Towards a SPARQL 1.1 Feature Benchmark on Real-World Social Network Data

Martin Przyjaciel-Zablocki, Alexander Schätzle,
Thomas Hornung, and Io Taxidou

Department of Computer Science, University of Freiburg, Germany
{zablocki,schaetzle,hornungt,taxidou}
@informatik.uni-freiburg.de

Abstract. In recent years, social networks have fundamentally changed our perception of the web and the way we interact with it. At the same time we have witnessed the vision of the *"Semantic Web"* picking up pace. From a general perspective, the inherent complex intertwined structure of a social network contains a flood of semantic information about users, objects and their relations. On the other hand, social graph structures are hardly covered by current state-of-the-art RDF benchmarks. Moreover, synthetic graph generators do not model all properties of a social network, especially structural correlations are either neglected or underrepresented. Considering the complex structure of a social graph, the enhanced features of SPARQL 1.1 open up new valuable possibilities, but these features are also currently neglected by most of the existing benchmarks. In this paper we introduce our concept of a new RDF benchmark based on real-word social network data gathered from *Last.fm* with a special focus on SPARQL 1.1.

1 Introduction

The advent of the *"Semantic Web"* promotes the growing adoption of RDF and SPARQL as its core technologies. We believe that current initiatives like schema.org, Google's Knowledge Graph, the Linking Open Data (LOD) cloud as well as structured data markups for search engine optimization will further drive the propagation of these technologies. Furthermore, social networks like Google+, Facebook, Twitter, Last.fm etc. dramatically change the way how people interact, collaborate and share information, turning the traditional *"Web of Documents"* into an highly interactive and personalized interlinked *"Web of Data"*. According to this perspective, one can also interpret a social network graph as structured semantic data interlinking people and objects that can also be represented in RDF. This is also underpinned by the support for RDF added to Facebook's Graph API in 2011 [18].

On the other hand, there is a lack of real-world RDF benchmark data in general, and social network data in particular [4, 16]. Most of the existing RDF benchmarks like BSBM [2], LUBM [7] or SP^{2}Bench [15] use artificial data generated according to observed frequency distributions of a specific domain. While this approach allows to easily scale the size, the generated datasets have little in common with real RDF data as they resemble relational database benchmarks [12] with a high level of structuredness [4]. One of the few benchmarks using real data is the DBpedia SPARQL benchmark (DBPSB) [12] based on data dumps from DBpedia. However, compared to the size and dynamics of social graphs, the DBpedia data is rather small and also limited in growth. The dataset used in [12] had a total size of ∼150 million RDF triples which shouldn't pose a challenge for state-of-the-art RDF triple stores.

Structural correlations are ubiquitous in social graphs, e.g. friendship relationships are correlated with the place of residence. Knowledge of these correlations can have an important impact on query optimization but identifying them is a non-trivial task. The S3G2 data generator for structure-correlated social graphs [14] that is used for the Social Network Intelligence Benchmark (SIB) [3] focuses on this aspect. It can be used to generate arbitrary large social graphs with a pre-defined set of structural correlations. However, the authors emphasize that the generator will not produce "realistic" social network data as these networks are expected to have many more (yet unknown) correlations. To overcome the conceptual shortcomings of synthetic data generators we outline a benchmark based on real-world data gathered from the social music network *Last.fm*. We decided to use Last.fm since it exhibits the characteristics of a social network (cf. Section 2) with millions of users and provides a public API[1] to access the data. In addition, our benchmark will focus on the new features of SPARQL 1.1 [8], i.e. *Property Paths*, *Aggregates*, *Subqueries* and *Negation*, as they open up new possibilities for more sophisticated graph queries (cf. Section 3) which are of special interest for social networks regarding their typically complex graph structure. Indeed, these new features are neglected or not considered at all by current popular RDF benchmarks. To the best of our knowledge, this will be the first RDF benchmark on large-scale real-world social network data with a special focus on SPARQL 1.1.

Paper Structure. Section 2 discusses the social network characteristics of Last.fm as well as the fragment that we will use for our benchmark dataset. In Section 3 we introduce some exemplary queries to demonstrate the power of SPARQL 1.1 for querying social network graphs, followed by a conclusion in Section 4.

2 Last.fm Benchmark Data

Last.fm is an online music service with manifold relations between people, artists, tracks, etc. that constitute a highly connected graph with a large variety of correlations. In order to justify Last.fm as an appropriate base for our benchmark dataset, we analyzed its underlying social graph and investigated common social network characteristics. First of all, we crawled about 1.7 million users with close to 13.6 million friendship relationships using a Breadth-First Search (BFS) strategy. We are aware of the biases introduced by BFS in terms of degree

[1] `http://www.last.fm/api`

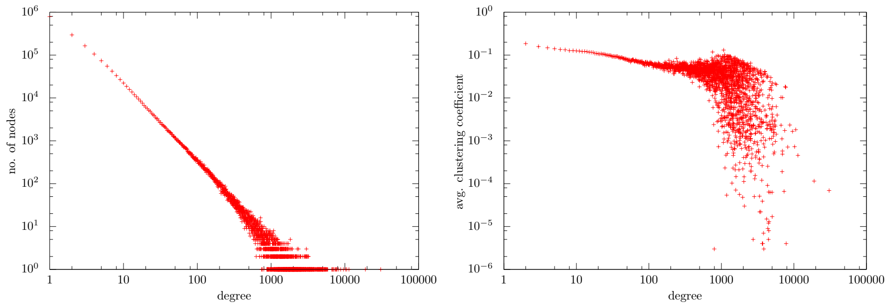


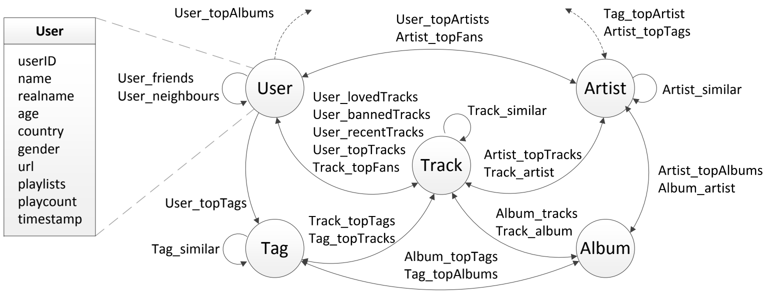
Fig. 1: (a) Degree distribution (left), (b) Avg. cc-degree distribution (right)

distribution and clustering coefficient [9, 19] as it tends to visit nodes of high degree to the detriment of nodes with lower degree. As a result, average degree is overestimated while clustering coefficient is underestimated since high degree nodes are characterized by a low clustering coefficient [11, 19]. This issue will be addressed for the final benchmark dataset with more sophisticated crawling techniques, in particular a modified *Metropolis-Hasting Walk* as proposed in [5, 6] that corrects the bias directly during the walk.

Overall, the skewed degree distribution (cf. Figure 1 (a)), average clustering coefficient per degree (cf. Figure 1 (b)) and average path length of 4.2 indicate typical scale free properties of a social network [17]. A more detailed discussion is provided in Appendix A.

2.1 Benchmark Dataset

Our benchmark dataset considers more than only the friendship relationships (cf. Figure 2 for an overview) and will contain several billion RDF triples while retaining typical properties of the underlying social graph from Last.fm.



durch

Fig. 2: Schema of our Last.fm benchmark dataset

To transform the obtained social graph into an RDF graph it is crucial to define an ontology for the schema shown in Figure 2. While some entities and relations are easy to define using existing vocabularies like the FOAF-Ontology[2], others require the introduction of new ones. An excerpt of the resulting RDF graph is shown in the following:

```
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .
@prefix lb:    <http://example/lastfmbenchmark/> .
lb:user1    a               foaf:Person, lb:User ;
            foaf:age        "30" ;
            lb:lovedTrack   lb:track1, lb:track2, lb:track3, lb:track4 .
lb:track1   a               lb:Track ;
            lb:artist       lb:artist1 ;
            lb:topFan       lb:user1, lb:user2, lb:user3 .
```

An important aspect for RDF benchmarks is data *structuredness* as available RDF datasets have a highly varying structure in contrast to the strongly structured relational data model. However, most existing benchmarks also exhibit a high level of structuredness, similar to relational data, that is practically fixed, i.e. scaling the size of the dataset has no real influence [4]. Since we agree that an RDF triple store should be tested against heterogeneously structured datasets, we envision to use the benchmark generator described in [4] to downsize the overall dataset such that it is not only possible to vary the size of the dataset but also the desired level of structuredness. In contrast, increasing the dataset artificially by means of collected statistics is not considered since it contradicts our idea of a benchmark based on real-world social network data.

3 Last.fm Benchmark Queries

SPARQL is the W3C recommended query language for RDF. With the proposed recommendation for SPARQL 1.1 [8], the W3C addresses the lack of important features ranging from intuitive navigational queries of arbitrary length via some (limited) interference possibilities to complex matchings with support for subqueries, aggregation and negation. The importance of efficient and comprehensive support for these kind of queries justifies the endeavour for an appropriate benchmark geared towards comparing and improving the performance of SPARQL 1.1 expressions in current RDF triple stores.

The following example queries are only intended to illustrate how to exploit SPARQL 1.1 features for exploring interesting graph properties in an intuitive and easy manner within our Last.fm benchmark dataset[3].

A. Find all people that are connected to a user via an arbitrary FOAF distance.
According to [1], the evaluation of this query might show poor performance using the SPARQL 1.1 specification from 2011. Recent changes to the property path specification adopt the idea of a (non-counting) semantics as proposed in [1] that can be evaluated more efficiently.

```
SELECT DISTINCT ?name
WHERE { ?userA foaf:name %username% . ?userA (foaf:knows)* ?userB . ?userB foaf:name ?name
        FILTER (?userA != ?userB) }
```

[2] http://www.foaf-project.org/

[3] Placeholders are indicated by leading and trailing "%".

B. What are the top-k track recommendations for a user based on the listening history of his friends? The ranking considers all kind of tracks of a friend but excludes already known tracks. This query exploits the capabilities of expressing negation, alternative property paths and aggregation in SPARQL.

```
SELECT ?track count(*) as ?playcount
WHERE { ?userA foaf:name %username% . ?userA foaf:knows ?userB .
        ?userB (lb:recentTrack | lb:lovedTrack | lb:topTracks) ?track .
        MINUS { ?userA (lb:recentTrack | lb:bannedTrack | lb:lovedTrack | lb:topTrack) ?track }}
GROUP BY ?track
ORDER BY DESC(?playcount)
LIMIT %k%
```

C. What are the top-k friends of a user with a common music taste and a maximum FOAF distance of 3? The ranking considers common ”loved” tracks.

```
SELECT ?friend count(?friendLovedTrack) as ?commonTracks
WHERE { ?user foaf:knows/foaf:knows?/foaf:knows? ?friend .
        ?user lb:lovedTrack ?userLovedTrack . ?friend lb:lovedTrack ?friendLovedTrack .
        FILTER (?userLovedTrack = ?friendLovedTrack)
        FILTER (?user != ?friend) }
GROUP BY ?friend
ORDER BY DESC(?commonTracks)
LIMIT %k%
```

D. Who are the most popular artists with an average fan age above 30 years? The popularity of an artist can be estimated by counting the number of users that refer to him as top artist. The query uses a subquery to first determine those artists with an average fan age above 30 years. Then, an inverse path (object to subject) is used to obtain top artists for users to compute the popularity.

```
SELECT ?artist count(?user) as ?noOfFans
WHERE { ?artist ^lb:topArtist ?user .
        { SELECT ?artist
          WHERE { ?artist lb:topFan / foaf:age ?age }
          GROUP by ?artist
          HAVING ( avg(?age) > 30 ) }}
GROUP BY ?artist
ORDER BY ?noOfFans
```

4 Conclusion

Although social networks are an increasingly important source of semantic information, they are hardly covered by existing RDF benchmarks. The most comprehensive approach is the Social Network Intelligence Benchmark (SIB) [3] that is built upon an artificial structure-correlated social graph [14]. However, the authors emphasize that the generated network is not “realistic” in the sense that many more (often unknown) structural correlations can be expected in real social networks. For this reason, we will build our benchmark on real data gathered from the social music network *Last.fm*. Since RDF triple stores should be tested against varying levels of data structuredness, we plan to use the benchmark generator described in [4] such that it is not only possible to downsize the dataset but also to vary the desired level of structuredness.

Extending the rather limited path query expressiveness of SPARQL 1.0, the new features of SPARQL 1.1 allow more sophisticated queries for new application fields ranging from graph analysis to recommender functionalities. Although the efficient support of these features will be a key requirement for modern RDF triple stores, they are currently neglected by most existing RDF benchmarks. Therefore, we will especially design a comprehensive set of benchmark queries as indicated in Section 3 that cover a wide range of SPARQL 1.1 features.

References

1. Arenas, M., Conca, S., Pérez, J.: Counting Beyond a Yottabyte, or how SPARQL 1.1 Property Paths will Prevent Adoption of the Standard. In: WWW. pp. 629–638 (2012)
2. Bizer, C., Schultz, A.: The Berlin SPARQL Benchmark. International Journal on Semantic Web and Information Systems (IJSWIS) 5(2), 1–24 (2009)
3. Boncz, P., Pham, M.D., Erling, O., Mikhailov, I., Rankka, Y.: Social Network Intelligence BenchMark, http://www.w3.org/wiki/Social_Network_Intelligence_BenchMark
4. Duan, S., Kementsietsidis, A., Srinivas, K., Udrea, O.: Apples and Oranges: A Comparison of RDF Benchmarks and Real RDF Datasets. In: SIGMOD. pp. 145–156 (2011)
5. Gjoka, M., Butts, C.T., Kurant, M., Markopoulou, A.: Multigraph Sampling of Online Social Networks. IEEE Journal on Selected Areas in Communications 29(9), 1893–1905 (2011)
6. Gjoka, M., Kurant, M., Butts, C.T., Markopoulou, A.: Walking in Facebook: A Case Study of Unbiased Sampling of OSNs. In: INFOCOM. pp. 2498–2506 (2010)
7. Guo, Y., Pan, Z., Heflin, J.: LUBM: A benchmark for OWL knowledge base systems. Web Semantics: Science, Services and Agents on the World Wide Web 3, 158 – 182 (2005)
8. Harris, S., Seaborne, A., Prud'hommeaux, E.: SPARQL 1.1 Query Language. W3C Proposed Recom. (2008), http://www.w3.org/TR/sparql11-query/
9. Kurant, M., Markopoulou, A., Thiran, P., Thiran, P.: On the bias of BFS (Breadth First Search). In: International Teletraffic Congress. pp. 1–8 (2010)
10. Milgram, S.: The small world problem. Psychology today 2(1), 60–67 (1967)
11. Mislove, A., Marcon, M., Gummadi, P.K., Druschel, P., Bhattacharjee, B., Bhattacharjee, B.: Measurement and analysis of online social networks. In: Internet Measurement Comference. pp. 29–42 (2007)
12. Morsey, M., Lehmann, J., Auer, S., Ngomo, A.C.N.: DBpedia SPARQL Benchmark - Performance Assessment with Real Queries on Real Data. In: International Semantic Web Conference (1). pp. 454–469 (2011)
13. Newman, M.E., Park, J.: Why social networks are different from other types of networks. Physical Review E 68(3), 036122 (2003)
14. Pham, M.D., Boncz, P., Erling, O.: S3G2: A Scalable Structure-Correlated Social Graph Generator. In: Selected Topics in Performance Evaluation and Benchmarking, LNCS, vol. 7755, pp. 156–172. Springer Berlin Heidelberg (2013)
15. Schmidt, M., Hornung, T., Lausen, G., Pinkel, C.: SP2Bench: A SPARQL Performance Benchmark. In: ICDE. pp. 222–233 (2009)
16. Voigt, M., Mitschick, A., Schulz, J.: Yet Another Triple Store Benchmark? Practical Experiences with Real-World Data. In: Proceedings of the 2nd International Workshop on Semantic Digital Archives. vol. 912, pp. 85–94 (2012)
17. Watts, D., Strogatz, S.: The small world problem. Collective Dynamics of Small-World Networks 393, 440–442 (1998)
18. Weaver, J., Tarjan, P.: Facebook Linked Data via the Graph API. Semantic Web Journal http://iospress.metapress.com/content/T2745678826V6422
19. Ye, S., Lang, J., Wu, S.F.: Crawling Online Social Graphs. In: APWeb. pp. 236–242 (2010)

A Analysis

The Last.fm network contains many entities and relationships. For the analysis we focus on *reciprocated* friendship relationships in order to grasp the social aspect. We crawled 1.860.215 users and 13.690.576 friendship relationships using Breadth-First Search (BFS).

Degree distribution is crucial in order to characterize a network as social network. The degree of a node is defined by the number of links incident to a node. On Figure 1a the degree distribution is skewed with the majority of nodes having a low degree while very few nodes have significantly higher degree. This is a typical behaviour of social networks. Clustering coefficient is another important characteristic of social graphs and represents the tendency of nodes to form tight clusters. This metric is defined as the number of links that exist between a node's neighbours divided by the maximum possible links that could exist among a node's neighbours. The clustering coefficient in a social network is higher than in other types of networks [13]. Figure 1b depicts average clustering coefficient with regard to degree. We can observe that low degree nodes demonstrate higher clustering coefficient which means that there is a significant clustering among them. On the other hand, as the number of neighbours increases clustering coefficient drops. These results are consistent with previous research on social networks [11].

Lastly, short paths in the network indicate that nodes are reachable through a small number of hops. The average path length of the network is 4.2 and is even shorter than the expected famous "six degrees of separation" of Milgram's experiment [10, 17]. This surprisingly low average path length is probably influenced by the bias of BFS towards the high degree nodes which tend to reduce distances in the network. Another characteristic of social networks is the largest shortest path, the so called diameter. The network has a diameter of 8 which again is low in comparison with other social networks [11] also probably due to the bias introduced by the crawling technique. The aforementioned characteristics skewed degree distribution, high clustering coefficient and short path lengths are typical social network properties and indicate that the Last.fm network has small world and scale free properties [17] as mentioned in the main analysis part (cf. Section 2).