fournissant des milliers d'événements musicaux par jour et ayant plus de 30 millions d'utilisateurs actifs.
2. Eventful : il héberge le plus grand annuaire d'événements couvrant différents domaines tels que le sport, l'éducation, les loisirs, etc.
3. Upcoming : C'est un autre annuaire d'événements divers, mais il n'est plus en ligne depuis 2013. Nous l'avons utilisé au début de la thèse.
4. Lanyrd : c'est un référentiel assez vaste de conférences, d'ateliers et d'autres événements professionnels.

Dans EventMedia, 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 [5] 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/`). Les événements sont associés à des médias extraits de trois annuaires publics, à savoir Flickr, Youtube

5. `http://fr.wikipedia.org/wiki/Tri_par_cartes`

et Twitter. Pour représenter les médias, nous exploitons les ontologies de référence existantes. Ainsi, l'ontologie du W3C pour les ressources médias [79] a été utilisée pour décrire les photos et les vidéos extraits respectivement de Flickr et Youtube. De même, on a utilisé l'ontologie SIOC [12] pour décrire les micro-messages extraits de Twitter. Les relations entre les médias et les événements sont représentées par une propriété de l'ontologie LODE où son label est `lode:illustrate`. Pour créer ces relations, nous avons exploré le recouvrement explicite en termes des méta-données entre les sites Web déjà mentionnés. Plus précisément, un recouvrement a été découvert entre :

1. (Last.fm, Upcoming) et Flickr : plusieurs photos de Flickr sont attachées à une balise spéciale de type machine-tag telle que `lastfm:event=ID` ou `upcoming:event=ID` où l'`ID` est l'identifiant d'un événement sur Last.fm ou Upcoming. Nous utilisons ces tags lors de collecte de données pour extraire les photos pertinentes.
2. Last.fm et YouTube : Il existe quelques vidéos de YouTube dont la description contient l'URL d'un événement Last.fm. Nous utilisons le mot clé "lastfm event" pour extraire les vidéos pertinentes et les associer aux événements correspondants.
3. Lanyrd et Twitter : Lanyrd (annuaire de conférences) fournissent des hashtags que les participants utilisent lors de partage des micro-messages dans Twitter. Ces hashtags servent donc à extraire les micro-messages qui sont ensuite reliés aux conférences correspondants.

EventMedia est aujourd'hui l'une des bulles de Web de données [6] et il contient plus de 30 millions de triplets RDF. Le tableau C.1 présente un aperçu sur le nombre total des ressources par type et par source.

		Event	**Agent**	**Location**	**Media**	**User**
Annuaires d'événements	Last.fm	69,185	81,006	18,653	7,795	213,351
	Upcoming	29,418	78	14,372	29	23,977
	Eventful	84,225	11,226	30,572	15,532	547
	Lanyrd	2,151	-	624	-	-
Annuaires de médias	Flickr	-	-	-	1,879,343	25,219
	Youtube	-	-	-	517	-
	Twitter	-	-	-	1,060,879	267,138

TABLE C.1 – Nombre de ressources par type et par source dans EventMedia

Nous créons nos propres URIs dans notre espace de noms pour représenter les :

1. événements (`http://data.linkedevents.org/event`)
2. agents (p. ex. des artistes) (`http://data.linkedevents.org/agent`)
3. lieux (`http://data.linkedevents.org/location`)

6. Voir la description d'EventMedia sur CKAN `http://datahub.io/dataset/event-media`

4. participants (http://data.linkedevents.org/user)
5. médias (http://data.linkedevents.org/media)

Tous les URIs d'EventMedia sont déréférençables sur le Web et accessibles autant que des fichiers RDF statiques sérialisés dans différents formats tels que RDF/XML, N3 et N-Triples. EventMedia est muni aussi d'un service en ligne [7] pour exécuter des requêtes SPARQL, et d'une API REST [8] configuré à l'aide de l'implémentation ELDA de Linked Data API [9]. En effet, ELDA permet d'accéder aux données RDF en utilisant des requêtes REST qui sont traduites en des requêtes SPARQL.

C.4 Interconnexion de données événementielles

Dans une étude exploratoire [40], l'enrichissement sémantique d'événements a été perçu comme un moyen de répondre au problème relatif à la qualité et la complétude des données. En effet, un recouvrement important existe entre les ressources provenant des référentiels d'événements, mais aussi de Web de données. Par exemple, la description des artistes liée à un événement dans le référentiel Upcoming est souvent inexistante, ce qui peut être compensé en exploitant les informations relatives au même événement dans le référentiel Last.fm. Pour ce faire, nous avons donc cherché à interconnecter à large échelle les données événementielles en proposant un système de réconciliation adapté. Notre système a pour objectif d'aligner en temps réel les flux de données entrants afin de soutenir un enrichissement en continu. Le gain majeur est de rassembler les avantages de chaque référentiel afin de fournir une vue unifiée et complète sur un événement.

La réconciliation des instances est une tâche d'une importance capitale dans le Web sémantique. Il vise à créer une relation d'identité `owl:sameAs` entre deux instances. Notre objectif est de réconcilier les événements, les personnes et les lieux provenant de jeux de données distribués qui peuvent être représentés par différents schémas. Afin de pallier à l'hétérogénéité sémantique, il y a un besoin d'une approche de réconciliation indépendante du domaine des instances et de schéma utilisé. Par exemple, nous avons remarqué que certaines propriétés, qui sont sémantiquement différentes, peuvent avoir une relation "latente" entre eux. Par exemple, il existe un événement dans le référentiel Last.fm ayant comme titre *"Cale Parks at Pehrspace"*, tandis que le titre du même événement dans le référentiel Upcoming est *"Cale Parks, The Flying Tourbillon Orchestra, One Trick Pony, Meredith Meyer"* qui énumère les artistes impliqués. Ces artistes sont plutôt représentés par la propriété `lode:involvedAgent` dans le référentiel Last.fm. Cependant, ce type d'hétérogénéité a été rarement pris en compte dans les outils de réconciliation existants qui se basent principalement sur une configuration manuelle des propriétés à comparer ou sur une com-

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

paraison automatique des propriétés ayant une sémantique similaire telles que `dc:title` et `rdfs:label`.

C.4.1 Approche de réconciliation

Dans ce travail, nous considérons les différents types de données et nous proposons une technique supervisée de réconciliation basée sur la corrélation et la couverture des propriétés. Notre approche comprend deux étapes : (1) elle détecte les propriétés clés pour sélectionner les candidats ; (2) ensuite, elle 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). Dans ce qui suit, nous présentons la mesure de similarité utilisée selon le type de données tels qu'une chaîne de caractères, numérique ou temporelle.

1. **Chaîne de caractères :** Pour de longues chaînes de caractères (p. ex. description), nous utilisons l'algorithme Porter pour appliquer la désuffixation (*stemming* en anglais), ainsi que la métrique *Cosine* pour calculer la similarité. Pour de courtes chaînes (p. ex. étiquettes), nous proposons une métrique hybride appelée *Token-wise* comme suivant :
$$Token\text{-}Wise(S,T) = \frac{\sum_{s\in S, t\in T} Levenshtein(s,t)}{max(|S|,|T|)} \tag{C.1}$$
où S et T sont les ensembles des jetons (p. ex. mots) formant les chaînes à comparer.

2. **Donnée temporelle :** Nous proposons une métrique qui permet de mesurer la distance entre deux instances, l'inclusion entre un instant et un intervalle de temps, et le recouvrement entre deux intervalles de temps. On considère deux événements (e_1, e_2) qui ont respectivement les couples (d_1, d_1') et (d_2, d_2') (où d la date de début et d' la date de fin d'un événement). La métrique est représentée par la formule suivante :
$$Tmp\text{-}Inc\ (e_1,e_2) = \begin{cases} 1 & \text{if } |d_1 - d_2| \leq \theta \text{ where } (d_1', d_2') = 0 \\ 1 & \text{if } d_1 \pm \theta \in [d_2, d_2'] \text{ where } d_1' = 0 \text{ (idem for } d_2) \\ 1 & \text{if } min(d_1', d_2') - max(d_1, d_2) \geq 0 \text{ where } (d_1', d_2') \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{C.2}$$

3. **Donnée numérique :** On calcule simplement l'inverse de la valeur absolue de la différence entre deux valeurs numériques.

Pour faire face à l'hétérogénéité, il est nécessaire de déduire tout d'abord quelles sont les propriétés à comparer lors du calcul de similarité entre deux instances. Pour ce faire, nous mesurons la corrélation entre les propriétés, ainsi que leur couverture en se basant sur des données de vérification. La corrélation reflète l'information mutuelle en terms de valeurs partagées entre deux propriétés provenant de deux jeux de données source et cible. La cou-

verture reflète le nombre de fois qu'une propriété a été utilisée par toutes les instances. Nous prenons comme exemple deux ensembles d'instances appariées I_s (source) et I_t (target). Pour chaque ensemble I_i ($i \in \{s,t\}$), nous formons l'ensemble des littéraux L_i liés aux instances I_i par le biais des propriétés p_i. Nous associons, à chaque type de données dans L_i, une fonction de similarité $sim_{datatype}$ comme décrit ci-dessus. Le taux de corrélation et de couverture entre deux propriétés peuvent être formalisé comme suivant :

$$Corr(p_s, p_t) = \frac{\sum_{l_s \in L_s, l_t \in L_t} sim_{datatype}(l_s, l_t)}{min(|L_s|, |L_t|)} \tag{C.3}$$

$$Cov(p_s, p_t) = \frac{min(|L_s|, |L_t|)}{|I_s|} \tag{C.4}$$

Les propriétés ayant un faible taux de corrélation et de couverture sont filtrées. Nous considérons que les propriétés clés permettant la sélection des candidats sont associés à un taux élevé de corrélation et de couverture. Les autres propriétés sont utilisées pour le calcul du score total de similarité. Pour pondérer la contribution de ces propriétés dans le calcul de similarité et pour déterminer le seuil de similarité, on utilise une méthode d'optimisation par essaims particulaires appelé PSO [62]. Cette méthode initialise une population de solutions aléatoires appelées particules qui sont mis à jour à chaque 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.

C.4.2 Évaluation de performance

Intuitivement, la similarité entre les événements dépend des propriétés "factuelles", à savoir : le titre (what), la date (when), le lieu (where) et les agents (who). Néanmoins, les taux de corrélation et de couverture entre ces propriétés varient d'un jeu de données à un autre. Dans cette section, nous évaluons la réconciliation d'événements provenant de Last.fm et Upcoming, en utilisant un ensemble de vérification comprenant 300 paires d'événements appariés. Le tableau C.2 montre les coefficients de corrélation et de couverture obtenus dans l'ordre descendant (corrélation $\geqslant 0{,}3$) calculés sur un ensemble de 100 paires d'événements. D'après ce tableau, les propriétés $date$ et $lieu$ ont des valeurs maximales de couverture et de corrélation. Cette dimension spatio-temporel est considérée donc comme un prédicat clé pour la sélection des candidats. Nous constatons également qu'il y a une corrélation importante entre $agents_s$ et $title_t$ qui sont des propriétés sémantiquement différentes, mais véhiculant une relation connotative. Pour sélectionner les candidats cibles pour chaque instance dans la source I_s, nous extrayons les instances de I_t dont la combinaison des propriétés clés $(time, place)$ est supérieur à un seuil α. Les propriétés restantes sont utilisées pour trouver la bonne instance parmi les instances candidates.

Pour l'évaluation des performances, nous avons mis en place quelques expérimentations :

P_{source}	P_{target}	*Correlation*	*Coverage*
$time_s$	$time_t$	**1**	**1**
$place_s$	$place_t$	**0.80**	**1**
$title_s$	$title_t$	0.59	**1**
$agent_s$	$title_t$	0.53	**1**
$(lat_s, long_s)$	$(lat_t, long_t)$	(0.43, 0.97)	0.92
$agent_s$	$description_t$	0.24	0.48

TABLE C.2 – Corrélation et couverture entre les propriétés en utilisant 100 paires d'événements de Last.fm (source) et Upcoming (target)

1. La méthode *LC* : elle est basée sur une combinaison linéaire des scores de similarité de toutes les propriétés sans un mécanisme de sélection de candidats.
2. La méthode *Two-step LC* : elle sélectionne les candidats en utilisant les propriétés clés, et ensuite elle utilise la combinaison linéaire des scores de similarité entre les propriétés restantes.
3. La méthode *Two-step OR* : elle sélectionne les candidats en utilisant les propriétés clés, et ensuite elle se base sur un raisonnement booléen où il suffit que l'un des scores de similarité des propriétés restantes soit supérieur à un seuil.

On utilise la technique d'optimisation PSO pour déduire le poids du 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 [107] qui se base sur un algorithme génétique (GA) pour déduire automatiquement les composants d'une meilleure fonction de similarité. Ces composants comprennent les paires de propriétés à comparer, les distances de similarités, les poids et le seuil. Pour cela, nous avons intégré la métrique *Token-wise* et la métrique temporelle dans Knofuss. Le tableau C.3 résume les résultats obtenus.

	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**

TABLE C.3 – Résultats de différentes méthodes de réconciliation entre Last.fm et Upcoming (avec 50% ensemble d'apprentissage et 50% de test)

Nous constatons que Knofuss produit des appariements avec une bonne précision mais avec un mauvais rappel. Cela est dû à sa stratégie de maximiser la précision en mode supervisé étant donné qu'un appariement erroné est moins tolérable qu'un appariement manquant. Les résultats montrent aussi que les méthodes utilisant la sélection des candidats ont une meilleure performance et ce grâce au filtrage des instances non pertinents. En particulier,

l'approche *Two-step OR* basée sur le raisonnement booléen a réussi à pallier le manque de couverture des propriétés géographiques (latitude et longitude). 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.

C.4.3 Réconciliation en temps réel

Afin de traiter la quantité croissante d'événements crées chaque jour, il convient d'assurer une réconciliation en temps réel. Pour ce faire, nous proposons de construire un service REST qui permet d'aligner les données récemment stockées dans le triple store. Le service est exécuté à un intervalle de temps régulier. A chaque exécution, il envoie deux requêtes SPARQL. La première requête extrait les instances du jeu de données source en les filtrant par la propriété `rdf:type` et la date de stockage représentée par le prédicat `dc:issued`. La deuxième requête extrait, pour chaque instance source, les candidats cibles en utilisant le prédicat `rdf:type` conjointement avec les propriétés clés déterminées à partir des taux de corrélation et de couverture. Pour évaluer la performance de cette méthodologie, nous exécutons le service de réconciliation chaque 10 minutes, et nous mesurons l'intervalle de réconciliation qui représente la différence entre la date de stockage d'une instance et la date de son alignement. La Figure C.3 montre des intervalles de réconciliation assez faibles attestant ainsi l'aspect temps réel.

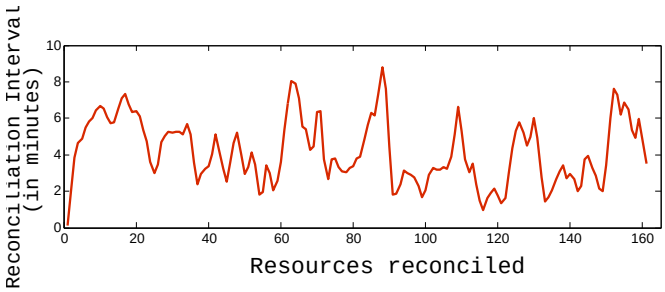

FIGURE C.3 – Évaluation de l'intervalle de réconciliation

C.5 Enrichissement d'événements par des micro-messages

Il est devenu courant aux utilisateurs de partager des micro-messages pour exprimer leurs réflexions et leurs expériences dans les réseaux sociaux. La majorité de ces messages sont, en revanche, non structurés, ce qui empêche les machines d'exploiter pleinement l'information véhiculée. Une partie de ces données concerne des événements du monde réel tels que les débats politiques et les conférences professionnelles où les médias sociaux, notamment Twitter, sont omniprésents pour communiquer. Dans le milieu de la recherche, la fouille des relations entre les événements et les micro-messages a été le sujet de plusieurs travaux récents [140, 124, 8, 54, 121]. La difficulté réside dans la nature textuelle des micro-messages

qui sont souvent bruités et contiennent peu d'information du fait du nombre limité de caractères (p. ex. 140 caractères pour Twitter). Ces caractéristiques textuelles révèlent des défis pour plusieurs applications telles que la détection et le suivi de thèmes émergents. Dans ce travail, nous proposons une approche qui vise à enrichir en temps réel des événements sémantiques par des micro-messages. Cela nécessite donc une technique qui permet de combler le fossé entre les données structurées et les données non structurées.

C.5.1 Structuration des micro-messages

La première étape de notre approche consiste à présenter les micro-messages dans un format structuré et exploitable par les machines. L'enjeu majeur est de capter l'information portée par ces messages et d'accéder à leurs sens afin de pouvoir les réconcilier automatiquement avec des événements. Cet enjeu requiert des outils qui permettent d'extraire le sens de ces messages et les convertir en RDF. Dans ce contexte, les outils de Traitement Automatique des Langues (TAL) ont été développés afin d'extraire de ce qu'on appelle les entités nommées. Cette tâche de reconnaissance des entités nommées (REN) a récemment fait l'objet d'une attention plus soutenue, et vise à identifier et classer des éléments sémantiques de textes qui peuvent se référer à une personne, une organisation, un lieu ou une expression temporelle [99]. Chaque élément, appelé entité nommée, est attaché à une étiquette (par exemple "Roger Federer") et classifié dans une catégorie (par exemple Personne). Par exemple, les entités nommées reconnues dans ce tweet *"Kihara is attending Biophysical Society meeting at San Diego until Tuesday morning #bps12"* sont *NE*= { *(Kihara, Person), (Biophysical Society, Organization), (San Diego, Place)* }. La description en RDF de cet exemple est représenté dans la Figure C.4. Objet de quelques études récentes, la reconnaissance des entités nommées dans les micro-messages gagne de plus en plus de terrain sur les textes traditionnels tels que les articles de presse. Du fait que le nombre de caractères autorisés est limité, les messages sont souvent très bruités contenant par exemple des abréviations (*eske*, *grav*) et des rébus typographique (*lgtps*, *slt*). Ce bruit rend difficile la reconnaissance et la catégorisation des entités nommées, ce qui nécessite un outil performant et adapté à des messages courts et informels.

Bien qu'ils aient un but commun, les outils REN proposés font usage de différents algorithmes, dictionnaires et données d'apprentissage. Leur comportement peut être différent d'un domaine à un autre, ce qui dépend de la diversité des domaines des données d'apprentissage. Cherchant à évaluer les performances de ces outils, Rizzo et al [117] les combinent dans un système appelé NERD. Ce dernier permet de présenter une vue unifiée sur toutes les entités nommées reconnues par les outils intégrés. Ces entités sont classifiées dans une ontologie [10] unifiée qui exprime des correspondances entre les différentes taxonomies utilisées par les outils intégrés. NERD permet de bénéficier des avantages de chaque outil et de reconnaitre ainsi un nombre important d'entités nommées. Il garantit également une proba-

10. http://nerd.eurecom.fr/ontology

```
@prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>.
@prefix sioc:<http://rdfs.org/sioc/ns#>.
@prefix lode:<http://linkedevents.org/ontology/>.
@prefix owl:<http://www.w3.org/2002/07/owl#>.
@prefix dcterms:<http://purl.org/dc/terms/>.
@prefix dbpedia-owl:<http://dbpedia.org/ontology/>.
<http://data.linkedevents.org/tweet/ab675d40-7f38-4fb0-93a1-1cf352f03ee5>
 a sioc:Post;
 sioc:id "173693229584232448";
 sioc:content "Kihara is attending Biophysical Society meeting at San Diego
 until Tuesday morning #bps12";
 sioc:hasCreator <http://twitter.com/kiharalab>;
 lode:illustrate <http://www.biophysics.org/2012meeting>;
 owl:sameAs      <http://twitter.com/kiharalab/status/173693229584232448> ;
 dcterms:date "2012-02-26 09:57:47+00:00";
 dcterms:subject [ a dbpedia-owl:Person ; rdfs:label "Kihara" ],
                 [ a dbpedia-owl:Location ; rdfs:label "San Diego" ],
                 [ a dbpedia-owl:Organization ; rdfs:label "Biophysical Society" ].
```

FIGURE C.4 – Description en RDF/Turtle d'un micro-message provenant de Twitter

bilité plus élevée qu'une entité nommée soit correctement catégorisée tel qu'il a été prouvé dans [117].

C.5.2 Lier des micro-messages aux événements

L'enrichissement d'événements par des micro-messages consiste plus concrètement en la découverte des liens représentés par la propriété `lode:illustrate`. Après la structuration des micro-messages, l'enrichissement se réfère finalement au problème de réconciliation d'instances RDF. Il importe donc de déterminer quelles sont les propriétés et les valeurs à comparer ensemble. Pour ce faire, nous proposons d'exploiter le recouvrement en termes de concepts entre l'ontologie qui décrit les événements et celle qui décrit les catégories des entités nommées. Nous créons manuellement des correspondances entre les concepts sémantiquement similaires, par exemple, la classe Personne peut correspondre à plusieurs catégories d'entités nommées telles que Auteur, Étudiant, Professeur, etc. Pour chaque mico-message entrant, nous utilisons NERD pour extraire les entités nommées. Ensuite, pour chaque entité nommée ayant comme étiquette *NE-label* et comme catégorie *NE-class*, nous appliquons l'une des deux opérations suivantes pour extraire les événements candidats pour le micro-message entrant :

- La première opération sélectionne les événements qui sont associés à une ressource ayant comme étiquette *NE-label*, et une classe sémantiquement similaire à *NE-class*.
- La deuxième opération sélectionne les événements qui sont associés à des littéraux qui contiennent *NE-label*.

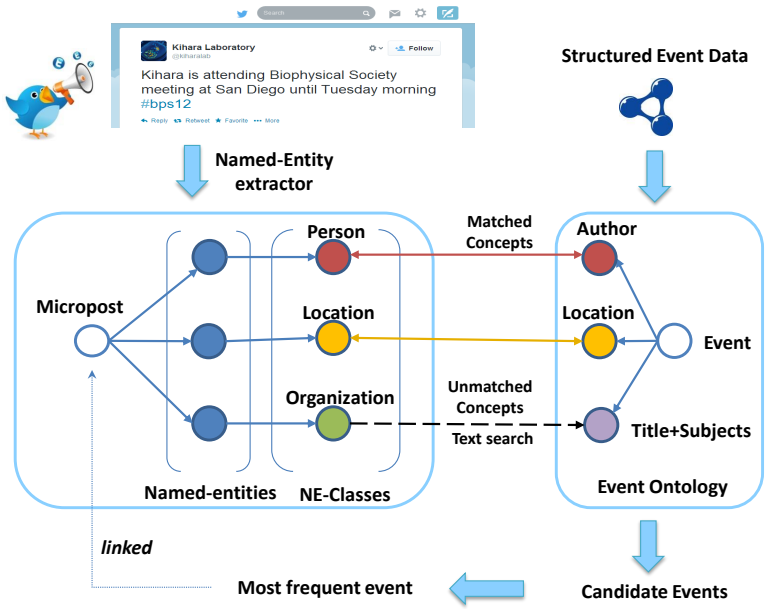

FIGURE C.5 – Aperçu sur l'approche d'enrichissement des événements avec des micro-messages exploitant les entités nommées

La Figure C.5 illustre notre approche. Nous utilisons des requêtes SPARQL pour générer une liste d'événements candidats en appliquons les deux opérations. L'événement pertinent correspond à celui qui a la fréquence la plus élevée dans la liste. Lorsque plus d'un événement pertinent existe, nous sélectionnons celui ayant la date la plus proche de date de création du micro-message. Cette stratégie a l'avantage de combler le fossé entre les données structurées et non structurées, et peut être exploitée dans plusieurs domaines.

C.5.3 Cas d'usage et évaluation

L'évaluation de notre approche a été menée sur un corpus provenant d'un jeu de données appelé Semantic Web Dog Food [11] (SWDF). Il s'agit d'une bibliographie de différentes conférences dans le domaine du Web sémantique. Il fournit une description granulaire et sémantique sur chaque conférence. Plus précisément, il décrit le programme détaillé, les travaux présentés, les participants et d'autres méta-données tels que le lieu et la date. Comme un cas d'usage, nous prenons l'exemple de la Conférence Internationale de Web sémantique (ISWC) qui a eu lieu en octobre 2011. La description de cette conférence est représentée par l'ontologie SWC [12] qui décrit la conférence principale et ses sous-événements tels que les présentations des papiers, les tutoriels et les sessions de démonstrations. Nous avons collecté les micro-messages de Twitter durant les six jours de la conférence en utilisant le hashtag pertinent (*#iswc2011*). Nous avons construit manuellement un ensemble de données de véri-

11. http://data.semanticweb.org/
12. http://data.semanticweb.org/ns/swc/ontology

fication composé de micro-messages qui concernent les sous-événements de la conférence. Nous comparons trois approches de réconciliation, à savoir : (1) notre approche basée sur la reconnaissance des entités nommées, (2) la même approche mais basée cette fois-ci sur des mots-clé (sans catégories) en utilisant le service en ligne AlchemyAPI [13], (3) et une approche hybride qui combine les deux précédentes. Le tableaue C.4 présente les résultats obtenus. Ces derniers montrent relativement une bonne performance si on considère la manque des méta-données importantes sur certains sous-événements dans le corpus SWDF. Nous constatons également que l'approche basée sur les entités nommées réussit à filtrer plus de correspondances erronées que celle basée sur les mots-clés. Ceci peut être justifié par le sens donné aux micro-messages à travers la détection et la classification des entités nommées.

	Precision	Recall	F-score
Named-entity-based algorithm	61%	49%	54%
Keyword-based algorithm	40%	55%	46%
Hybrid approach (Named-entity + Keyword)	43%	64%	51%

TABLE C.4 – Précision-Rappel de l'approche de réconciliation entre des événements et des micro-messages

C.6 Approche hybride 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 rapidement insuffisantes, ce qui rend indispensable la présence d'un système de recommandation. En particulier, la recommandation d'événements met en jeu plusieurs facteurs comme le temps, le lieu, la popularité des artistes et le degré d'amitié avec les participants. Cette pluralité rend inefficace les systèmes de recommandation classiques tels que ceux basés sur le contenu ou sur le filtrage collaboratif. Nous proposons donc un système hybride qui combine ces deux approches tout en exploitant les technologies du Web sémantique.

C.6.1 Recommandation thématique dans le Web sémantique

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 (voir Appendix B.3). Le système compare le profil d'un objet avec d'autres profils intéressants afin de prédire la préférence de l'utilisateur envers cet objet. Pour représenter le profil d'un objet, la méthode la plus commune est la représentation des

13. http://www.alchemyapi.com/api/keyword-extraction

méta-données en utilisant TF-IDF (*Term Frequency-Inverse Document Frequency*) [118] 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 données pour transformer une description non structurée en une forme structurée. Cette extraction devient extrêmement simple avec les technologies de Web sémantique grâce à la structuration des données dans une ontologie. Ainsi, nous avons adopté la méthode proposée dans [32] qui considère intuitivement que deux ressources dans un graphe RDF sont similaires si elles sont le sujet de deux triplets ayant le même prédicat et le même objet (où un triplet = <sujet,prédicat,objet>). Pour chaque objet o lié à l'événement e_i suivant le prédicat p, le poids TF-IDF est :

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

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 d'événements, $m_{o,p}$ est le nombre d'événements liés à l'objet o via le prédicat p. La similarité entre deux événements e_i et e_j suivant le prédicat p est calculée à l'aide de la métrique Cosine :

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

Cependant, cette approche est limitée lorsque la matrice d'adjacence est creuse parce qu'elle est associée à un prédicat discriminant. Par exemple, le vecteur représenté par le prédicat `lode:atPlace` a une seule valeur non nulle puisqu'un événement est souvent associé à un seul lieu. Afin de pallier ce problème, nous définissons tout d'abord l'équation suivante pour mesurer le pouvoir discriminant d'un prédicat [131] :

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

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. Nous proposons ensuite 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 C.5 devient :

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

où H est l'ensemble des objets qui sont déjà liés à l'événement e_i via le prédicat p. Avec l'interpolation de ces similitudes, nous avons réussi à diviser par 3 le nombre de cases vides

des matrices d'adjacence.

C.6.2 Recommandation thématique d'événements

Pour prédire la participation d'un utilisateur u à un événement e_i, on utilise les similitudes entre les événements de la manière suivante :

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

où P est l'ensemble des prédicats partagés entre les événements e_i et e_j, E_u est l'ensemble d'événements passés dans le profil de l'utilisateur u, et α_p est le poids attribué au prédicat p reflétant sa contribution dans la recommandation. Nous utilisons par la suite des méthodes d'optimisation pour estimer les valeurs de α_p. Suivant l'ontologie LODE, 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 fera le sujet d'une contribution future. De plus, nous exploitons le jeu de données DBpedia et son appariement avec Event-Media pour enrichir la description des artistes.

L'objectif de la recommandation thématique est de suggérer des produits qui répondent au profil de l'utilisateur. En revanche, elle devient inefficace si l'utilisateur a interagi avec des objets marqués par une grande diversité thématique. Par exemple, un événement peut concerner plusieurs thèmes comme le cas des grands festivals qui couvrent plusieurs genres musicaux. Cependant, un utilisateur qui participe à un événement peut être intéressé à un nombre limité de thèmes. Pour déceler les intérêts effectifs d'utilisateurs, nous proposons une méthode basée sur la technique LDA (Allocation de Dirichlet Latente [15]) permettant de modéliser des thèmes dans un corpus. 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, nous calculons la variance pour chaque thème t dans tout l'ensemble d'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. Nous classifions les événements dans le profil utilisateur en deux catégories : la première catégorie inclut les événements qui correspondent aux pics intérêts (θ^t élevé), la deuxième catégorie contient le reste d'événements. Chaque catégorie est associée à un poids β que nous allons estimer par des méthodes d'optimisation. Ainsi, la recommandation thématique devient :

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

où $\beta_p = 1$ si le prédicat p est différent de `dc:subject`, sinon $\beta_{subject}$ est un poids qui varie selon que l'événement e_j correspond à un pic d'intérêt ou non.

C.6.3 Recommandation basée sur le filtrage collaboratif

Une forme d'interaction sociale est la participation collaborative (co-participation) aux événements. Nous supposons que le nombre d'événements communs entre deux participants est relatif au degré de leur lien "d'amitié" ou similarité. Dans notre approche, nous visons non seulement à considérer la similarité entre deux participants, mais aussi la similarité entre 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 le filtrage collaboratif :

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

où C est l'ensemble des co-participants qui ont confirmé leur présence à l'événement e, E_i est l'ensemble d'événements passés 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 considère, d'une part, la similarité avec chaque participant individuellement, et d'une autre part, la similarité avec un groupe de participants où nous supposons que le nombre d'événements partagés reflète la force de lien d'amitié.

C.6.4 Recommandation hybride

Pour exploiter à la fois la recommandation thématique et le filtrage collaboratif, nous proposons un système hybride par pondération d'une combinaison linéaire. Combinant les équations (C.10) et (C.11), nous proposons la fonction de recommandation suivante :

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

où α_{cf} est le poids attribué au filtrage collaboratif, estimé conjointement avec les poids de l'Équation C.10 de la recommandation thématique .

C.6.5 Expérimentations et évaluation

Nous évaluons notre approche sur un ensemble contenant 2436 événements provenant de Last.fm et situés dans la ville *Londres* qui regroupe un nombre important d'utilisateurs actifs. Les critères d'évaluation utilisés sont 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 données, contre 30 % pour le jeu de test. Le tableau C.5 souligne la réduction des cases nulles dans les matrices d'adjacences associées aux prédicats discriminants. Ces résultats attestent l'efficience de l'interpolation des similarités ainsi que l'enrichissement des données.

Task	location	agent	subject
(1)	0.9942	0.9174	0.3175
(2)	0.6854	0.7392	0.2843

TABLE C.5 – Taux de sparsité des matrices d'adjacences avant (1) et après (2) l'interpolation des similarités (pour location et agent) et l'enrichissement avec DBpedia (pour subject)

Pour estimer les poids de notre fonction de recommandation hybride, nous utilisons trois méthodes d'optimisation : la régression linéaire qui minimise la mesure d'erreur RMSE, L'algorithme génétique (GA) [143] et l'optimisation par essaims particulaires (PSO) [62] qui maximisent la précision.

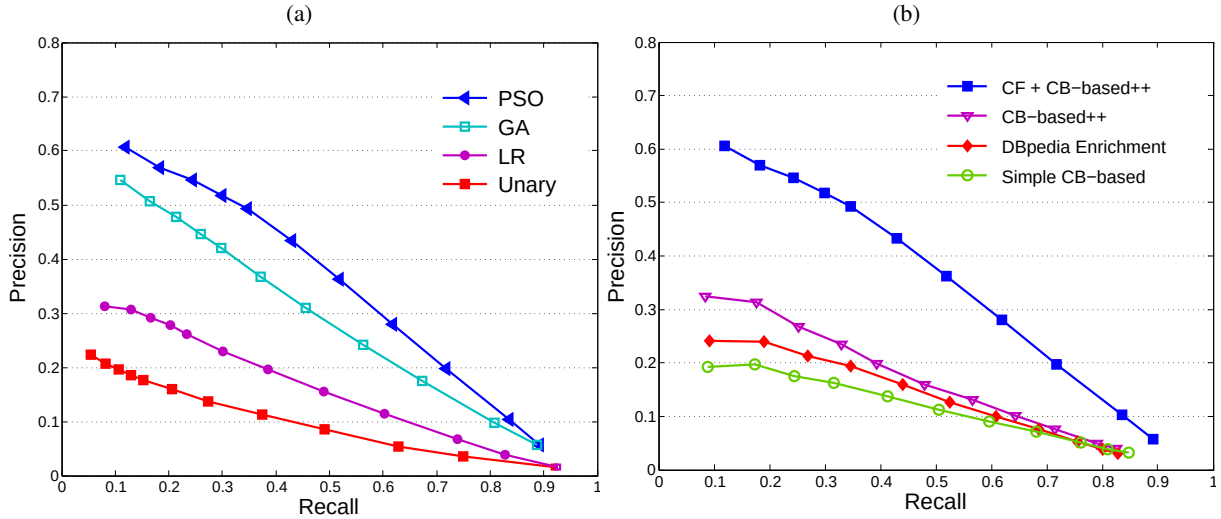

FIGURE C.6 – (a) Évaluation de différentes méthodes d'apprentissage ; (b) Évolution de performance du système en intégrant l'enrichissement avec DBpedia, la diversité thématique (CB-based++) et le filtrage collaboratif (CF)

Les résultats obtenus sont présentés dans la Figure C.5(a). La méthode PSO a la meilleure performance où nous avons constaté une convergence rapide vers la solution optimale par rapport à l'algorithme génétique. Dans ce qui suit, nous l'utilisons pour le reste d'expérimentation. Nous examinons, ensuite, l'évolution de la performance du système en intégrant par ordre l'enrichissement à l'aide de DBpedia, la diversité thématique du profil utilisateur et le filtrage collaboratif. Nous observons dans la Figure C.5(b) que l'enrichissement des données avec DBpedia a légèrement amélioré la recommandation. En effet, l'introduction des données plus cohérentes est un avantage pour réduire le bruit induit par l'annotation collective dans un service social. La modélisation effective des intérêts d'utilisateurs a également amélioré les résultats (approche CB-based++). D'ailleurs, nous avons observé que le poids attribué aux événements situés dans les pics d'intérêt est quatre fois plus important que le poids attribué au reste d'événements. Enfin, l'intégration du filtrage collaboratif (CF)

a augmenté considérablement la performance du système.

C.7 Détection de communautés sémantiques et recouvrantes

Au lendemain d'une révolution technologique, les réseaux sociaux sont devenus un espace privilégié d'échange, de partage et de communication. Les événements constituent, en particulier, l'épicentre de plusieurs interactions sociales virtuelles ou réelles. De ces interactions, il se dégage un réseau d'individus connectés, ce que l'on désigne par *un réseau événementiel* ou EBSN (*Event-based Social Network en anglais.*). Ce dernier revêt deux types de réseaux : le premier est appelé *EBSN virtuel* construit à partir des activités en ligne tels que le partage de médias et de micro-messages portant sur les mêmes événements ; le deuxième appelé *EBSN réel* reflète la présence physique des participants aux mêmes événements [87]. L'analyse de ce réseau est un moyen important pour parvenir à une compréhension fine des individus ainsi que des groupes d'individus. L'un des enjeux majeurs de cet analyse est la détection de communautés qui consiste à regrouper ensemble des individus ayant potentiellement des rôles similaires. Ces communautés dotés d'un sens particulier (groupe d'amis, équipe, famille, etc.) servent comme brique de base pour d'autres objectifs tels que la recommandation et la personnalisation de l'expérience utilisateur [72, 109]. Dans ce travail, nous étudions la détection de communautés dans un réseau événementiel, ce qui peut servir comme un moyen de personnalisation dans plusieurs référentiels d'événements. Bien que relativement récent, le problème de la détection de communautés dans les réseaux sociaux a déjà suscité plusieurs travaux. La plupart d'entre eux se focalisent sur l'analyse des liens et des propriétés structurelles afin de former des groupes de nœuds fortement liés entre eux, et plus faiblement liés avec le reste du réseau. Ils visent la détection de communautés disjointes, c'est à dire qu'elles ne partagent aucun membre en commun [23, 103, 105]. Toutefois, cela n'est toujours pas conforme à la réalité et n'assure pas la détection de communautés sémantiquement homogènes appelées *communautés sémantiques*. Par exemple dans un cadre événementiel, l'analyse de la co-présence aux événements peut mener à des groupes d'individus qui sont potentiellement des amis, mais il n'y aucune garantie qu'ils partagent des intérêts pour des thématiques similaires. Il importe ainsi de prendre en compte, non seulement la connectivité entre les nœuds, mais aussi la dimension sémantique. Notre objectif consiste à détecter des communautés recouvrantes et sémantiques dans un réseau événementiel.

C.7.1 Similarité d'événements dans l'espace latent

Dans le but de détecter des communautés sémantiques, l'une des méthodes est d'utiliser un algorithme de clustering basé sur la comparaison d'utilisateurs. Cette méthode risque cependant d'être inopérante quand il s'agit de traiter une énorme quantité d'utilisateurs. Il importe donc de raisonner sur la comparaison d'événements qui sont beaucoup moins nombreux, ce qui permet de réaliser un gain important en coût d'exécution. Cette comparaison

devrait assurer une prise en compte conjointe de deux types d'information : structurelle et sémantique. Pour ce faire, nous considérons un événement dans deux espaces : un espace utilisateur et un autre sémantique. Un événement dans l'espace utilisateur est représenté par un vecteur binaire où la valeur de la ligne i indique la participation ou non (*resp.* 1 ou 0) de l'utilisateur i. De même, il est représenté par un vecteur binaire de tags dans l'espace sémantique. Les matrices d'adjacence qui en découlent restent néanmoins difficile à analyser à cause de la haute dimensionnalité (nombre d'utilisateurs et de tags) et de la variabilité des observations. Ceci nécessite une technique qui permet de représenter les données originelles dans un espace de dimension réduite tout en minimisant la perte d'information. La technique la plus répandue en réduction de données est la décomposition orthogonale aux valeurs propres. Elle permet de projeter les données sur des plans principaux permettant de représenter au mieux les corrélations entre les variables et de retirer la redondance. En particulier, la méthode basée sur la décomposition en valeurs singulières SVD (Singular Value Decomposition) est appliquée dans le cas des matrices non carrées. Soit A une matrice rectangulaire d'adjacence événement-utilisateur, de dimension $(m \times n)$ où $A_{i,j}$ décrit la relation entre l'événement i et l'utilisateur j, et soit r le rang de A. La décomposition SVD de A est la factorisation $U\Sigma V^T$ où U est la matrice des vecteurs propres de AA^T de dimension $(m \times r)$, V^T est la matrice des vecteurs propres de A^TA de dimension $(r \times n)$, et Σ est une matrice diagonale de dimension $(r \times r)$ contenant les valeurs singulières rangées en ordre décroissant. La représentation de l'événement $\tilde{e}_i$ dans l'espace latent (réduit) de l'utilisateur peut être formalisé par :

$$\begin{aligned}
& e_i(u_1,u_2,...,u_n) = \quad \{U\Sigma\}_i \cdot V^T \\
\Leftrightarrow \quad & e_i(u_1,u_2,...,u_n) = \quad \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_r) \cdot V^T \\
\Leftrightarrow \quad & \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k) = \quad e_i(u_1,u_2,...,u_n) \cdot V \\
\Leftrightarrow \quad & \tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_r) = \quad A \cdot V
\end{aligned}$$

où $e_i(u_1,u_2,...,u_n)$ et $\tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k)$ sont les vecteurs de l'événement i respectivement dans l'espace original et l'espace latent. La représentation dans l'espace latent capture l'information importante en exploitant la corrélation entre les utilisateurs. Notre approche repose, en particulier, sur la méthode de classification spectrale qui permet de localiser l'information sur le partitionnement. Telle qu'utilisée dans [31], cette méthode normalise d'abord la matrice A en une matrice $A_n = D_1^{-1/2} A D_2^{-1/2}$, où D_1 et D_2 sont des matrices diagonales qui représentent respectivement les degrés d'événements et d'utilisateurs (un degré d'un nœud est le nombre de ses liens avec d'autres nœuds). Comme prouvé dans [31], les premiers vecteurs singuliers, à l'exception du premier vecteur, contiennent l'information sur le partitionnement. Ainsi, il suffit de choisir un sous ensemble de vecteurs singuliers à droite $V'_n = (v_2, v_3, ..., v_k)$ avec $k \ll r$. Enfin, la représentation d'un événement dans l'espace latent d'utilisateur est formalisée par :

$$\tilde{e}_i(\tilde{u}_1,\tilde{u}_2,...,\tilde{u}_k) = A_n \cdot V'_n \tag{C.13}$$

Nous appliquons la même méthode sur la matrice d'adjacence événement-tag pour représenter un événement dans un espace latent sémantique. Ensuite, la distance Cosine est appliquée pour calculer la similarité S_u et S_t entre les événements respectivement dans l'espace utilisateur et l'espace sémantique. La combinaison linéaire de ces deux similarités forme la fonction de distance entre deux événements, avec comme paramètre α qui contrôle la balance entre les deux similarités.

C.7.2 Clustering hiérarchique et formation de communautés

Le clustering hiérarchique part d'une structure dans laquelle chaque nœud est identifié comme un cluster. A chaque itération, il calcule la distance entre les clusters et fusionne les deux clusters les plus proches en un nouveau cluster. Il forme ainsi une structure hiérarchique, et il s'arrête lorsqu'il n'y a plus qu'un seul cluster ou lorsque le critère d'arrêt est satisfait. Nous utilisons cette technique afin de produire des communautés sémantiques tout en maximisant une fonction que l'on désigne par la modularité sémantique *SemQ*. Notre approche appelé SMM (Semantic Modularity Maximization) vise à maximiser la distance sémantique interne aux communautés et la minimiser entre les communautés. Soit C l'ensemble de clusters d'événements détectés, la modularité sémantique est formalisée par :

$$IntraSem = \frac{1}{|C|} \sum_{C_k \in C} \left(\frac{}{\sum\limits_{\substack{i,j \in C_k \\ j>i,\ S_t(i,j)\neq 0}} S_t(i,j)} \right) \tag{C.14}$$

$$InterSem = \frac{1}{|C|} \sum_{C_k \in C} \left(\frac{}{\sum\limits_{\substack{i \in C_k,\ j \in C_l \\ l>k,\ S_t(i,j)\neq 0}} S_t(i,j)^2} \right) \tag{C.15}$$

Finally, the semantic modularity *SemQ* is defined as follows :

$$SemQ = IntraSem - InterSem \tag{C.16}$$

La valeur maximale de la modularité sémantique correspond au critère d'arrêt de notre processus. Après avoir construit les communautés d'événements, nous déduisons les communautés d'utilisateurs. Pour un utilisateur u_i donné, nous calculons son degré d'appartenance à chaque communauté d'événements C_j. Soit $C_j(u_i)$ est le degré d'utilisateur u_i dans la communauté C_j, et $D(u_i)$ est son degré dans tout le réseau, le degré d'appartenance correspond au rapport de $D_{C_j}(u_i)$ sur $D(u_i)$. L'utilisateur u_i appartient à la communauté C_j si son degré d'appartenance est supérieur à un seuil. Ce seuil est la moyenne des degrés d'appartenance strictement positives. Cette stratégie permet de détecter des communautés recouvrantes contenant des individus fortement liés et ayant des intérêts sur des thèmes sémantiquement homogènes.

C.7.3 Évaluation de la qualité des communautés

L'évaluation de la qualité des communautés sémantiques devrait prendre en compte conjointement l'information structurelle et la sémantique. Nous adoptons ainsi la métrique $PurQ_\beta$ proposée par Zhao et al. [146] qui se base sur la pureté sémantique et la modularité. La pureté sémantique mesure la moyenne des fractions des tags appartenant au même thème dans une communauté. Quant à la modularité Q, elle a été définie par Newman et al [103] et elle est couramment utilisée pour évaluer la qualité de connectivité des communautés. Soit β un paramètre qui contrôle la balance entre la pureté sémantique et la modularité Q, la métrique $PurQ_\beta$ est formalisée par :

$$PurQ_\beta = \frac{(1+\beta^2)(Purity \cdot Q)}{\beta^2 Purity + Q} \tag{C.17}$$

Nous avons expérimenté notre approche sur la base de quatre réseaux événementiels comme illustré dans le tableau C.6. Nous avons tout d'abord constaté que la pureté sémantique décroit drastiquement avec l'augmentation de la modularité Q. De faibles valeurs $\alpha \in$[0,0.5] sont alors privilégiées pour mettre l'accent sur la similarité sémantique.

EBSN	Users	Events	Tags	Edges	Density	ClustCoeff
Last.fm Offline	2847	915	272	95897	0.0237	0.1144
Last.fm Online	1729	470	248	9936	0.0067	0.398
Flickr Online	868	375	221	7071	0.0188	0.2624
Twitter Online	768	275	166	14237	0.0483	0.4852

TABLE C.6 – Statistiques sur les réseaux événementiels dans les données d'expérimentation

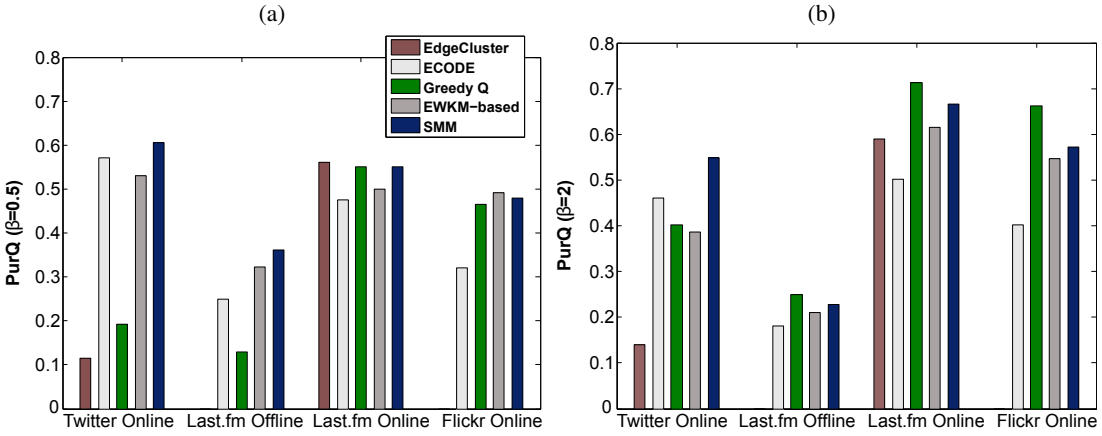

FIGURE C.7 – Évaluation de $PurQ_\beta$ avec (a) $\beta = 0.5$ et (b) $\beta = 2$

La Figure C.7 montre la comparaison de notre approche avec quatre algorithmes existants, à savoir *EdgeCluster* [138], *ECODE* [83], *Greedy Q*[104] et *EWKM-based* [146].

Nous remarquons que ces algorithmes ont des performances plus ou moins proches pour les réseaux Last.fm Online et Flickr Online. Ces deux réseaux sont caractérisés par des interactions sporadiques et par de faibles densités. Au maximum, un utilisateur est associé aux deux événements, ce qui engendre une très faible variation dans son profil thématique. Les communautés détectées par l'approche Greedy Q qui maximise la modularité Q ont une variation sémantique faible, induisant une bonne pureté sémantique. En revanche, cette pureté décroit considérablement pour les réseaux Twitter Online et Lastfm Offline qui sont relativement assez denses. La prise en compte de la dimension sémantique devient donc primordiale tel est le cas de notre approche SMM et de la méthode *EWKM-based*. En revanche, notre approche a l'avantage d'employer un clustering hiérarchique qui ne requiert aucune connaissance préalable du nombre de clusters, et elle permet de détecter des communautés avec une meilleure modularité. De plus, nous avons comparé les profils d'utilisateurs représentés par des tags et appartenant au même communauté. Notre approche a réussi à regrouper le plus d'utilisateurs ayant des profils sémantiquement similaires et à former des communautés cohésives.

C.8 Conclusion

Dans le cadre de ce travail, nous avons étudié l'intégration des données événementielles réparties sur plusieurs médias sociaux. L'objectif est de concevoir une architecture qui fait face à l'hétérogénéité des données et à l'évolution dynamique du Web 2.0. Nous avons particulièrement démontré l'avantage du Web sémantique pour assurer une intégration flexible et extensible. Ainsi, nous avons présenté une modélisation sémantique des événements, ainsi que la construction d'un jeu de données appelé EventMedia composé de descriptions d'événements et de média les illustrant. Ensuite, nous avons proposé un système de réconciliation basé sur une approche indépendante du domaine pour aligner les données événementielles structurées. Notre approche souligne l'importance de la corrélation et la couverture des propriétés pour surmonter l'hétérogénéité. Nous avons aussi enrichi les événements par des micro-messages en exploitant les entités nommées pour combler le fossé entre les données structurées et celles non structurées. Finalement, des techniques de personnalisation ont été proposées. En particulier, nous avons mis en valeur l'avantage du Web sémantique dans un système de recommandation, et nous avons proposé une approche pour assurer la détection de communautés sémantiques et recouvrantes.

Les résultats obtenus dans cette thèse ouvrent la voie vers de nouvelles directions de recherches. Une étude plus approfondie pourrait être envisagée sur les différents types de relations entre les événements tels que la causalité, la temporalité et la spatialité. En outre, il sera d'un avantage considérable que les systèmes de réconciliation et de recommandation puissent prendre en compte la dynamicité des données. Cela pourrait être assuré par l'utilisation des algorithmes non supervisés et incrémentales faisant face à la variation des

données dans le temps.

Bibliography

[1] James Allan. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publishers, 2002. (Cited on page 8.)

[2] James Allan, Ron Papka, and Victor Lavrenko. On-line new event detection and tracking. In *21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, 1998. (Cited on pages 7, 8, 13, and 40.)

[3] James F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, November 1983. (Cited on page 118.)

[4] James F. Allen and George Ferguson. Actions and events in interval temporal logic. Technical report, University of Rochester, 1994. (Cited on page 13.)

[5] Rohit Ananthakrishna, Surajit Chaudhuri, and Venkatesh Ganti. Eliminating fuzzy duplicates in data warehouses. In *28th International Conference on Very Large Data Bases*, Hong Kong, China, 2002. (Cited on pages 41 and 127.)

[6] James E. Baker. Reducing bias and inefficiency in the selection algorithm. In *2nd International Conference on Genetic Algorithms and Their Application*, Cambridge, Massachusetts, USA, 1987. (Cited on page 129.)

[7] H. Becker, M. Naaman, and L. Gravano. Event Identification in Social Media. In *12th International Workshop on the Web and Databases (WebDB'09)*, Providence, USA, 2009. (Cited on page 2.)

[8] Hila Becker, Mor Naaman, and Luis Gravano. Beyond trending topics: Real-world event identification on twitter. In *5th International Conference on Weblogs and Social Media*, Barcelona, Spain, 2011. (Cited on pages 13, 51, and 145.)

[9] Cosmin Adrian Bejan and Sanda Harabagiu. Unsupervised event coreference resolution with rich linguistic features. In *48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, 2010. (Cited on page 40.)

[10] Tim Berners-Lee. Uniform Resource Identifier (URI): Generic Syntax - RFC 3986 (January 2005). `http://tools.ietf.org/html/rfc3986`. (Cited on page 15.)

[11] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):34–43, 2001. (Cited on page 14.)

[12] Diego Berrueta, Dan Brickley, Stefan Decker, and Sergio Fernández andChristoph Görn et al. SIOC Core Ontology Specification (March 2010). `http://rdfs.org/sioc/spec`. (Cited on pages 33 and 140.)

[13] Aline Bessa, Alberto H. F. Laender, Adriano Veloso, and Nivio Ziviani. Alleviating the sparsity problem in recommender systems by exploring underlying user communities. In *6th Alberto Mendelzon International Workshop on Foundations of Data Management*, Ouro Preto, Brazil, 2012. (Cited on page 119.)

[14] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009. (Cited on page 17.)

[15] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. (Cited on pages 81, 89, 98, 107, and 151.)

[16] DCMI Usage Board. DCMI Metadata Terms (June 2012). `http://dublincore.org/documents/dcmi-terms`. (Cited on page 31.)

[17] David Booth, Hugo Haas, Francis McCabe, Eric Newcomer, Michael Champion, Chris Ferris, and David Orchard. Web Services Architecture (February 2004). `http://www.w3.org/TR/ws-arch`. (Cited on page 23.)

[18] Dan Brickley and R.V. Guha. RDF Schema 1.1 - W3C Recommendation (February 2014). `http://www.w3.org/TR/rdf-schema`. (Cited on page 16.)

[19] Dan Brickley and Libby Miller. FOAF Vocabulary Specification (January 2014). `http://xmlns.com/foaf/spec`. (Cited on page 31.)

[20] Roberto Casati and Achille Varzi. Events, 2010. (Cited on page 7.)

[21] S. Castano and S. Montanelli. Semantic self-formation of communities of peers. In *Proc. of the ESWC Workshop on Ontologies in Peer-to-Peer Communities*, Heraklion, Greece, May 2005. (Cited on page 97.)

[22] Jilin Chen, Werner Geyer, Casey Dugan, Michael Muller, and Ido Guy. Make new friends, but keep the old: recommending people on social networking sites. In *27th International Conference on Human Factors in Computing Systems*, Boston, MA, USA, 2009. (Cited on page 131.)

[23] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70:066111, 2004. (Cited on pages 97, 98, and 154.)

[24] W. Cohen, P. Ravikumar, and S. Fienberg. A Comparison of String Distance Metrics for Name-Matching Tasks. In *1st International Workshop on Information Integration on the Web (IIWeb'03)*, pages 73–78, Acapulco, Mexico, 2003. (Cited on pages 45 and 129.)

[25] Ilaria Corda, Vania Dimitrova, and Brandon Bennett. An ontological approach to unveiling connections between historical events. In *International Workshop on Intelligent Exploration of Semantic Data (IESD'12)*, Galway,Ireland, 2012. (Cited on pages 13, 40, and 118.)

[26] Chris Cornelis, Xuetao Guo, Jie Lu, and Guanquang Zhang. A fuzzy relational approach to event recommendation. In *2nd Indian International Conference on Artificial Intelligence*, Pune, India, 2005. (Cited on pages 3, 13, 83, and 135.)

[27] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *ACM Conference on Recommender Systems (RecSys'10)*, Barcelona, Spain, 2010. (Cited on page 93.)

[28] Juan David Cruz, Cécile Bothorel, and François Poulet. Entropy based community detection in augmented social networks. In *International Conference on Computational Aspects of Social Networks (CASoN)*, pages 163–168, Salamanca, Spain, 2011. (Cited on pages 4 and 98.)

[29] Richard Cyganiak and Anja Jentzsch. The Linking Open Data cloud diagram (September 2011). `http://lod-cloud.net`. (Cited on page 17.)

[30] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990. (Cited on page 131.)

[31] Inderjit S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *7^{th} ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2001. (Cited on pages 103 and 155.)

[32] Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, Davide Romito, and Markus Zanker. Linked open data to support content-based recommender systems. In *8^{th} International Conference on Semantic Systems, I-SEMANTICS*, Graz, Austria, 2012. (Cited on pages 84, 85, 86, and 150.)

[33] M. Doerr. The CIDOC Conceptual Reference Module: an Ontological Approach to Semantic Interoperability of Metadata. *AI Magazine*, 24(3):75–92, 2003. (Cited on page 31.)

[34] Xin Dong, Alon Halevy, and Jayant Madhavan. Reference reconciliation in complex information spaces. In *ACM SIGMOD International Conference on Management of Data*, Baltimore, Maryland, USA, 2005. (Cited on page 2.)

[35] Simon Dooms, Toon De Pessemier, and Luc Martens. A user-centric evaluation of recommender algorithms for an event recommendation system. In *Workshop on Human Decision Making in RecSys'11*, Chicago, IL, USA, 2011. (Cited on page 84.)

[36] Russell C. Eberhart and Yuhui Shi. Particle swarm optimization: developments, applications and resources. In *IEEE Congress on Evolutionary Computation*, volume 1, pages 81–86, 2001. (Cited on page 131.)

[37] Martin Ebner and Wolfgang Reinhardt. Social networking in scientific conferences - twitter as tool for strengthen a scientific community. In *4^{th} European Conference on Technology Enhanced Learning,*, Nice, France, 2009. (Cited on page 52.)

[38] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer-Verlag New York, Inc., 2007. (Cited on page 55.)

[39] Maryam Fatemi and Laurissa Tokarchuk. A community based social recommender system for individuals & groups. In *5th International Conference on Social Computing*, Washington, DC, USA, 2013. (Cited on page 119.)

[40] A. Fialho, R. Troncy, L. Hardman, C. Saathoff, and A. Scherp. What's on this evening? Designing User Support for Event-based Annotation and Exploration of Media. In *1st International Workshop on EVENTS - Recognising and tracking events on the Web and in real life*, pages 40–54, Athens, Greece, 2010. (Cited on pages 1, 11, 37, 95, 133, and 141.)

[41] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. In *25th International Conference on Very Large Data Bases*, Edinburgh, UK, 1999. (Cited on pages 103 and 119.)

[42] G. Gouriten and P. Senellart. API BLENDER: A Uniform Interface to Social Platform APIs. In *21st World Wide Web Conference*, Lyon, France, 2012. (Cited on page 24.)

[43] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *National Academy of Sciences of the United States of America*, 101:5228–5235, 2004. (Cited on page 107.)

[44] W3C OWL Working Group. OWL 2 Web Ontology Language - W3C Recommendation (December 2012). `http://www.w3.org/TR/owl2-overview`. (Cited on page 17.)

[45] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220, 1993. (Cited on page 17.)

[46] Junwei Han, Jianwei Niu, Alvin Chin, Wei Wang, Chao Tong, and Xia Wang. How online social network affects offline events: A case study on douban. In *9th International Conference on Ubiquitous Intelligence and Computing*, Fukuoka, September, 2012. (Cited on page 101.)

[47] Benjamin Heitmann, Richard Cyganiak, Conor Hayes, and Stefan Decker. An empirically grounded conceptual architecture for applications on the web of data. *Trans. Sys. Man Cyber Part C*, 42:51–60, 2012. (Cited on page 67.)

[48] Martin Hepp. Tickets Ontology. `http://purl.org/tio/ns`. (Cited on page 118.)

[49] Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. Explaining collaborative filtering recommendations. In *the 2000 ACM Conference on Computer Supported Cooperative Work*, pages 241–250, Philadelphia, Pennsylvania, United States, 2000. (Cited on page 131.)

[50] Antonio Garrote Hernández and María N. Moreno García. Restful writable apis for the web of linked data using relational storage solutions. In *Workshop on Linked Data on the Web (LDOW'11)*, Hyderabad, India, 2011. (Cited on page 69.)

[51] Jerry R. Hobbs and Feng Pan. Time Ontology in OWL (September 2006). `http://www.w3.org/TR/owl-time`. (Cited on pages 32 and 74.)

[52] Thomas Hofmann. Probabilistic latent semantic indexing. In *22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57, Berkeley, CA, USA, 1999. (Cited on page 98.)

[53] Renato Iannella and James McKinney. vCard Ontology For describing People and Organisations (September 2013). `http://www.w3.org/TR/vcard-rdf`. (Cited on page 31.)

[54] Elena Ilina, Claudia Hauff, Ilknur Celik, Fabian Abel, and Geert-Jan Houben. Social event detection on twitter. In *12th International Conference on Web Engineering*, Berlin, Germany, 2012. (Cited on pages 51 and 145.)

[55] Paul Jaccard. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901. (Cited on page 125.)

[56] Ramesh Jain. Toward eventweb. *IEEE Distributed Systems Online*, 8, 2007. (Cited on page 13.)

[57] Vikramaditya R. Jakkula and Diane J. Cook. Learning temporal relations in smart home data. In *2nd International Conference on Technology and Aging*, Toronto, Canada, 2007. (Cited on page 118.)

[58] Matthew A. Jaro. Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *Journal of the American Statistical Association*, 84:414–420, 1989. (Cited on page 127.)

[59] A. Jentzsch, R. Isele, and C. Bizer. Silk - Generating RDF Links while publishing or consuming Linked Data. In *9th International Semantic Web Conference (ISWC'10)*, Shanghai, China, 2010. (Cited on pages 39 and 45.)

[60] Damir Juric, Laura Hollink, and Geert-Jan Houben. Discovering links between political debates and media. In *Proceedings of Web Engineering - 13th International Conference, ICWE 2013, Aalborg, Denmark, July 8-12, 2013*, pages 367 – 375. Springer LNCS 7977, 2013. (Cited on page 52.)

[61] Mehmet Kayaalp, Tansel Özyer, and Sibel Tariyan Özyer. A collaborative and content based event recommendation system integrated with data collection scrapers and services at a social networking site. In *International Conference on Advances in Social Networks Analysis and Mining*, Athens, Greece, 2009. (Cited on page 84.)

[62] James Kennedy and Russell C. Eberhart. Particle swarm optimization. In *the IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995. (Cited on pages 43, 130, 143, and 153.)

[63] Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. Aggregating Social Media for Enhancing Conference Experience. In *1st International Workshop on Real-Time Analysis and Mining of Social Streams (RAMSS'12)*, pages 34–37, Dublin, Ireland, 2012. (Cited on pages 5, 76, and 77.)

[64] Houda Khrouf, Ghislain Auguste Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. Confomaton: A conference enhancer with social media from the cloud. In *The Semantic Web: ESWC 2012 Satellite Events -, Heraklion, Crete, Greece, May 27-31, 2012*, pages 463–467, 2012. (Cited on page 5.)

[65] Houda Khrouf, Vuk Milicic, and Rapahël Troncy. Eventmedia live: Exploring events connections in real-time to enhance content. In *Semantic Web Challenge at 11th International Semantic Web Conference*, Boston, USA, 2012. (Cited on pages 5 and 73.)

[66] Houda Khrouf, Vuk Milicic, and Raphaël Troncy. Mining events connections on the social web: Real-time instance matching and data analysis in EventMedia. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 24:3–10, 2014. (Cited on page 5.)

[67] Houda Khrouf and Raphaël Troncy. EventMedia : visualizing events and associated media. In *Demo Session at the 10th International Semantic Web Conference*, Bonn, Germany, 2011. (Cited on page 5.)

[68] Houda Khrouf and Raphaël Troncy. Eventmedia live: Reconciliating events descriptions in the web of data. In *Proceedings of the 6th International Workshop on Ontology Matching*, pages 250–25, Bonn, Germany, 2011. (Cited on pages 5 and 40.)

[69] Houda Khrouf and Raphaël Troncy. Eventmedia: a LOD dataset of events illustrated with media. *Semantic Web Journal, Special Issue on Linked Dataset descriptions*, page 1570–0844, 2012. (Cited on pages 5 and 34.)

[70] Houda Khrouf and Raphaël Troncy. Hybrid event recommendation using linked data and user diversity. In *Proceedings of the 7th ACM Conference on Recommender Systems*, pages 185–192, Hong Kong, China, 2013. (Cited on pages 5 and 84.)

[71] Houda Khrouf and Raphaël Troncy. De la modélisation sémantique des événements vers l'enrichissement et la recommandation. *Revue d'Intelligence Artificielle*, 28(2-3):321–347, 2014. (Cited on page 5.)

[72] Joseph A. Konstan and John Riedl. Recommender systems: From algorithms to user experience. *User Modeling and User-Adapted Interaction*, 22, 2012. (Cited on pages 97 and 154.)

[73] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer Society*, 42, 2009. (Cited on pages 131 and 132.)

[74] Carl Lagoze and Jane Hunter. The abc ontology and model. In *International Conference on Dublin Core and Metadata Applications 2001*, Tokyo, Japan, 2001. (Cited on page 31.)

[75] T. K Landauer and S. Dumais. Latent semantic analysis. *Scholarpedia*, 3(11):4356, 2008. (Cited on page 103.)

[76] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998. (Cited on page 84.)

[77] Ora Lassila and Ralph R. Swick. Resource Description Framework (RDF) Model and Syntax Specification - W3C Recommendation (February 1999). http://www.w3.org/TR/1999/REC-rdf-syntax-19990222. (Cited on page 15.)

[78] Danielle Hyunsook Lee. Pittcult: trust-based cultural event recommender. In *ACM Conference on Recommender Systems (RecSys'08)*, Lausanne, Switzerland, 2008. (Cited on page 84.)

[79] WonSuk Lee, Werner Bailer, Tobias Bürger, Pierre-Antoine Champin, and Jean-Pierre Evain et al. Ontology for Media Resources 1.0 (February 2012). http://www.w3.org/TR/mediaont-10. (Cited on pages 33 and 140.)

[80] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. Statistical properties of community structure in large social and information networks. In *17th International Conference on World Wide Web*, WWW '08, pages 695–704, New York, NY, USA, 2008. (Cited on page 111.)

[81] Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10:707–710, 1966. (Cited on page 127.)

[82] Liyun Li and Nasir Memon. Mining groups of common interest: Discovering topical communities with network flows. In *9th International Conference on Machine Learning and Data Mining in Pattern Recognition*, Berlin, Heidelberg, 2013. (Cited on page 98.)

[83] Xiaoli Li, Aloysius Tan, Philip S. Yu, and See-Kiong Ng. Ecode: Event-based community detection from social networks. In *Database Systems for Advanced Applications*, Hong Kong, China, 2011. (Cited on pages 99, 103, and 157.)

[84] Zhiwei Li, Bin Wang, Mingjing Li, and Wei-Ying Ma. A probabilistic model for retrospective news event detection. In *28th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 106–113, Salvador, Brazil, 2005. (Cited on page 7.)

[85] Guoqiong Liao, Yuchen Zhao, Sihong Xie, and Philip S. Yu. An effective latent networks fusion based model for event recommendation in offline ephemeral social networks. In *22nd ACM International Conference on Information and Knowledge Management*, San Francisco, USA, 2013. (Cited on page 14.)

[86] X. Liu, R. Troncy, and B. Huet. Finding media illustrating events. In *1st ACM International Conference on Multimedia Retrieval*, Trento, ITALIE, 2011. (Cited on pages 2 and 14.)

[87] Xingjie Liu, Qi He, Yuanyuan Tian, Wang-Chien Lee, John McPherson, and Jiawei Han. Event-based social networks: Linking the online and offline social worlds. In

18th ACM SIGKDD conference on Knowledge Discovery and Data Mining, KDD'12, Beijing, China, 2012. (Cited on pages 91, 98, 99, 100, 101, and 154.)

[88] Xueliang Liu and Benoit Huet. Event representation and visualization from social media. In *14th Pacific-Rim Conference on Multimedia*, Nanjing, China, 2013. (Cited on page 51.)

[89] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, and Kai Li. Multi-probe lsh: Efficient indexing for high-dimensional similarity search. In *33rd International Conference on Very Large Data Bases*, Vienna, Austria, 2007. (Cited on page 103.)

[90] Silviu Maniu, Bogdan Cautis, and Talel Abdessalem. Building a signed network from interactions in wikipedia. In *Databases and Social Networks*, Athens, Greece, 2011. (Cited on page 119.)

[91] Michael Martin and Sören Auer. Categorisation of semantic web applications. In *4th International Conference on Advances in Semantic Processing*, Florence, Italy, 2010. (Cited on page 67.)

[92] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001. (Cited on page 102.)

[93] Pablo N. Mendes, Alexandre Passant, and Pavan Kapanipathi. Twarql: Tapping into the wisdom of the crowd. In *6th International Conference on Semantic Systems*, Graz, Austria, 2010. (Cited on page 52.)

[94] Peter Mika. *Social Networks and the Semantic Web*, volume 5 of *Semantic Web and Beyond*. Springer, 2007. (Cited on page 14.)

[95] Alistair Miles and Sean Bechhofer. SKOS Simple Knowledge Organization System Namespace Document (August 2009). `http://www.w3.org/2009/08/skos-reference/skos.html`. (Cited on page 32.)

[96] Einat Minkov, Ben Charrow, Jonathan Ledlie, Seth J. Teller, and Tommi Jaakkola. Collaborative future event recommendation. In *19th ACM Conference on Information and Knowledge Management*, Toronto, Ontario, Canada, 2010. (Cited on page 84.)

[97] A. Monge and C. Elkan. The eld-matching problem: algorithm and applications. In *2nd International Conference on Knowledge Discovery and Data Mining (KDD'96)*, Portland, Oregon, 1996. (Cited on page 128.)

[98] Óscar Muñoz-García and Raul Garcia-Castro. Guidelines for the specification and design of large-scale semantic applications. In *4th Annual Asian Semantic Web Conference*, Shanghai, China, 2009. (Cited on page 67.)

[99] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007. (Cited on pages 52 and 146.)

[100] Aparna Nagargadde, V. Sridhar, and Krithi Ramamritham. Representation and processing of information related to real world events. *Knowledge-Based Systems*, 20:1–16, 2007. (Cited on page 40.)

[101] Katsuko T. Nakahira, Masashi Matsui, and Yoshiki Mikami. The use of xml to express a historical knowledge base. In *16th International Conference on World Wide Web*, Banff, Alberta, Canada, 2007. (Cited on page 8.)

[102] Felix Naumann and Melanie Herschel. *An Introduction to Duplicate Detection*. Morgan and Claypool Publishers, 2010. (Cited on pages 2 and 38.)

[103] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, 2004. (Cited on pages 98, 108, 154, and 157.)

[104] M.E.J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133, September 2004. (Cited on pages 109 and 157.)

[105] M.E.J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006. (Cited on pages 98 and 154.)

[106] Maximilian Nickel and Volker Tresp. Tensor factorization for multi-relational learning. In *Machine Learning and Knowledge Discovery in Databases -European Conference*, Prague, Czech Republic, 2013. (Cited on page 119.)

[107] Andriy Nikolov, Mathieu d'Aquin, and Enrico Motta. Unsupervised learning of link discovery configuration. In *9th Extended Semantic Web Conference*, Heraklion, Crete, Greece, 2012. (Cited on pages 47 and 144.)

[108] Xing Niu, Shu Rong, Yunlong Zhang, and Haofen Wang. Zhishi.links results for oaei 2011. In *6th International Workshop on Ontology Matching*, Bonn, Germany, 2011. (Cited on pages 40 and 43.)

[109] Georgios Paliouras. Discovery of web user communities and their role in personalization. *User Modeling and User-Adapted Interaction*, pages 151–175, 2012. (Cited on pages 3, 97, 135, and 154.)

[110] Toon De Pessemier, Sam Coppens, Kristof Geebelen, Chris Vleugels, Stijn Bannier, Erik Mannens, Kris Vanhecke, and Luc Martens. Collaborative recommendations with content-based filters for cultural activities via a scalable event distribution platform. *Multimedia Tools Appl.*, 58(1):167–213, 2012. (Cited on page 95.)

[111] Daniele Quercia, Neal Lathia, Francesco Calabrese, Giusy Di Lorenzo, and Jon Crowcroft. Recommending social events from mobile phone location data. In *10th IEEE International Conference on Data Mining*, Sydney, Australia, 2010. (Cited on page 88.)

[112] Willard V. Quine. Events and reification. In *Actions and Events: Perspectives on the Philosophy of Davidson*, pages 162–71. Blackwell, 1985. (Cited on page 40.)

[113] Y. Raimond, S. Abdallah, M. Sandler, and F. Giasson. The Music Ontology. In *8th International Conference on Music Information Retrieval (ISMIR'07)*, Vienna, Austria, 2007. (Cited on page 31.)

[114] Tye Rattenbury, Nathaniel Good, and Mor Naaman. Towards automatic extraction of event and place semantics from flickr tags. In *30th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 103–110, Amsterdam, The Netherlands, 2007. (Cited on page 7.)

[115] Dave Reynolds, Jeni Tennison, and Leigh Dodds. Resource Description Framework (RDF) Model and Syntax Specification - W3C Recommendation (February 1999). `https://code.google.com/p/linked-data-api/wiki/Specification`. (Cited on page 69.)

[116] Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. What fresh media are you looking for?: Retrieving media items from multiple social networks. In *International Workshop on Socially-aware Multimedia*, Nara, Japan, 2012. (Cited on page 24.)

[117] Giuseppe Rizzo, Marieke van Erp, and Raphaël Troncy. Benchmarking the extraction and disambiguation of named entities on the semantic web. In *9th International Conference on Language Resources and Evaluation*, Reykjavik, ICELAND, 2014. (Cited on pages 55, 56, 146, and 147.)

[118] S. Robertson. Understanding inverse document frequency: On theoretical arguments for idf. *Journal of Documentation*, 60(5):503–520, 2004. (Cited on page 150.)

[119] Ekkawut Rojsattarat and Nuanwan Soonthornphisaj. Hybrid Recommendation: Combining Content-Based Prediction and Collaborative Filtering. In *Intelligent Data Engineering and Automated Learning*, volume 2690, pages 337–344. Springer Berlin Heidelberg, 2003. (Cited on page 95.)

[120] Peter Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, 20(1):53–65, 1987. (Cited on page 113.)

[121] Matthew Rowe and Milan Stankovic. Aligning Tweets with Events: Automation via Semantics. *Semantic Web Journal*, 3(2):115–130, 2012. (Cited on pages 2, 51, 52, and 145.)

[122] Mrinmaya Sachan, Danish Contractor, Tanveer A. Faruquie, and L. Venkata Subramaniam. Using content and interactions for discovering communities in social networks. In *21st World Wide Web Conference*, Lyon, France, 2012. (Cited on page 98.)

[123] Shaghayegh Sahebi and William Cohen. Community-based recommendations: a solution to the cold start problem. In *Workshop on Recommender Systems and the Social Web, held in conjunction with ACM RecSys'11*, Chicago, USA, 2011. (Cited on page 119.)

[124] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *19th International Conference on World Wide Web*, Raleigh, North Carolina, USA, 2010. (Cited on pages 2, 13, 51, and 145.)

[125] Gerard Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of The ACM*, 18:613–620, November 1975. (Cited on page 85.)

[126] Robert Scholes. Language, narrative, and anti-narrative. *Critical Inquiry*, 7(1):pp. 204–212, 1980. (Cited on page 7.)

[127] Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The semantic web revisited. *IEEE Intelligent Systems*, 21(3), 2006. (Cited on page 30.)

[128] Upendra Shardanand and Pattie Maes. Social information filtering: Algorithms for automating “word of mouth”. In *SIGCHI Conference on Human Factors in Computing Systems*, Denver, Colorado, USA, 1995. (Cited on page 97.)

[129] R. Shaw, R. Troncy, and L. Hardman. LODE: Linking Open Descriptions Of Events. In *4th Asian Semantic Web Conference (ASWC'09)*, pages 153–167, Shanghai, China, 2009. (Cited on pages 8, 31, 74, and 138.)

[130] David A. Smith. Detecting and browsing events in unstructured text. In *25th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 73–80, Tampere, Finland, 2002. (Cited on page 7.)

[131] Dezhao Song and Jeff Heflin. Automatically generating data linkages using a domain-independent candidate selection approach. In *10th International Semantic Web Conference (ISWC'11)*, Bonn, Germany, 2011. (Cited on pages 40, 43, and 150.)

[132] Karen Sparck Jones and Peter Willett, editors. *Readings in Information Retrieval*. Morgan Kaufmann Publishers Inc., 1997. (Cited on page 43.)

[133] Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas L. Griffiths. Probabilistic author-topic models for information discovery. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 306–315, 2004. (Cited on page 98.)

[134] York Sure, Stephan Bloehdorn, Peter Haase, Jens Hartmann, and Daniel Oberle. The swrc ontology - semantic web for research communities. In *12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence*, Covilha, Portugal, 2005. (Cited on page 56.)

[135] Martin Szomszor, Terry R. Payne, and Luc Moreau. Using semantic web technology to automate data integration in grid and web service architectures. In *5th International Symposium on Cluster Computing and the Grid*, Cardiff, UK, 2005. (Cited on page 30.)

[136] Raphaël Troncy, André T. S. Fialho, Lynda Hardman, and Carsten Saathoff. Experiencing events through user-generated media. In *1st International Workshop on Consuming Linked Data*, Shanghai, China, 2010. (Cited on page 1.)

[137] Alexey Tsymbal. The problem of concept drift: Definitions and related work. Technical Report TCD-CS-2004-15, The University of Dublin, Trinity College, Ireland, 2004. (Cited on page 119.)

[138] Zhu Wang, Xingshe Zhou, Daqing Zhang, Dingqi Yang, and Zhiyong Yu. Cross-domain community detection in heterogeneous social networks. *Personal and Ubiquitous Computing*, 18(2):369–383, 2014. (Cited on pages 99, 109, and 157.)

[139] Melanie Weis and Felix Naumann. Dogmatix tracks down duplicates in xml. In *ACM SIGMOD International Conference on Management of Data*, Baltimore, Maryland, 2005. (Cited on pages 41 and 127.)

[140] Jianshu Weng and Bu-Sung Lee. Event detection in twitter. In *5th International Conference on Weblogs and Social Media*, Barcelona, Spain, 2011. (Cited on pages 13, 51, and 145.)

[141] William E. Winkler. The state of record linkage and current research problems. Technical report, Statistical Research Division, U.S. Census Bureau, 1999. (Cited on page 127.)

[142] Hao Wu, Vikram Sorathia, and Viktor Prasanna. When diversity meets speciality: Friend recommendation in online social networks. *ASE Human Journal*, 1:52–60, 2012. (Cited on pages 81 and 90.)

[143] Jen yuan Yeh, Jung yi Lin, Hao ren Ke, and Wei pang Yang. Learning to rank for information retrieval using genetic programming. In *SIGIR Workshop on Learning to rank for Information Retrieval*, 2007. (Cited on pages 129 and 153.)

[144] J.M Zacks, T.S Braver, M.A. Sheridan, D.I Donaldson, A.Z Snyder, J.M. Ollinger, RL Buckner, and M.E Raichle. Human brain activity time-locked to perceptual event boundaries. *Nature Neuroscience*, 4:2126–2431, 2001. (Cited on page 1.)

[145] Kuo Zhang, Juan Zi, and Li Gang Wu. New event detection based on indexing-tree and named entity. In *30th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 215–222, Amsterdam, The Netherlands, 2007. (Cited on page 7.)

[146] Zhongying Zhao, Shengzhong Feng, Qiang Wang, Joshua Zhexue Huang, Graham J. Williams, and Jianping Fan. Topic oriented community detection through social objects and link analysis in social networks. *Knowledge-Based Systems*, 26:164–173, 2012. (Cited on pages 4, 98, 108, and 157.)

[147] Ding Zhou, Eren Manavoglu, Jia Li, C. Lee Giles, and Hongyuan Zha. Probabilistic models for discovering e-communities. In *15th International Conference on World Wide Web*, pages 173–182, New York, NY, USA, 2006. (Cited on page 98.)