SView: Smart Views for Browsing Linked Entities

Yuzhong Qu, Saisai Gong, Gong Cheng, Jiang Xu, Xiangqian Li, Liang Zheng, and Jidong Jiang

State Key Laboratory for Novel Software Technology, Nanjing University,
Nanjing 210023, PR China
{yzqu,gcheng}@nju.edu.cn, {ssgong,zhengliang}@smail.nju.edu.cn,
{xujiang0804,skyline0623,jdkjiang}@gmail.com

Abstract. For years, it has been a great challenge to provide general users with smart views for browsing interlinked RDF descriptions of entities. To meet the challenge, we developed a system called SView for browsing linked entities. It groups and orders entity descriptions via lenses for a neat presentation, and provides various mechanisms for exploring related entities such as exploration based on link pattern and similarity-based entity recommendation. Besides, users can personalize their browsing experience, and they never work alone. They can edit lenses and consolidate entities following their own opinions, and their efforts are minimized due to crowdsourced contributions from all users. Furthermore, SView leverages users' contributions to generate smart views, e.g. global lenses and global viewpoints on entity consolidation. These smart views are, in turn, shared among all users when browsing linked entities.

1 Pitch (for Non-experts)

As a fan of James Cameron and his movies, yesterday you learned his early life at Wikipedia, checked the list of all his works at IMDb, and searched for comments on his latest work at Twitter. As always, it took you a great deal of time to find these pages on the Web and extract interesting information from a mass. When you have been tired of this tedious work, fortunately, the emerging "Linked Data" technology is causing a revolution in the Web. With Linked Data, information from different websites is also available as a kind of reusable data. Then all you need is a handy assistant that integrates all pieces of relevant data and presents it to you in an appropriate way, without taking you too much effort.

SView is undoubtedly the assistant you are looking for. As you might expect, with SView which retrieves and integrates data from different websites, you are not going to open them in many tabs in your browser, and you will not be forced to scan those long, boring, invariable pages for particular information. Instead, you are free to, following your own taste, customize the way to present information by creating and editing as many "lenses" as you wish. You can, for example, put descriptions of Cameron's early life and his recent cool pictures in a lens called “personal life”. You may also want to enrich the list of his works with comments from social networking services, and see both of them in one place, namely a lens called “works and reviews”. One fancy thing here is SView’s “do once, use many times” feature. That is, next time when you become a fan of Ang Lee, his information will also come in a similar way—“personal life” and “works and reviews”. So yes, SView is your personalized assistant.

More than that, you don’t really have to start every lens from scratch because you never work alone. Your and other people’s lenses are automatically aggregated and shared among all users. Why don’t you stand on the shoulders of the crowd and reuse their lenses or, if you are willing to, also make your contribution to the crowd? So yes, SView is an assistant based on a collaborative effort, for your convenience and expecting your participation.

SView can do even more for you. Are you wondering which English stage actors have starred in films directed by Cameron? Come to SView and find the answers by yourself, with only a couple of clicks.

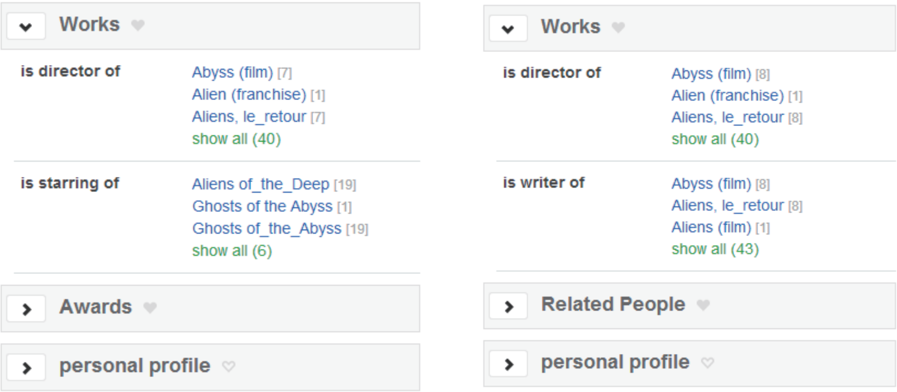
2 Introduction and Overview (for Experts)

Linked Data complements the Web of documents with interlinked entity descriptions from different sources. It enables a wide range of novel Web applications, but also challenges existing softwares for the Web, in particular Web browsers. Specifically, among others, RDF descriptions about entities retrieved from different sources, which have been separated from their presentation, are in need of a general-purpose tool for human users to browse. This tool should also facilitate the exploration of related entities that builds upon and goes beyond explicit RDF links. To truly embrace the decentralized nature of Linked Data, the tool is also expected to find and fuse entity descriptions from different sources.

To meet these challenges, we developed a system called SView[1] for browsing linked entities, which is a Web application in itself. In SView, entity descriptions are fused based on entity and property consolidation, and are appropriately grouped and ordered by lenses for a neat presentation. Related entities can be found and explored via not only explicit links organized meaningfully but also implicit links induced by various similarity measures. Furthermore, the main philosophy of our system, which is very different from other systems such as Tabulator, Haystack and Disco, is to leverage the force of the user community to generate smart views for browsing linked entities. A user will enjoy personalized browsing experience by putting effort into, e.g. editing lenses and consolidating entities, and she actually puts far less effort than needed due to crowdsourced contributions from all users. Particularly, SView leverages users’ contributions to generate smart views such as lenses in global scope, features in global scope and global viewpoints on entity consolidation. These smart views are, in turn, shared among all users when browsing linked entities.

From Sect. 3 to Sect. 5, we elaborate on browsing, exploration, and fusion of entity descriptions in SView. In Sect. 6, we introduce other functionalities such

[1] `http://ws.nju.edu.cn/sview`



Works

is director of	Abyss (film) [7] Alien (franchise) [1] Aliens, le_retour [7] show all (40)
is starring of	Aliens of_the_Deep [19] Ghosts of the Abyss [1] Ghosts of_the_Abyss [19] show all (6)

Awards

personal profile

(a) User X.

Works

is director of	Abyss (film) [8] Alien (franchise) [1] Aliens, le_retour [8] show all (40)
is writer of	Abyss (film) [8] Aliens, le_retour [8] Aliens (film) [1] show all (43)

Related People

personal profile

(b) User Y.

Fig. 1: Lenses personalized by different users for James Cameron.

as entry recommendation and data export. Finally, Sect. 7 gives some lessons we learned from the SView system.

3 Browsing Entity Descriptions via Personalized and Crowdsourced Lenses

In SView, an entity description is composed of a set of features. A feature is either an RDF property or a set of consolidated equivalent properties (cf. Sect. 5). An entity description may contain hundreds of or even more features, which are not very readable if there is no appropriate organizing method. To address this issue, SView groups and orders features via lenses [1], which can be personalized by registered users, as illustrated in Fig. 1. When browsing an entity, each user is provided with lenses appropriately selected from both her personalized ones and global ones. Global lenses are automatically generated by the system based on the personalized lenses of the crowd (i.e. all users).

3.1 Lens Selection

In SView, a group of ordered features is called a lens [1], which is supposed to comprise features describing closely related aspects of an entity and thus to help user capture related information quickly. When browsing an entity, the user not necessarily sees all the personalized and global lenses that are relevant to this entity because they may contain many common features. To minimize redundancy, firstly, priority is given to the user's personalized lenses. A weighted set cover problem is formulated and solved to automatically select a small number of these lenses that can cover as many relevant features as possible. Then, this process is repeated on the global lenses to cover the rest of features.

3.2 Lens Personalization and Crowdsourcing

During browsing, every registered user is free to personalize the way of grouping and ordering features by either marking a global lens as favorite or creating a new lens, and then conveniently editing them via drag and drop. A single user does not need to create all lenses due to limited time and effort. Our system provides relevant global lenses for all users, and these global lenses come from the wisdom of all users. Actually, the system leverages the personalized lenses of the crowd to generate a set of global lenses that are representative of their major opinions on grouping and ordering features.

Regarding the generation of global lenses, we observe that each user's personalized lenses can naturally induce a (partial) clustering of features/properties. Clusterings from all users are then aggregated into a single median clustering that, in general, best agrees with each individual clustering [2]. This median clustering gives rise to the grouping of features in global lenses. To determine the ordering of features in each global lens, the idea is similar. Each user's personalized lenses can induce a (partial) ordering of features/properties, based on which a global ordering of features is formed [3] to be used in global lenses.

4 Exploring Linked and Similar Entities

In SView, a feature having an entity as its value is called a link. When browsing an entity description via lenses, the user can navigate from this entity to its related entities by following links. In addition to such traditional feature-oriented method, SView provides two main mechanisms for exploring related entities: exploration based on link pattern, and similarity-based entity recommendation.

4.1 Exploration based on Link Pattern

SView extracts links from entity descriptions and organizes them in an entity-oriented manner in the "Links" panel, as illustrated in Fig. 2. Specifically, we observe that an entity may have more than one link to another entity. For instance, James Cameron is the writer and also the director of Avatar. The set of all links to a linked entity is called its link pattern. Linked entities sharing a common link pattern are grouped together to be explored. For instance, Fig. 2 shows three link patterns, the second of which (i.e. the expanded one) comprises two links and is shared by six linked entities. From another point of view, link patterns induce a partition on linked entities.

The user can browse the union of the descriptions of entities sharing a common link pattern. That is, SView supports pivoting from an entity (or a set of entities) to another set of entities[2]. The user can also filter the results by choosing the types of target entities. By using pivoting and filtering, more powerful queries can be incrementally constructed to meet complex information needs, e.g. English stage actors starring in films directed by James Cameron.

[2] When browsing a set of entities, lenses are transformed into tables where rows and columns represent entities and features, respectively.

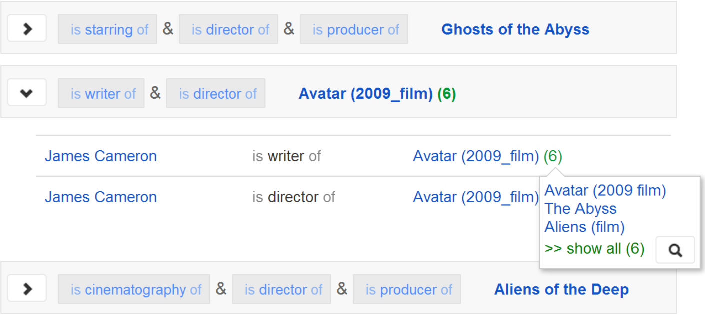

is starring of & is director of & is producer of Ghosts of the Abyss
is writer of & is director of Avatar (2009_film) (6)
James Cameron is writer of Avatar (2009_film) (6)
James Cameron is director of Avatar (2009_film)
Avatar (2009 film)
The Abyss
Aliens (film)
>> show all (6)
is cinematography of & is director of & is producer of Aliens of the Deep

Fig. 2: Exploration based on link pattern.

4.2 Similarity-based Entity Recommendation

When browsing an entity description, SView can use multiple strategies to recommend its similar entities to be explored. Entities to be recommended come from the browsing history of all users, and they may or may not be reachable by following links. Therefore, similar entities complement explicitly linked entities.

One default strategy is to find entities sharing common types with the current entity. Entities sharing more common types are more likely to be recommended. Furthermore, this strategy can be generalized to other features as well. Specifically, SView allows the user to choose one or more features, and then finds entities sharing common values with the current entity on these features.

5 Fusing Entity Descriptions via Personalized Learning

SView is able to fuse and present entity descriptions from different sources. To achieve it, data fusion is carried out at two levels: entity consolidation and property consolidation, both due to the combined efforts of users and machines.

5.1 Entity Consolidation

Entity consolidation, namely to identify entity descriptions that refer to the same real-world entity, is a difficult task due to not only the complexity of semantic heterogeneity but also users' different opinions. For instance, some but not all users agree to consolidate the descriptions of two different editions of a movie. In light of this, SView enables personalized entity consolidation, allowing different users to own different results of consolidation. Specifically, when a user browses an entity description, SView finds other entity descriptions that may be consolidated with it by using machine learning and will be confirmed by the user. Candidates come from entities linked by `owl:sameAs`, results of consolidation from other users, and results retrieved from services like sameas.org and Falcons. A perceptron predicates the confidence score of consolidating each candidate, which is continuously trained online based on the results of consolidation that have been confirmed by the user. Highly confident ones are automatically consolidated, and the others are to be confirmed by the user. To help the user efficiently make a decision, a compact summary of entity descriptions is computed and presented [4]. All users' results of consolidation will be collected and integrated into a partition on entities [5], i.e. global results of entity consolidation, which increases the scale of entity consolidation and resolves the disagreements simultaneously.

5.2 Property Consolidation

Property consolidation, namely to consolidate a set of semantically equivalent properties into a feature, is done in a way similar to the grouping of features described in Sect. 3. In brief, during browsing, every registered user is free to conveniently group equivalent properties into features. Features created by different users are then solicited and integrated into a single median partition [2], i.e. a set of global features, which is, in turn, shared by all users.

6 Other Functionalities

6.1 Entry Recommendation

Basically in SView, browsing starts with an entity identified by a URI, or begins with keyword search for entities to be explored. A registered user can also start with an entity in her bookmarks. To serve casual users, a wide range of widgets are implemented based on the browsing history of all users, and are incorporated into the portal to offer various kinds of entry entities.

- "Categories" generates ten clouds of concepts by mapping the types of entities in browsing history to WordNet concepts. Clicking a concept will trigger a search for entities of this type.
- "Hot Entities" are those frequently browsed in browsing history.
- "Today in History" finds and shows entities involved in events that happened today in history.

6.2 Data Export

In SView, all personalized lenses, personalized features, and entities in bookmarks are stored in RDF. Whenever needed, registered users can export, in JSON-LD format, all of these artifacts and bookmarks as well as raw RDF data being browsed, to be reused in other applications.

7 Lesson Learned

We have successfully deployed the SView system, which enables general users to handily browse linked entities. From the early-stage experience of using the system, we also learned some lessons. Firstly, retrieving entity descriptions from many sources is time-consuming, especially in the case of browsing a large collection of entities, which may negatively influence user experience. It implies that powerful data caches and more sophisticated data access methods are needed. Secondly, the diversity and heterogeneity of entity description is much more serious than imagined. We should pay more attention to the issue of data fusion. For example, active learning can be incorporated in the system to quickly find out equivalent properties. Thirdly, how to measure the goodness of the global lenses is an interesting research problem, which is one of our future work.

Acknowledgments. This work was supported by the NSFC under Grant 61170068, 61100040, and 61321491.

References

1. D. Quan, D. Karger. How to Make a Semantic Web Browser. In Proc. WWW, pages 255–265, 2004.
2. N. Ailon, M. Charikar, A. Newman. Aggregating Inconsistent Information: Ranking and Clustering. J. ACM, 55(5):23, 2008.
3. T. Coleman, A. Wirth. Ranking Tournaments: Local Search and a New Algorithm. J. Exp. Algor., 14:6, 2009.
4. D. Xu, G. Cheng, Y. Qu. Facilitating Human Intervention in Coreference Resolution with Comparative Entity Summaries. In Proc. ESWC, pages 535–549, 2014.
5. S. Gong, W. Hu, Y. Qu. Leveraging Distributed Human Computation and Consensus Partition for Entity Coreference. In Proc. ESWC, pages 411–425, 2014.

Appendix

Table 1: Minimal requirements

Requirement	How SView meets
1. The application has to be an end-user application.	As a Web application, SView enables general users to browse entities in the Linked Data.
2. The information sources used should be under diverse ownership or control, be heterogeneous, and contain substantial quantities of real world data.	SView can be used to browse entities from any sources in the Linked Data, and can particularly handle heterogeneous data from different sources via entity and property consolidation.
3. The meaning of data has to play a central role.	(1) The manipulation of features and lenses exactly revolves around the meaning of properties. (2) Class hierarchy is used for results filtering.

Table 2: Additional desirable features

Feature	How SView meets
1. The application provides an attractive and functional Web interface (for human users).	(1) SView provides lenses for browsing and various mechanisms for exploration. (2) Users can personalize browsing experience by editing lenses and consolidating entities and properties. (3) Various entries are provided for casual users.
2. The application should be scalable (in terms of the amount of data used and in terms of distributed components working together). Ideally, the application should use all data that is currently published on the Semantic Web.	SView can be used to browse any entities in the Linked Data. Furthermore, it includes the "ID" (Entity Identity) subsystem for entity consolidation, and workbenches for editing lenses and features in the process of browsing entity.
3. Novelty, in applying semantic technology to a domain or task that have not been considered before.	(1) SView allows personalized organization of entity descriptions via lenses, and supports personalized entity and property consolidation. (2) SView minimizes each individual user's effort by soliciting and integrating contributions from all users and sharing the results.
4. Functionality is different from or goes beyond pure information retrieval.	SView focuses on both the retrieval of and the interaction with information, e.g. lenses for browsing, various mechanisms for exploration, and entity and property consolidation.
5. Multimedia documents are used in some way.	Among others, images of entities are displayed.
6. There is a use of dynamic data (e.g. workflows), perhaps in combination with static information.	(1) Entity descriptions are retrieved online from Linked Data. (2) During browsing, lenses are dynamically selected and can be dynamically personalized. (3) In entity consolidation, the perceptron is continuously trained online and updates its model whenever the user confirms a new consolidation.
7. The results should be as accurate as possible (e.g. use a ranking of results according to context).	(1) A global ordering of features is periodically updated by aggregating users' personalized orderings on features. (2) Entity descriptions to be consolidated are found by a personalized, continuously trained learner. The results will be confirmed by the user.