Leveraging Cross-Domain Social Media Analytics to Understand TV Topics Popularity

Pensa, Ruggero G.; Sapino, Maria Luisa; Schifanella, Claudio; Vignaroli, Luca. Leveraging Cross-Domain Social Media Analytics to Understand TV Topics Popularity. IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE. 11 (3) pp: 10-21.
DOI: 10.1109/MCI.2016.2572518

Leveraging Cross-domain Social Media Analytics to Understand TV Topics Popularity

Ruggero G. Pensa, *Department of Computer Science, University of Torino, Torino, Italy*
Maria Luisa Sapino, *Department of Computer Science, University of Torino, Torino, Italy*
Claudio Schifanella, *RAI-CRIT, Torino, Italy*
Luca Vignaroli, *RAI-CRIT, Torino, Italy*

Abstract

The way we watch television is changing with the introduction of attractive Web activities that move users away from TV to other media. The social multimedia and user-generated contents are dramatically changing all phases of the value chain of contents (production, distribution and consumption). We propose a concept-level integration framework in which users' activities on different social media are collectively represented, and possibly enriched with external knowledge, such as information extracted from the Electronic Program Guides, or available ontological domain knowledge. The integration framework has a knowledge graph as its core data model. It keeps track of active users, the television events they talk about, the concepts they mention in their activities, as well as different relationships existing among them. Temporal relationships are also captured to enable temporal analysis of the observed activity. The data model allows different types of analysis and the definition of global metrics in which the activity on different media concurs with the measure of success.

I. Introduction

The introduction of attractive Web activities that move users away from television to other media is changing the way we watch TV. Also the media broadcasting models are changing in order to cover the TV-Web convergence. VoD (Video on Demand) and EPGs (Electronic Program Guides) provided by broadcasters are examples of services that allow new forms of user navigation within the television content. At the same time, the popularity of online social networks has changed the Internet ecosystem, thus leading to more collaborative environments, reflecting the structure and dynamics of the society. The social multimedia and user-generated contents are dramatically changing all phases of the value chain of contents (production, distribution and consumption).

For the broadcasters and advertisers, social TV means much deeper real-time understanding of what viewers think about shows and the brands that advertise on them. Consequently, it provides them with data-driven understanding of their investments in contents. This will be the most significant change that social TV brings to the TV business. Both the commissioning and scheduling of TV contents and the pricing of the spot ads and program sponsorship are based on the way the TV audience is measured. There is no doubt that traditional TV ratings still rank as an

Corresponding Author: Ruggero G. Pensa (Email: ruggero.pensa@unito.it)

important measurement that advertisers pay attention to when buying advertising inventory, however, it is possible that in the upcoming years social TV data will shift attention away from traditional audience ratings.

Traditionally, measures about people's habits and reactions are gathered in two ways: firstly, by viewing habits of panels of TV viewers and parsing the results of network surveys on the opinion (e.g. the Nielsen ratings); secondly, by generating traditional live broadcast audience figures with the so-called set meters (small devices connected to TVs in a small number of selected homes) on a daily basis. However, this approach misses the explosive growth and increasing diversity of comments and opinions in real time from an expanding number of online social platforms. In the typical offline scenario, audience profiles are obtained manually by gathering a set of predefined socio-demographics characteristics obtained from a statistically significant sample of the possible consumers. By contrast, in an online scenario, audience profiles and impact of TV programs might be obtained by tracking social media sites (e.g. Twitter, Facebook) and applying Natural Language Processing (NLP) technologies and data mining techniques on their contents. This might enable TV reprogramming and media planning strategies, such as contextual advertisement or behavioral targeting.

In this paper, we observe that, although different social media are characterized by different users' activity styles, they all carry useful information (i.e., non-redundant with respect to each other). While Twitter activities have the peculiarity of being very timely and immediate — usually users tweet in real time, while watching the program they are commenting about — a good portion of the activities on YouTube and Facebook happens with some time shift with respect to the on-air show. Users post fragments of videos, which potentially trigger comments and discussions for days, in some cases even weeks or months. Thus, we propose a concept-level integration framework in which users' activities on different social media are collectively represented by means of conceptual abstractions, possibly enriched with external knowledge, such as information extracted from the EPGs, or available ontological domain knowledge.

The framework has a knowledge graph as its core data model, which keeps track of active users, the television events they talk about, the concepts they mention in their activities, as well as different relationships existing among them, including temporal relationships which enable temporal analysis of the observed activity. The data model allows different types of analysis and the definition of global metrics in which the activity on different media concurs with the measure of success.

Note that, although we concentrate our cross-media analysis on the study of the popularity of topics tackled in television programs, we do believe that the concept-level integration platform has the property of being very general. As such, it has the potential of being populated and enriched with information of interest in different domains (such as tracking political dynamics, tracking the correlation between users' social activities and economic patterns). Thus, in the rest of the paper, we first define the general concept, and then show how they are instantiated in the TV domain.

The paper is organized as follows: after presenting a survey of related literature in Section II, we introduce our integration framework in Section III. We formally define the graph integration model in Section IV, and describe the source processing steps which extract the concepts and relationships that will populate the graph in Section V. In Section VI, we define formally some queries of interest. Section VII shows the potential of the graph as a data

source to analyze topics' popularity. Finally, we draw the conclusion in Section VIII.

II. Related work

In this section, we will provide an overview of relevant related work. As the main focus of this paper is on cross network analysis to capture the interest users show towards topics addressed in TV programs, we will first survey literature on cross-network analysis. Then we will discuss graph based multi-source data integration, which relates to our paper in that our analysis relies on a graph data model to represent multi-source social media information.

Cross-network analysis: Social network analysis has recently been a core method to understand various phenomena potentially influenced by the exchange of users' opinions. Retrieving information from social media is then a crucial preliminary task. At this purpose, in [1], the authors investigate how to automatically retrieve a context-relevant social network content without user intervention, by considering both the participatory and implicit-topical properties of the context to improve the retrieval performance. Users' activities on Twitter are used to support the prediction of economic phenomena, such as stock prices [2], and for tracking online social movements [3]. Kaschesky *et al.* [4] use sentiment analysis to predict the political orientation of a person (Republican vs. Democrat) or agreement/disagreement on political issues. Alashri *et al.* [5] analyze online ideological political debates, defined as a formal discussion on a set of related issues in which opposing perspectives and arguments are put forward.

Recently TV broadcasters recognized that users' activities on social media are valuable sources of information about their interests towards TV programs. In 2012, Nielsen — a global information and measurement company — and Twitter agreed to create the "Nielsen Twitter TV Rating" for the US market. The main goal of their agreement is the definition of a metric relying on conversations about TV programs on Twitter to measure users' interests. The metric provides valuable information for TV contents recommendation, including personalized commercial campaigns. In the context of TV and social Web integration, Bluefin Labs[1] releases a suite of analytics tools to explore the social content related to Social TV programs and to analyze the data generated by the "TV Genome", i.e., the mapping between social media and TV media. This software is based in large part on researches on natural language processing, speech-to-text and video-entity recognition carried out by the two co-founders [6], [7].

Personalization and recommendation in the TV domain: the study [8] introduces a linear time algorithm to solve the problem that involves selecting different program slots telecast on different television channels in a day so as to reach the maximum number of viewers. O'Sullivan *et al.* [9] address the problem of creating personalized EPGs in the digital TV domain by applying data mining methods to extract new program metadata from user profiles. Yan *et al.* [10] propose a YouTube video recommendation solution via cross-network collaboration: the authors concentrate on those users who are active both on Twitter and on YouTube, and exploit the users' profile information that they can learn by analyzing their activity on Twitter to personalize YouTube video recommendations. While being similar to our approach for the basic idea of integrating information coming from different social media, the work presented in [10] significantly differs from the cross-network concept-level integration proposed in this paper.

Graph based knowledge representation, integration and querying: The necessity of structuring knowledge in a graph was already identified in 1988 by [11] as a means of representing knowledge from multiple sources

[1] http://bluefinlabs.com/

in knowledge-based systems. Ten years after, this necessity has been translated into the design of information storage and retrieval systems such as the one presented by [12]. Today, knowledge graphs are exploited by semantic analysis [13], sentiment analysis [14] and opinion mining [15]. Furthermore, time is also a key question in knowledge representation and analysis [16]. As an example, Google uses a knowledge graph for its search engine.

In more recent years, many researchers have focused their efforts in identifying a way to represent heterogeneous, multimedia and multi-language ontological knowledge embracing a wide range of domains. For instance, the studies [17], [18] introduce Freebase, a tuple database used to structure general human knowledge. Navigli and Ponzetto [19] present an automatic approach to the construction of BabelNet, a very large, wide-coverage multilingual semantic network by integrating lexicographic and encyclopedic knowledge from WordNet and Wikipedia.

Querying and analyzing these knowledge graphs is a key issue in heterogeneous knowledge-based systems. The paper [20] presents an abstract machine dedicated to querying knowledge graphs as the result of an abstraction process performed to reach a generic solution to the problem of querying graphs in various models. The authors of [21] present a web-based system for visual and interactive analysis of large sets of documents using statistical topic models. This work proposes a range of visualization types and control mechanisms to support knowledge discovery, including corpus and document specific views, iterative topic modeling, search, and visual filtering.

As graphs become increasingly large, scalability quickly becomes the major research challenge for the reachability computation today. Many works propose different indices to answer reachability queries efficiently [22], [23], [24]. Jin *et al.* [25] propose a unified reachability computation framework scaling reachability indices to help speed up the online query answering approaches. When knowledge graphs become huge, the relevance of the returned results is a key issue at least as the response time. The study [26] addresses the problem of an index structure through the design and implementation of a concept-based model using domain-dependent ontologies.

Our work is transverse to the presented related researches. Not only do we provide a theoretical framework for concept-level heterogenous and time-evolving data integration, management and querying, but we also develop a web-based application guiding the user in the exploration, analysis and visualization of the complex and dynamic interactions constituting the “extended life” of TV events.

A preliminary version of this work has been published in [27], [28]. This paper significantly extends our former publications by adding many previously missing technical details. In particular, we now provide the formal definitions of the domain-specific model as well as the theoretical foundations of our querying framework. Finally, in this work we report the results of an experiment conducted on a more recent and large-scale scenario.

III. The integration framework

Our cross-network analysis framework was first introduced in [27] and consists of three main layers, covering all the phases from data collection, representation and integration to data analysis. More specifically, a **source processing layer** contains the different modules for collecting all the data to be conveyed in the knowledge representation model. It accesses a number of predefined web/social/media sources (e.g., broadcasters official web sites, social networks, TV channels, ontological information sources) and extracts from them those information units which denote relevant concepts (e.g., people names, geographical names, temporal information, topic names,

etc.) as well as information supporting the existence of relationships (which will be modeled as edges in the graph) among them.

The collected concepts and relationships among them will be organized in a structured knowledge graph by the **knowledge graph layer**, which contains all the modules needed to define and store the knowledge graph.

The **knowledge query and analysis layer** offers functionalities for querying, browsing and analyzing the knowledge graph. More specifically, a query module extracts subgraphs from the knowledge graph based on user's requirements and constraints. Each extracted subgraph can be seen as a "view" over the complete knowledge graph, only containing nodes and edges potentially relevant to the user query. An analysis module provides a set of analysis and data mining components to extract models and patterns from the knowledge graph. Both the entire knowledge graph, and the individual views (subgraphs extracted from the query module) can be subject to data analysis. Tensor based representations [29] are also provided, to enable the direct application of existing matrix and tensor based analysis libraries, as well as the definition of innovative analysis algorithms efficiently dealing with the multidimensional characteristics of the modeled knowledge.

Notice that in our integration framework a fundamental role is played by a *semantic engine*. First, it is adopted in the source processing layer to provide an interpretation to web/social/media elements extracted from the heterogeneous sources. In the source processing phase, the semantic engine helps understand whether the considered entities should be modeled as concepts or relationships among existing concepts, and helps provide a suitable set of features based on their characteristics. Second, the semantic engine plays an important role in the graph query and analysis layer, where it assigns a semantic role to each selected node/edge.

In the following sections, we describe the three phases of the frameworks in details. We first present in details the definition of the knowledge graph, which is the core of our proposal. Then we discuss how the data sources are processed to extract the relevant information and populate the knowledge graph. Finally, we define the formal query and analysis framework.

IV. Modeling cross-network knowledge

The core of our framework is the knowledge base that represents the result of public actions of users in social environments [27], [28]. Combining different theories from cognitive science [30], [31], language philosophy [32] and social ontology [33], we recognize three classes of entities, which will be mapped into three types of nodes in the knowledge graph: *subjects*, i.e., users who take public actions (such as posting a tweet), *social objects*, i.e., the result of public acts (such as a set of tweets posted by a user), and *concepts*, physical and/or ideal objects mentioned by subjects via their public actions. Any act (or set of acts) that can be identified by its trace, and has a recognized social value is a social object. Given the size of the domain of interest, and the granularity of the analysis we are interested in, in this paper we choose to model social objects to represent groups of similar actions instead of keeping track of the individual subjects' actions. This assumption could be relaxed if we were interested in distinguishing every single users' action (for example, to work towards personalized recommendation systems).

We capture different existing relationships between subjects and social objects, and between social objects and concepts: a group of subjects that recognize a social value of an act *supports* the resulting social object (e.g. the

contractors *support* the contract); a social object *represents* a social instance of some concepts on a precise context (e.g. a video may represent a volleyball match). Other relationships exist among entities of the same type. We call these relationships *structural dependencies*. A social object o_1 is *structurally dependent* on another object o_2 if o_1 is a part of o_2 (e.g. a comment is a part of a video). A subject can be *structurally dependent* on a group of subjects (e.g. a subscriber is a part of playlist subscribers) that performed the same kind of actions on the same social object. A concept may be *structurally dependent* on a more general concept (e.g. hilarity is a specialization of joy).

Finally, we capture the fact that social objects evolve with time. Hence, as a special case of representation relationship, we consider the *temporal relation* between a social object and a *temporal concept* (e.g. a video has been posted in a specific time instant, and has been viewed during a specific time period).

Based on the above, in the following subsection, we formally define the knowledge graph.

A. Knowledge Graph

The knowledge graph (first introduced in [27]) models all the relationships between social objects, subjects and concepts introduced so far.

Definition 1 (Knowledge Graph): Let $\mathcal{O}$, $\mathcal{S}$ and $\mathcal{C}$ be the sets of all social objects, subjects and concepts, respectively. Let $\mathcal{T} \subseteq \mathcal{C}$ be the set of temporal concepts. The cross-network knowledge graph on $\mathcal{O}$, $\mathcal{S}$, and $\mathcal{C}$, is the directed weighted graph $G^K(V, E, W)$, where the set of vertices is $V = \mathcal{O} \cup \mathcal{S} \cup \mathcal{C}$, the set of edges is $E = E^{sup} \cup E^{rep} \cup E^{str}$ including edges representing support relationships, representation relationships as well as structural dependency relationships. In particular $E^{sup} = \{(s_i, o_j)\ s.t.\ s_i \in \mathcal{S},\ o_j \in \mathcal{O}\}$ is the set of *support* edges; $E^{rep} = \{(o_i, c_j)\ s.t.\ o_i \in \mathcal{O},\ c_j \in \mathcal{C}\}$ is the set of *representation* edges, and $E^{str} = \{(v_i, v_j)\ s.t.\ v_i, v_j \in \mathcal{S} \vee v_i, v_j \in \mathcal{O} \vee v_i, v_j \in \mathcal{C},\ i \neq j\}$ is the set of *structural dependency* edges. The edge weighting function is $W : E \to (0, 1]$.

Each node $v \in V$ has three attributes: $v.label$, $v.subtype$ and $v.magnitude$, representing the name of the concept associated to the node, an application-specific type and the number of instances of such concept recognized in the data sources, respectively. Optionally, each edge $e \in E$ may be characterized by an attribute $e.subtype$ which specifies an application-specific type. Moreover, given the set of time concepts $\mathcal{T} \subseteq \mathcal{C}$, $E^{tmp} \subseteq E^{rep}$ denotes the set of edges (o_i, t_j), where $o_i \in \mathcal{O}$, and $o_j \in \mathcal{T}$.

A special subgraph of G^K is the ontology graph. Vertices in the knowledge base are all concepts belonging to $\mathcal{C}$, defined as follows.

Definition 2 (Ontology Graph): The ontology graph $G^O(V^O, E^O, W^O)$ is the subgraph of G^K induced by $V^O = \mathcal{C}$.

Thus, E^O is a set of *structural dependency* edges encoding several ontology relationships (such as: “is a”, “part of”), possibly specified by the attribute $e.subtype$.

Fig. 1 (left side) shows a small example of knowledge graph in which every type of node and edge is represented.

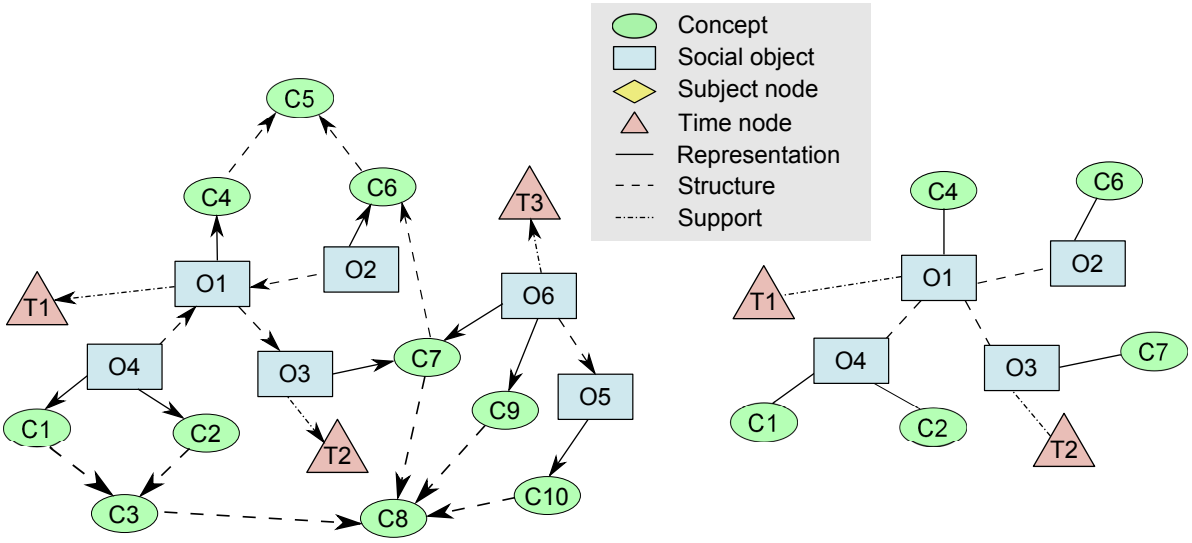


Fig. 1. An example of a knowledge graph representation with all types of nodes and edges (left) and the social context of node $C1$ (right).

V. Extracting information from social media data sources

In this section, we describe how the user-generated content publicly available on social media is processed and mapped to the concepts and relationships to be represented in the knowledge graph defined in Section IV.

A. Facebook content

Facebook is the most famous and widespread social networking platform. Since it has about 1.5 billions of monthly active users, TV companies use it for stimulating discussions around TV shows. Through the website, users may post, watch, comment, like or dislike any kind of multimedia comments: text comments, video, picture, news articles and so on. Usually, Facebook posts have a short life, but some posts can be commented by users even many days (or even months) after their first publication. We then associate each Facebook post to a plurality of *commentset* nodes defined as:

Definition 3 (commentset): Given a time interval $\Delta T_j = (t^j_{start}, t^j_{end})$ and a Facebook post f, a commentset $CS(\Delta T_j, f)$ is a collection of comments posted during ΔT_j by users as a reaction to post f.

We store posts that are closely related to a TV event, e.g., because they are published by the broadcaster's official channel/user or by other users mentioning the TV event in the title or in the description. Here, we define as *TV event* an individual episode of a television series (e.g., news programs, fiction series, magazines) or a single production (e.g., a football match, a film, a live concert).

In particular, for each Facebook post f we store: i) a social object node o_i representing the Facebook post; ii) a representation edge e^{rep}_{ie} linking o_i to c_e (the node corresponding to the target TV event e); iii) a set N of social object nodes o_j representing the N commentsets $CS(\Delta T_j, f)$, $j = 1, \ldots, N$; iv) the attributes $o_j.magnitude$ that represent the number of comments for f in $CS(\Delta T_j, f)$; v) two time nodes t_c and t_v for each commentset node o_j, with $t_c.label = t^j_{start}$ and $t_v.label = t^j_{end}$; vi) two edges e^{tmp}_{jc} and e^{tmp}_{jv} connecting each o_j to t_c and t_v.

Instead of storing the full textual content of the comments associated to the commentset, we only consider the most relevant concepts referred by the text and belonging to the following categories: *people*, *places*, *events*, *emotions*, *polarities*. We refer to these concepts as *named entities*. In particular, for each category, we store: i) a social object node o_c representing the category; ii) a concept node c_j for each relevant concept referred by $CS(\Delta T, f)$ and related to the category of o_c; iii) an edge e_{jc}^{rep} connecting each concept node c_j to o_c, and a weight w_{jc} that expresses the relative importance of the concept c_j (note that $\sum_j w_{cj} = 1$); iv) a structure edge e_{ci}^{str} that connects o_c to o_i, and a weight w_{ci} expressing the relative importance of the category o_c w.r.t. the other categories (note that $\sum_c w_{ci} = 1$); v) the attribute $o_c.magnitude$ that expresses the total occurrences of all concepts connected to o_c.

Notice that, in this work, we refer to the word *concept* both as an abstraction (e.g., person, tv show, event) and as an instantiation of an abstract concept (e.g., "Barak Obama", "Late Show", "Rio 2016 Olympic Games"). This choice is justified by the necessity of considering the evolution of users' perception of persons, events, places in time. According to our assumption, for instance, "Arnold Schwarzenegger" may be considered as a general concept that can be associated to his career as an actor or as a politician.

People who have contributed to the creation of social objects are represented as well. As for concepts, we include in the model only the most representative users belonging to the following categories: *viewers*, *active users*, *uploaders*, and *broadcasters*. In details, for each user category, we store: i) a subject node s_c representing the user category; ii) a subject node s_j for each relevant user extracted from $CS(\Delta T, f)$ and related to the category referred by s_c; iii) a structure edge e_{jc}^{str} connecting each subject node s_j to s_c, and a weight w_{jc} that expresses the relative importance of the user v_j^S in the considered category (note that $\sum_j w_{jc} = 1$); iv) a support edge e_{ci}^{sup} that connects s_c to o_i, and a weight w_{ci} expressing the relative importance of the category s_c w.r.t. the other categories (note that $\sum_c w_{ci} = 1$); v) the attribute $s_c.magnitude$ that expresses the total occurrences of all concepts connected to s_c.

It is worth noting that YouTube videos can be processed as Facebook posts.

B. Twitter content

Twitter is certainly one of the most dynamic social networking platforms due to its well-known peculiarities (among others, Twitter posts — called tweets — are limited to 140 characters and can be sent from any mobile device). Any kind of live events is often followed by thousands of tweets, thus providing a huge source of information for the analysts. TV programs' editorial boards are used to propose specific topics of discussions by using hashtags triggering huge amounts of new tweets. Consequently, Twitter is often adopted as the preferred means to let the audience express instant feelings and opinions about what is being broadcasted. As such, it is a key source of information for our application. However, unlike Facebook that has a clear social unit (the post) corresponding to a knowledge graph node, a social node in Twitter is harder to identify. We may think about creating a node for each tweet, but it has two main drawbacks: (i) it is often semantically poor and (ii) the knowledge graph is subject to a rapid explosion. Another possibility is to identify a prolific Twitter user as a central social node. However, this choice is questionable too: a Twitter user is similar to a Facebook user. It provides subjective representations of the reality. To cope with this issue, we define a new social entity called *tweetset*, defined as follows:

Definition 4 (tweetset): Given a time interval $\Delta T = (t_{start}, t_{end})$ and a TV event e, a tweetset $TS(\Delta T, e)$ is a collection of tweets posted during ΔT and closely related to e.

For instance, the retained tweets are those mentioning hashtags and posted by users associated to a specific TV event (e.g., the official Twitter users/hashtags). As such, the portion of knowledge graph corresponding to a tweetset $TS(\Delta T, e)$ is given by: i) a social object node o_i representing the tweetset; ii) a representation edge e_{ie}^{rep} connecting node o_i to node c_e corresponding to the TV event e; iii) the node attribute $o_i.magnitude$, initialized with the number of tweets in $TS(\Delta T, e)$; iv) two time nodes t_c and t_v with $t_c.label = t_{start}$ and $t_v.label = t_{end}$; v) two edges e_{ic}^{tmp} and e_{iv}^{tmp} connecting o_i to t_c and t_v.

Finally, we consider the concepts referred by the text in the tweetsets and the most relevant users. Both concepts and users are stored in the same way as described in Section V-A.

C. Enriching social media information with knowledge from the EPG content

The official EPG is often provided as a static content by broadcasters themselves. Although this makes us think about EPGs as non-social content, in our application this assumption is false. In fact, EPGs may be enriched by information coming from the social platform that provides users' rating, reviews, descriptions, and so on. Including these social sources of information is required when the official EPG content is poor. We consider a central social node corresponding to the TV event, the unit of an EPG schedule. This node is connected on one side to concepts identifying persons, places and events referred by each TV event, on the other side to concepts related to the TV event itself and its related TV program. In detail, the portion of the knowledge graph related to a TV event broadcasted from time t_{start} to time t_{end}, is structured as follows: i) a social object node o_i representing the TV event; ii) a representation edge e_{ie}^{rep} connecting node o_i to node c_e, the conceptual node corresponding to the TV event; iii) two time nodes t_c and t_v with $t_c.label = t_{start}$ and $t_v.label = t_{end}$; iv) two edges e_{ic}^{tmp} and e_{iv}^{tmp} connecting o_i to t_c and t_v.

Finally, we consider some concepts referred by EPG sources in the following categories: *people*, *places*, *events*, *genre* and *tv channel*. We refer to Section V-A for a detailed description of how these concepts are stored.

D. Enriching social media information with domain ontologies

In addition to the social part of the knowledge graph, we consider other sources of knowledge that form the subgraph G^O of G^K. In particular, we import ontology nodes from DBPedia[2] (for general purpose concepts nodes) and a simplified version of WordNet-Affect[3] (for sentiment/emotion concept nodes). Moreover, we enrich the EPG with conceptual nodes related to the TV events. In particular, we link all TV event concept nodes to a TV program node (for instance, we may link an hypothetical node concerning "Dexter, Season 4, Episode 12" to another concept node related to "Dexter, Season 4" in its turn linked to the more general concept of "Dexter (TV series)".

[2]http://dbpedia.org/

[3]http://wndomains.fbk.eu/wnaffect.html

VI. Querying the Knowledge Graph

The knowledge graph described in the previous section can potentially represent huge amounts of rich, heterogenous and time-evolving information. Accessing and querying the graph in a simple but efficient way are then crucial for the usability of the system. The result-set of each query can be processed for visualization and analysis purposes. To this end, we must define a simpler model to represent the result-set of a query on the Knowledge Graph G^K. In particular, the result-set of a query is modeled as an undirected weighted graph $G^Q = (V^Q, E^Q, W^Q)$, where V^Q is a set of vertices; $E^Q = \{(v_i, v_j)\ s.t.\ v_i, v_j \in V^Q\}$ is a set of undirected edges; $W^Q : V^Q \times V^Q \rightarrow \mathcal{R}$ is the function that associates a weight w_{ij} to each edge $(v_i, v_j) \in E^Q$.

As an extension, the result-set may involve multiple query graphs. Here, we consider a more general model consisting of a collection of N query graphs $\mathcal{G}^Q = \{G_1^Q, \ldots, G_N^Q\}$.

We can now define the general form of a graph query:

Definition 5 (graph query): Given a knowledge graph G^K, a query $Q^K(G^K, \mathcal{P}, \mathcal{F})$ returns a collection of query graphs $\mathcal{G}^Q = \mathcal{F}(G')$, where: $\mathcal{F}$ is a mapping function $\mathcal{F}_{G^K} : G^K \rightarrow \mathcal{G}^Q$ that associates vertices, edges and weights in G^K to vertices, edges and weights in $\mathcal{G}^Q$; $\mathcal{P}$ is a selection predicate on G^K, i.e., a function $\mathcal{P}_{G^K} : G^K \rightarrow \{true, false\}$; $G' \subseteq G^K$ is the subgraph of G^K satisfying $\mathcal{P}$.

This general definition embraces potentially any kind of selection query. However, in our system, we focus on a specific type of query called *similarity query*. The goal of this query is to provide a graph where two vertices are connected if they are similar enough. The weight of the edge connecting them measures the strength of their similarity. Before providing the definition of similarity query, we briefly introduce the definitions of social context of a knowledge graph node. In the following, the set of all social sources is denoted by $\mathcal{KS}$.

Definition 6 (social context): Given a knowledge graph $G^K = (V, E, W)$ and a node $v_i \in V$, a time interval $\Delta T = (t_{start}, t_{end})$ and a set of social sources $KS \subseteq \mathcal{KS}$, the social context of v_i in KS during ΔT is given by the undirected graph $G^{sc}_{(v_i,\, G^K,\, KS,\, \Delta T)}$ built on the subgraph of G^K induced by the nodes in $V^O_{(v_i,\, G^K,\, KS,\, \Delta T)} \in \mathcal{O}$ and $V^C_{(v_i,\, G^K,\, KS,\, \Delta T)} \in \mathcal{C}$ where: $V^O_{(v_i,\, G^K,\, KS,\, \Delta T)}$ is the set of the nodes $o_j \in \mathcal{O}$ such that (i) $\exists (o_j, t_c) \in E^{tmp}$ s.t. $t_{start} \leq t_c.label \leq t_{end}$, (ii) $o_j.source \in KS$, (iii) there is a path

$$P_{ij} = ((v_i, v_{i+1}), (v_{i+1}, v_{i+2}), \ldots, (v_{i+n-1}, v_{i+n}))$$

where $v_{i+n} = o_j$ and $\forall k = 0 \ldots n$, $(v_{i+k}, v_{i+k+1}) \in E^{str} \vee (v_{i+k+1}, v_{i+k}) \in E^{rep}$; $V^C_{(v_i,\, G^K,\, KS,\, \Delta T)}$ is the set of nodes $c_k \in \mathcal{C}$ such that $\exists (o_j, c_k) \in E^{rep}$ and such that all edges are undirected.

In a few terms, the social context of a node is the subgraph induced by the social objects of a given social source and the concepts associated directly to it in a given time interval. An example of social context is given in Fig. 1, right side. The notion of social context is central for the definition of *similarity query* that provides the similarity graph of a given set of nodes. The *node similarity* between two nodes is defined as follows:

Definition 7 (node similarity): Given a knowledge graph $G^K = (V, E, W)$, two nodes $v_i, v_j \in V$, a set of knowledge sources $KS \subseteq \mathcal{KS}$ and a time interval ΔT, the similarity between v_i and v_j, namely $sim(v_i, v_j, KS, \Delta T)$ is a function of the graph $G^{sc}_{(v_i,\, G^K,\, KS,\, \Delta T)} \cup G^{sc}_{(v_j,\, G^K,\, KS,\, \Delta T)}$ s.t. $sim(v_i, v_j, KS, \Delta T) \in \mathbb{R}$.

We can now provide the formal definition of the central notion of *similarity query*:

Definition 8 (similarity query): Given a knowledge graph $G^K = (V, E, W)$, a selection predicate $\mathcal{P}$ on V, a mapping function $\mathcal{F}$, a set of knowledge sources $KS \subseteq \mathcal{KS}$ and a time interval ΔT, a similarity query $Q^{sim}(G^K, KS, \mathcal{P}, \mathcal{F}, \Delta T)$ returns the query graph $G^Q_{sim}(V^Q, E^Q, W^Q)$, where: $V^Q = \mathcal{F}(V')$, where $V' \subseteq V$ is the subset of V satisfying $\mathcal{P}$; $E^Q = \{(v_i, v_j)$ s.t. $v_i, v_j \in V^Q \wedge i < j \wedge sim(v_i, v_j, KS, \Delta T) = sim(v_j, v_i, KS, \Delta T) \neq 0\}$; $W^Q : V^Q \times V^Q \Rightarrow \mathbb{R}$ is the function that associates a weight $w_{ij} = sim(v_i, v_j, KS, \Delta T)$ to each edge $(v_i, v_j) \in E^Q$.

We now show how to instantiate the similarity query in the TV domain. In particular, we provide here a way to compute the *entity similarity* in different similarity query formulations. We are particularly interested in two types of queries: concept similarity queries and cross-source similarity queries.

A. Computing concept similarities

The simplest type of query is the one that involves a target subset of concept nodes $C_{target} \subseteq \mathcal{C}$. Such queries may involve nodes of the same subtype (i.e., only people) or nodes of two different subtypes (e.g., TV events and people). The subtype of a node is stored in a node attribute called $subtype$. Before introducing the definitions of similarity, we define the notion of path strength between two nodes:

Definition 9 (path strength): Given an undirected graph $G = (V, E, W)$ s.t. each node has a $magnitude$ property, two nodes $v_x, v_y \in V$, and a generic path between v_x and v_y denoted by

$$p(v_x, v_y, G) = v_x \xrightarrow{e_{x,x+1}} v_{x+1} \xrightarrow{e_{x+1,x+2}} v_{x+2} \ldots v_{x+i} \xrightarrow{e_{x+i,x+i+1}} \ldots \xrightarrow{e_{x+n-1,x+n}} v_{x+n},$$

where $v_{x+n} = v_y$, the strength of $p(v_x, v_y, G)$ is given by

$$str(p(v_x, v_y, G)) = v_x.magnitude \cdot \prod_{i=1}^{n} v_{x+i}.magnitude \cdot w_{x+i-1,x+i}.$$

Note that when the magnitude of a node or the weight of an edge are not defined, their default value is 1. Thanks to this basic definition, we may define several similarity metrics that involve any two nodes of the graphs in the following way:

Definition 10 (node similarity): Given a knowledge graph G^K, two nodes v_i and v_j, a time interval $\Delta T = (t_{start}, t_{end})$ and a set of social sources $KS \subseteq \mathcal{KS}$, the similarity between v_i and v_j is given by

$$sim\left(v_i, v_j, KS, \Delta T\right) = \sum_{p_i \in P} str\left(p_i\left(v_i, v_j, G^{sc}_{(v_i,\, G^K,\, KS,\, \Delta T)} \cup G^{sc}_{(v_j,\, G^K,\, KS,\, \Delta T)}\right)\right)$$

where P is the set of all paths in the graph resulting from the union of the social context of v_i and v_j.

In our application, this definition is too generic, since the similarity may also involve less relevant paths. Instead, we prefer to consider a TV event-driven similarity which involves multiple social sources whose individual contribution to the metric can be controlled by the user. Furthermore, we focus on the distance between the subset of concepts corresponding to the *named entities*. To this purpose we employ the similarity between a named entity node c_i and a TV event node v^C_e by considering only a social source $ks \in \mathcal{KS}$. This similarity is noted

$sim\left(v_i^C, v_e^C, \{ks\}, \Delta T\right)$. Using this similarity function, we can compute the so-called *TV event-based named entity similarity* between any two named entity concept nodes. In detail, given a set of TV events $C^{TV} \subseteq \mathcal{C}$, we associate, to each named entity concept $c_i \in \mathcal{C}$, a vector $\boldsymbol{c}_i = \{c_{i,1}, \ldots, c_{i,k}, \ldots, c_{i,n}\}$ where $n = |C^{TV}|$ and $c_{i,k} = sim(c_i, c_k^{TV}, \{ks\}, \Delta T)$. The TV event-based similarity between two concepts c_i and c_j is then given by:

$$sim_{TV}\left(c_i, c_j, \{ks\}, \Delta T\right) = \frac{1}{1 + \|\boldsymbol{c}_i - \boldsymbol{c}_j\|_2} = \frac{1}{1 + \sqrt{\sum_{k=1}^{n} |c_{i,k} - c_{j,k}|^2}}$$

i.e., the similarity between two concepts is inversely proportional to the Euclidean distance between their corresponding TV event vectors. Notice that $sim_{TV}\left(c_i, c_j, \{ks\}, \Delta T\right) \in (0, 1]$.

So far, we have considered each source as equivalent. However we may assign a different weight to each knowledge source in order to let the user control the importance of each source in computing the similarity. To this purpose, we slightly modify the definition of TV event-based named entity similarity by introducing an importance coefficient α_k for each source $ks_k \in KS$ (note that $\sum_k \alpha_k = 1$). The *weighted TV event-based named entity similarity* is then given by

$$sim_{TV}\left(c_i, c_j, KS, \Delta T\right) = \sum_{ks_k \in KS} \alpha_k \cdot sim_{TV}\left(c_i, c_j, \{ks_k\}, \Delta T\right).$$

B. Named entity similarity graph

In Section VI-A, we have provided the application-specific notion of TV event-based similarity. We now have all the necessary components to better describe how a named-entity similarity graph looks like. Let $V^{NE} \in \mathcal{C}$ be a set of m named entity concepts and $V^{TV} \in \mathcal{C}$ a set of TV events nodes. The *entity similarity matrix* associated to the knowledge source $ks \in \mathcal{KS}$ is a matrix $M_{NE}^{ks} \in \mathbb{R}^{m \times m}$ such that $m_{ij}^{ks} = sim_{TV}\left(v_i^{NE}, v_j^{NE}, \{ks\}, \Delta T\right)$. Furthermore, each value m_{ij}^{ks} of the matrix is normalized by $\|\boldsymbol{m}_j^{ks}\|$, where $\boldsymbol{m}_j^{ks}$ is the vector associated to the j-th column of matrix M_{NE}^{ks}. We call it the normalized matrix $\overline{M_{NE}^{ks}}$.

As for the weighted TV event-based named entity similarity, we can control the contribution of each knowledge source to the similarity matrix thanks to a weight vector $\boldsymbol{\alpha}$. For a given set $KS \subseteq \mathcal{KS}$ of N knowledge sources, the corresponding *combined entity similarity matrix* $\boldsymbol{M}_{NE}^{KS}$ is then given by

$$\boldsymbol{M}_{NE}^{KS} = \alpha_1 \overline{M_{NE}^{ks_1}} + \ldots + \alpha_k \overline{M_{NE}^{ks_k}} + \ldots + \alpha_N \overline{M_{NE}^{ks_N}}$$

where $\alpha_k \in [0, 1]$ and $\sum_k \alpha_k = 1$.

Definition 11 (TV event-based named entity similarity graph): Let $V^{NE} \in \mathcal{C}$ be a set of m named entity concepts and the associated TV event-based similarity matrix $\boldsymbol{M}_{NE}^{KS}$. The *TV event-based named entity similarity graph* is undirected weighted graph $G_{sim}^{NE}(V^{NE}, E^{NE}, W^{NE})$ where: $E^{NE} = \{(v_i^{NE}, v_j^{NE})$ s.t. $v_i^{NE}, v_j^{NE} \in V^{NE} \wedge i < j \wedge m_{ij}^{KS} = m_{ji}^{KS} \neq 0\}$; $W^{NE} : V^{NE} \times V^{NE} \Rightarrow [0, 1]$ is the function that associates a weight $w_{ij} = m_{ij}^{KS}$ to each edge $(v_i^{NE}, v_j^{NE}) \in E^{NE}$.

C. Cross-source similarities

We have shown how to compute similarities among concepts nodes, but the knowledge graph structure can be also leveraged to measure the strength of the connection between two nodes. In particular, we are interested in measuring the similarity of two social objects belonging to different sources, enabling what we call *cross-source analysis*. For example, computing the similarity between a Facebook post and a tweetset may enable the discovery of hashtags that are closely related to the post. To this purpose, we slightly modify the definition of node similarity (see Definition 10):

Definition 12 (cross-source similarity): Given a knowledge graph G^K, two social object nodes o_i and o_j belonging to sources ks_i and ks_j respectively, and a time interval $\Delta T = (t_{start}, t_{end})$, the similarity between o_i and o_j is given by

$$sim_{cs}\left(o_i, o_j, \Delta T\right) = \sum_{p_i \in P} str\left(p_i\left(o_i, o_j, G^{sc}_{(v_i, G^K, \{ks_i\}, \Delta T)} \cup G^{sc}_{(o_j, G^K, \{ks_j\}, \Delta T)}\right)\right)$$

where P is the set of all paths in the graph resulting from the union of the social context of o_i and o_j.
Note that, in our model, concepts represent the connecting elements among the social objects extracted from the considered sources. For this reason, the relationship among social contexts depends on all paths that involve shared named entities, like people, places, and events. It is worth noting that the cross source similarity can involve subject nodes as well, thus connecting users of different social media platforms.

VII. An Instantiation on Italian Politics

In this section, we describe a real use-case of our framework on an Italian TV show named *Ballarò*[4] dealing with Italian/European politics and broadcasted by RAI. The architecture implementing our framework has been described in [27], [28][5]. Our system has been collecting data related to 66 TV shows since September 1, 2014. However we focused our analysis on *Ballarò* episodes scheduled from January 1, 2015 to June 30, 2015 (twenty-four episodes). This period has been interestingly full of political events for many reasons: the important and controversial reforms of the labor market, school and justice, the immigration crisis (involving relations with the European Commission), and the corruption scandal around Rome administration (*Mafia Capitale*). Some statistics about our dataset are summarized in Table I.

We considered two social sources: Twitter and Facebook. For each episode, we collected all tweets containing *#Ballarò* (the official program hashtag) or *@RaiBallaro* (the official program account). Facebook comments were collected from posts appeared in the official Facebook page.

In particular, we provide three analysis scenarios embracing all analytical capabilities defined in Section VI: the first example is about the analysis of a similarity graph; the second one involves the direct analysis of the knowledge graph G^K; finally, in the third example we show an original cross-source analysis scenario.

TABLE I
DATASET STATISTICS

	Overall data	**Selected period**	**Selected episodes**
# months	18	6	6
# TV shows	66	44	1
# episodes	8,067	2,768	24
# tweets	26,924,690	8,697,890	321,503
# posts	295,912	112,262	6,702
# comments	5,807,955	2,436,969	35,251
$\|\mathcal{O}\|$	59,336	28,378	503
$\|\mathcal{S}\|$	72,012	33,842	612
$\|\mathcal{C}\|$	50,100	22,860	406
$\|E^{sup}\|$	27,816	13,303	230
$\|E^{rep}\|$	192,318	91,966	1,601
$\|E^{str}\|$	176,379	84,533	1,467
$\|E^{tmp}\|$	26,625	11,447	197

TABLE II
TOP BETWEENNESS CENTRALITY SCORES OF NODES FROM DIFFERENT SOCIAL NETWORKS.

(a) Twitter social network

Rank	Person	Centrality
1	Matteo Renzi	0.5244
2	Matteo Salvini	0.1624
3	Silvio Berlusconi	0.0477
4	Murizio Landini	0.0379
5	Elsa Fornero	0.0139

(b) Facebook social network

Rank	Person	Centrality
1	Matteo Renzi	0.1662
2	Massimo Giannini	0.1550
3	Silvio Berlusconi	0.1159
4	Matteo Salvini	0.0714
5	Beppe Grillo	0.0655

(c) Combined social network

Rank	Person	Centrality
1	Matteo Renzi	0.3089
2	Matteo Salvini	0.1208
3	Silvio Berlusconi	0.1008
4	Massimo Giannnini	0.0689
5	Maurizio Landini	0.0362

A. Social Centrality Study

The first example we consider here concerns the study of the importance (in terms of centrality) of persons (politicians, television people, presenters, guests), during the observation period. To perform this analysis, we

[4]http://www.ballaro.rai.it

[5]The details of the architecture and the experiments are available online at http://www.di.unito.it/~pensa/papers/cim16additional.pdf

consider all the persons referred by the tweets and Facebook posts associated to all episodes of the TV show and build the underlying social network following the definition of *TV event-based named entity similarity graph* (see Definition 11 in Section VI-B). According to this definition, there is an edge between two person nodes if there exists a path between these two persons, traversing a *TV Event* node. Interestingly, these paths may involve cross-source nodes, i.e., the analysis of an individual source, without our knowledge integration framework, would have led to a different, less precise, social network. Since more than one tweetset and Facebook post may exist during the week associated to each episode, for each episode, the similarity graph is such that all the tweetsets and Facebook posts are merged to obtain an aggregated episode representation. Moreover, each of them is associated to the set of the most mentioned persons during the considered period.

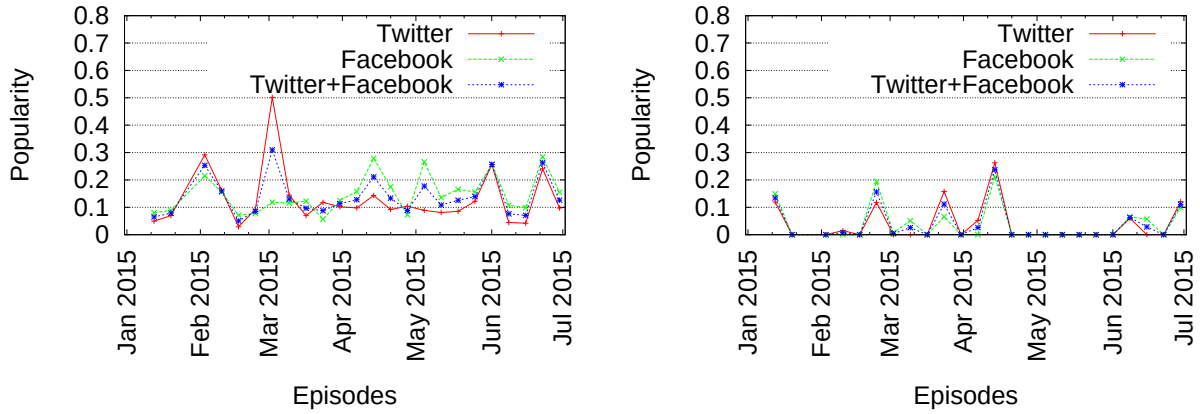
On our Twitter data, the above described analysis produced a social network that we analyzed by computing the betweenness centrality [34], [35] of each node (i.e., the number of shortest paths from all vertices to all others that pass through that node), obtaining the results in Table II (a). These results show that Matteo Renzi is very central for this TV program. He is the Italian prime minister and he was strongly involved in each reform cited before. It is important to note that he is the most central concept in terms of betweenness even if he participated only in few episodes during the observation period. The following concept is Matteo Salvini, that is one of the most active leaders of the center-right coalition especially known for his stance against illegal immigration and his strong criticism on the rules coming from the European Commission about economy and immigration.

The betweenness centrality computed for the Facebook data on people belonging to the knowledge graph is reported in Table II (b). The most important concept in terms of centrality is always Matteo Renzi, but, interestingly, his centrality appears less dominant w.r.t. other people. This analysis shows that one of the best ranked persons is Beppe Grillo. This is probably due to the fact that Grillo's supporters are particularly active in this social media platform. Thus, in this social network, the position of Grillo is more central than in the Twitter case.

By combining the two information sources, we may notice that all the relevant information for both sources is preserved, as shown by the ranked betweenness scores in Table II (c). In particular, Matteo Renzi, the Prime Minister is still the most central concept and, almost all the most important Italian politics actors are in the first positions of the ranking.

B. Popularity Study

The second experiment consists in computing the "episode popularity" of each person. The popularity of a given node is related to the percentage of citations of the associated persons' names in tweets and Facebook post comments. Notice that this information is stored in the knowledge graph as the weight of the edge connecting each person to the *People* node, by the resource extractors. Hence, to conduct this analysis, we only need to aggregate the weights of the out-edges of each person node. Within a single source the aggregation is performed by merging all social objects (a tweetset or a Facebook post) related to a given episode. Then, each edge weight is multiplied by the total number of occurrences of the concept node *People*. Finally, the cut-off method based on energy is employed to filter out the less important entries. To consider the popularity in both Twitter and Facebook as a



(a) Matteo Renzi (b) Maurizio Landini

Fig. 2. Episode popularity (computed according to Equation 1) of some cited persons from our knowledge graph

whole, we merged the Facebook post nodes and Tweetset nodes associated to each episode. The resulting weight for each person node i is then computed as

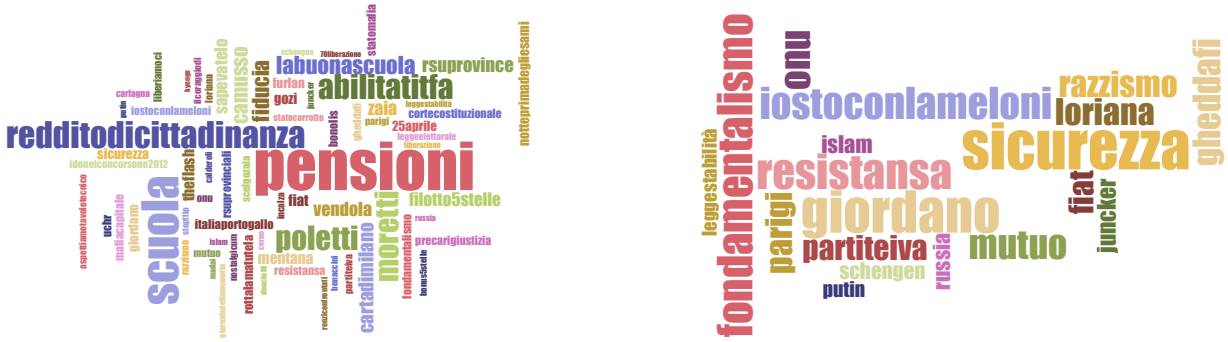
$$w(i)_{all} = \alpha \cdot w(i)_t + (1 - \alpha) \cdot w(i)_f, \tag{1}$$

where $w(i)_t$, $w(i)_f$ and $w(i)_{all}$ are, respectively, the node weights of the edge connecting i to the Tweetset node, the node weights of the edge connecting i to the Facebook post node, and the resulting weight associated to the edge connecting i to the aggregated social object node. In this experiment, we considered all sources with the same weight, i.e., $\alpha = 0.5$.

Figure 2 shows the results for the top-ranked personalities. As can be seen, the popularity of Matteo Renzi is quite stable during the observation period, while the popularity of Maurizio Landini, one of the most important trade unionists, is strongly related to episodes in which the main theme was the labor reform.

C. An Example of Cross-Source Analysis

As an example of the potential analysis scenarios that our framework may enable, we consider non-trivial associations between Facebook posts and Twitter hashtags. These two objects are not immediately linked: users' communities and social platforms are different. However, they may have in common several entities (persons, nouns, events, and emotions). Thanks to our framework, it is rather simple to compute the entities that connect posts and hashtags. We then construct a hashtags $\times$ post matrix (called M) in the following way. For a given post p and a given hashtag h, the value m_{hp} of matrix M, is given by $m_{hp} = sim_{cs}\,(o_h, o_p, \Delta t)$, where o_h is the social object associated to the hashtag h, o_p is the social object associated to the Facebook post o_p, Δt is the whole six months analysis period and sim_{cs} follows Definition 12 given in Section VI-C. We repeat this computation for each pair (h, p) of hashtags h and posts p. We ignore all concept nodes related to emotions in this case. As a result, the association of all posts and tweets related to the monitored period, leads to a matrix M of 179 most used hashtags and 1,679 Facebook posts, consisting of 144,023 non-zero values.



(a) (b)

Fig. 3. Two examples of hashtag clusters about (a) economic reforms and (b) foreign politics.

It is now interesting to obtain associations between groups of hashtags and groups of Facebook posts. As an example, we may imagine cross-domain recommendation of interesting Twitter hashtags to people reading and commenting Facebook posts. To compute relevant cross-associations, we use a *hierarchical co-clustering* algorithm [36]. It identifies a hierarchy of clusters of rows and an associated hierarchy of clusters of columns by optimizing the Goodman-Kruskal's τ association measure. The algorithm is parameter-less, and build compact hierarchies with n-ary splits. We apply this algorithm and consider two coupled levels of the hierarchy: the first level, with a coarse grid of 3×3 co-clusters; the third level with a more fine-grained grid of 14×20 co-clusters.

We then associate each cluster R of rows (Facebook posts) with the cluster C of columns (hashtags) such that $\frac{1}{|R|\cdot|C|} \sum_{h \in R} \sum_{v \in C} m_{hv}$ is maximized.

As an example, we considered two results at different level of the hierarchical co-clustering. In the former, considering the first hierarchical level, one of the row co-cluster contains 164 Facebook posts that are associated to a cluster of 72 hashtags. As can be seen in Fig. 3 (a), they are mostly related to discussions about the highly debated economic reforms and school. The second example (see Fig. 3 (b)) is extracted from the third level of the hierarchical clustering and associates a set of 49 Facebook posts to the reported set of hashtags: as can be seen, the set of terms depict the discussion, mostly debated by the center-right parties, around immigration and terrorism.

VIII. Conclusions and Future Work

In this paper, we proposed an integration framework in which TV users' activities on different social media are collectively represented, and possibly enriched with external knowledge, such as information extracted from the EPGs, or available ontological domain knowledge. We also discussed different types of analysis that the integration data model enables.

Many research problems remain open. As future work, we will address the scalability issues which immediately emerge when engineering an industrial system based on the presented framework. The intense activity of social media users turns into the high dynamicity of the knowledge graph. Hence, we are studying incremental (possibly approximate) versions of the algorithms computing graph based popularity measures.

The data model allows us to track the temporal evolution of users' activities. Thus, another future work includes the application of innovative algorithms and techniques to analyze time series extracted from the graph. This will allow us to capture and study how social phenomena (popularities, users' interests, and communities of users sharing common interests) evolve in time.

Finally, for future work, we plan to leverage the most recent research results in social media analytics and sentiment analysis to further improve our framework. In particular, we will adopt some event detection techniques (such as the one presented in [37]) to support the automatic detection of emerging topics in social media and we will consider sentic computing [38], [39] and AffectiveSpace [40], to bring sentiment analysis up to concept-level.

ACKNOWLEDGMENTS

We are grateful to Alessio Antonini, Roberto Del Pero and Fulvio Negro for their constructive discussions during the formalization of the integration framework.

REFERENCES

[1] X. Han, W. Wei, C. Miao, J. Mei, and H. Song, "Context-aware personal information retrieval from multiple social networks," *IEEE Comp. Int. Mag.*, vol. 9, no. 2, pp. 18–28, 2014.

[2] H. Alostad and H. Davulcu, "Directional prediction of stock prices using breaking news on twitter," in *IEEE/WIC/ACM International Conference on Web Intelligence (WI'15)*, Singapore, 2015.

[3] N. Kim, S. Gokalp, H. Davulcu, and M. Woodward, "LookingGlass: A visual intelligence platform for tracking online social movements," in *Proceedings of ASONAM '13*. ACM, 2013, pp. 1020–1027.

[4] M. Kaschesky, P. Sobkowicz, and G. Bouchard, "Opinion mining in social media: Modeling, simulating, and visualizing political opinion formation in the web," in *Proceedings of the 12th Annual International Conference on Digital Government Research, DG.O 2011*, 2011, pp. 317–326.

[5] S. Alashri, S. Alzahrani, L. Bustikova, D. Siroky, and H. Davulcu, "What animates political debates? Analyzing ideological perspectives in online debates between opposing parties," in *Proc. ASE/IEEE Int. Conf. on Social Computing (SocialCom-15)*, Stanford, CA, 2015.

[6] M. Fleischman and D. Roy, "Grounded language modeling for automatic speech recognition of sports video," in *Proceeding of ACL 2008*, Columbus, Ohio, USA, 2008, pp. 121–129.

[7] P. DeCamp and D. Roy, "A human-machine collaborative approach to tracking human movement in multi-camera video," in *Proceedings of ACM CIVR 2009*. Santorini Island, Greece: ACM, 2009.

[8] A. Saha, M. Pal, and T. K. Pal, "Selection of programme slots of television channels for giving advertisement: A graph theoretic approach," *Inf. Sci.*, vol. 177, no. 12, pp. 2480–2492, 2007.

[9] D. O'Sullivan, B. Smyth, D. C. Wilson, K. McDonald, and A. F. Smeaton, "Improving the quality of the personalized electronic program guide," *User Model. User-Adapt. Interact.*, vol. 14, no. 1, pp. 5–36, 2004.

[10] M. Yan, J. Sang, and C. Xu, "Unified youtube video recommendation via cross-network collaboration," in *Proceedings of ACM ICMR 2015*. ACM, 2015, pp. 19–26.

[11] F. Stokman and P. Vries, "Structuring knowledge in a graph," in *Human-Computer Interaction*, G. Veer and G. Mulder, Eds. Springer Berlin Heidelberg, 1988, pp. 186–206.

[12] A. S. Liaguno and A. F. Connor, "System for storage and retrieval of diverse types of information obtained from different media sources which includes video, audio, and text transcriptions," 1998, patent No. US 5729741.

[13] P. Yan and W. Jin, "Improving cross-document knowledge discovery using explicit semantic analysis," in *Proceedings of DaWaK 2012*. Vienna, Austria: Springer, 2012, pp. 378–389.

[14] Y. Yoshida, T. Hirao, T. Iwata, M. Nagata, and Y. Matsumoto, "Transfer learning for multiple-domain sentiment analysis – Identifying domain dependent/independent word polarity," in *Proceedings of AAAI 2011*. San Francisco, CA, USA: AAAI Press, 2011.

[15] Q. Zhang, Y. Wu, Y. Wu, and X. Huang, "Opinion mining with sentiment graph," in *Proceedings of IEEE/WIC/ACM Web Intelligence 2011*. Lyon, France: IEEE Computer Society, 2011, pp. 249–252.

[16] A. C. Kaluarachchi, D. Roychoudhury, A. S. Varde, and G. Weikum, “SITAC: Discovering *semantically identical temporally altering concepts* in text archives,” in *Proc. of EDBT 2011*. Uppsala, Sweden: ACM, 2011, pp. 566–569.

[17] K. D. Bollacker, R. P. Cook, and P. Tufts, “Freebase: A shared database of structured general human knowledge,” in *Proceedings of the 22nd AAAI Conference on Artificial Intelligence AAAI 2007*. AAAI Press, 2007, pp. 1962–1963.

[18] K. D. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, “Freebase: A collaboratively created graph database for structuring human knowledge,” in *Proceedings ACM SIGMOD 2008*. ACM, 2008, pp. 1247–1250.

[19] R. Navigli and S. P. Ponzetto, “BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network,” *Artif. Intell.*, vol. 193, pp. 217–250, 2012.

[20] O. Corby and C. Faron-Zucker, “The KGRAM abstract machine for knowledge graph querying,” in *Proc. of the 2010 International Conference on Web Intelligence, WI 2010*. IEEE, 2010, pp. 338–341.

[21] B. Gretarsson, J. O’Donovan, S. Bostandjiev, T. Höllerer, A. U. Asuncion, D. Newman, and P. Smyth, “TopicNets: Visual analysis of large text corpora with topic modeling,” *ACM TIST*, vol. 3, no. 2, 2012.

[22] S. Trißl and U. Leser, “Fast and practical indexing and querying of very large graphs,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2007*. ACM, 2007, pp. 845–856.

[23] R. Jin, N. Ruan, Y. Xiang, and H. Wang, “Path-Tree: An efficient reachability indexing scheme for large directed graphs,” *ACM Trans. Database Syst.*, vol. 36, no. 1, p. 7, 2011.

[24] H. Yildirim, V. Chaoji, and M. J. Zaki, “GRAIL: A scalable index for reachability queries in very large graphs,” *VLDB J.*, vol. 21, no. 4, pp. 509–534, 2012.

[25] R. Jin, N. Ruan, S. Dey, and J. X. Yu, “SCARAB: Scaling reachability computation on large graphs,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012*. ACM, 2012, pp. 169–180.

[26] L. Khan, D. McLeod, and E. H. Hovy, “Retrieval effectiveness of an ontology-based model for information selection,” *VLDB J.*, vol. 13, no. 1, pp. 71–85, 2004.

[27] A. Antonini, L. Vignaroli, C. Schifanella, R. G. Pensa, and M. L. Sapino, “MeSoOnTV: A media and social-driven ontology-based TV knowledge management system,” in *Proceedings of ACM HT ’13*. ACM, 2013, pp. 208–213.

[28] A. Antonini, R. G. Pensa, M. L. Sapino, C. Schifanella, R. T. Prioletti, and L. Vignaroli, “Tracking and analyzing TV content on the web through social and ontological knowledge,” in *Proceedings of EuroITV ’13*. ACM, 2013, pp. 13–22.

[29] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM Review*, vol. 51, no. 3, pp. 455–500, September 2009.

[30] B. Bara, *Cognitive Pragmatics: The Mental Processes of Communication*. MIT Press, 2010.

[31] P. Johnson-Laird, *Mental Models*. Harvard Univ Press, 1983.

[32] J. R. Searle, *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1970.

[33] ——, *The construction of social reality*. New York: Free Press, 1995.

[34] L. C. Freeman, “A set of measures of centrality based on betweenness,” *Sociometry*, vol. 40, no. 1, pp. pp. 35–41, 1977.

[35] M. Newman, *Networks: An Introduction*. Oxford University Press, 2010.

[36] R. G. Pensa, D. Ienco, and R. Meo, “Hierarchical co-clustering: Off-line and incremental approaches,” *Data Min. Knowl. Discov.*, vol. 28, no. 1, pp. 31–64, 2014.

[37] S. Rill, D. Reinel, J. Scheidt, and R. V. Zicari, “PoliTwi: Early detection of emerging political topics on twitter and the impact on concept-level sentiment analysis,” *Knowl.-Based Syst.*, vol. 69, pp. 24–33, 2014.

[38] S. Poria, E. Cambria, G. Winterstein, and G. Huang, “Sentic patterns: Dependency-based rules for concept-level sentiment analysis,” *Knowl.-Based Syst.*, vol. 69, pp. 45–63, 2014.

[39] S. Poria, E. Cambria, A. F. Gelbukh, F. Bisio, and A. Hussain, “Sentiment data flow analysis by means of dynamic linguistic patterns,” *IEEE Comp. Int. Mag.*, vol. 10, no. 4, pp. 26–36, 2015.

[40] E. Cambria, J. Fu, F. Bisio, and S. Poria, “AffectiveSpace 2: Enabling affective intuition for concept-level sentiment analysis,” in *Proceedings of AAAI 2015, January 25-30, 2015, Austin, Texas, USA*. AAAI Press, 2015, pp. 508–514.