Towards the Next Generation of Large-Scale Network Archives

Stijn Heldens[1](✉), Ana Varbanescu[2], Wing Lung Ngai[1], Tim Hegeman[1], and Alexandru Iosup[1]

[1] Delft University of Technology, Delft, The Netherlands
s.j.heldens@tudelft.nl
[2] University of Amsterdam, Amsterdam, The Netherlands

Abstract. Both data and computer scientists need graph (network) datasets in the design, comparison, and tuning of important scientific results and practical artifacts. Despite the abundance of data in practice, freely available datasets are usually difficult to access, limited in size and diversity, and are collected in small static archives.

This work presents our vision towards a next generation of graph data archives. Therefore, we formulate six key requirements to guide the design of such archives. We further propose GraphPedia, a prototype architecture that addresses these requirements, and provides a large collection of different graphs, in many different storage formats, rich metadata, advanced searching, and on-demand graph generation. Once the open implementation challenges are resolved, GraphPedia will become a dynamic meeting space for exchanging graphs.

1 Introduction

Data and computer scientists are increasingly using graphs (or networks[1]) datasets [9,14,18] in their work. Relevant graph datasets are useful in developing, comparing, and deploying both algorithms [1,16,21], whose results lead to data-driven decisions, and systems [8,10,19,23,25], that can execute these graph processing algorithms.

Despite the existence of large amount of data in practice, researchers have limited access to highly diverse and large-scale graphs. In fact, many research contributions in data and computer science are validated on just a handful of graphs [17,18,22] from a limited set of repositories or, in worse cases, on non-public datasets. This limits the credibility and reproducibility of results, and thus the quality of the scientific and practical processes.

There are many reasons for this sparsity of publicly available graph datasets. Some are technical, as researchers might lack the expertise or budget to set up an infrastructure to allow others to access their datasets. Some are organizational, as researchers from one field of study might not come into contact with other domains. Some are just inconveniences, like the lack of a universal storage format

[1] Throughout this paper, the terms *graph* and *network* have the same meaning and are used interchangeably.

F. Desprez et al. (Eds.): Euro-Par 2016 Workshops, LNCS 10104, pp. 571–579, 2017.
DOI: 10.1007/978-3-319-58943-5_46

for graphs which results in different incompatible formats [2,12,15] being used by different researchers.

A small number of archives do offer graph datasets: SNAP [15], KONECT [13], and UFSMC [5]. However, they represent an outdated generation of network archives, lacking the large-scale and diversity of needed to address the quickly changing needs of today's graph producers and consumers.

We believe that this problem is hindering research in graph processing and propose to build the next generation of graph archives. By drawing inspiration from the few existing archives, in this article we propose a new type of graph archive. Our contribution is threefold:

1. We identify a set of six requirements for next-generation network archives (Sect. 3). The requirements focus on a new type of archive: dynamic, with enhanced search and content generation, provenance and impact analysis, and support for user-content.
2. We introduce an archive following this new approach, GraphPedia (Sect. 4). We discuss the key features of our design, and how they match the requirements.
3. We discuss the main research challenges that lay ahead before GraphPedia can be fully implemented in practice (Sect. 5). We discuss in particular here issues of efficiency and convenience (e.g., data formats), community building, and supporting emerging trends in graph processing.

2 State-of-the-Art in Network Datasets

In practice, there are two ways in which researchers obtain graph datasets: download them from public graph repositories or create them using synthetic graph generators. We discuss both options in this section.

2.1 Real-World Network Repositories

We survey five major repositories for real-world graphs which are publicly accessible (Table 1), sorted chronologically by the year of their establishment. All these network archives share (at least) four significant drawbacks that characterize the state-of-the-art. First, the archives are small (less than 300 datasets, except UFSMC), static, and manually managed. They do not encourage their users to add new datasets to expand the collection. WEBSCOPE even requires an account which needs to manually approved before given access to the datasets. Second, the archives only provides datasets in specific storage formats. SNAP and KONECT offer only the edge list format, GTA stores dataset in their custom GTF format, while UFSMC offers three matrix formats. Third, the archives do not perform impact analysis that indicates how the datasets are used in research, with the exception of WEBSCOPE which requests explicitly attribution for usage. Fourth, it is difficult to select specific datasets by filtering and searching. While UFMSMC does offer a stand-alone Java program to browse through the datasets, the datasets are not categorized but rather loosely sorted by source.

Table 1. Five major repositories for real-world graphs datasets.

	Maintainer	Established	#Datasets	#Formats	Domains	Statistics
SNAP [15]	Stanford Univ	2005	Small ($\sim$ 100)	1	Various (16)	Basic
UFSMC [5]	Univ. of Florida	2011	Medium (2757)	3	Various	Comprehensive
GTA [9]	TU Delft	2012	Tiny (15)	1	Gaming	Description only
KONECT [13]	Univ. of Koblenz	2014	Small (253)	1	Various (23)	Comprehensive
WEBSCOPE [26]	Yahoo Labs	2016	Tiny (8)	1	Web	None

It is our goal to propose next generation archives that alleviate all these problems by design.

2.2 Synthetic Network Generators

Synthetic network generators are designed to enable graph generation based on users' input. Many graph generators emerged in the past [3], with the explicit goal of testing the correctness and scalability of graph processing algorithms.

For example, random graphs are generated by picking pairs of vertices under some random probability distribution and then connecting them by edges. Using a uniform probability leads to the well-known Erdős-Rényi model [6].

Because random graphs do not reflect the characteristics of real-world networks, more realistic generators have been proposed. For example, LDBC DATAGEN [7] generates large-scale social networks, R-MAT [20] generates scale-free networks, and the Internet Graph Generator [24] produces "World Wide Web"-like graphs.

Although most generators are publicly available, they are usually significantly limited in efficiency and usability: processing time is often prohibitive and deployement is non-trivial. Furthermore, the generated graphs or used parameters (e.g., seed) are rarely archived, making experiments difficult to reproduce and expand. These limitations forced existing archives to ignore synthetic graphs. Our goal is to alleviate these issues and incorporate synthetic graphs into next generation archives.

3 Requirements for Next-Generation Network Archives

Based on the observations listed in Sect. 2, we define a number of essential requirements for a next generation network archive.

(R1) **Variety.** Graphs from different domains have different properties and characteristics, which impacts the performance of graph algorithm and systems. It is important that the archive includes many types of graphs and reflects the variety of datasets in the real-world.

(R2) **Encourage Sharing.** An archive should not just be a static collection of datasets, it should be a meeting space for researchers to exchange both knowledge and data. The archive must provide the means for this collaboration.

(R3) **Different storage formats.** A universal storage format for graphs does not exist and many different formats are used in practice. Converting between formats is not always trivial, and this inconvenience can limit users' choices. An archive should offer as many popular graph formats as possible.

(R4) **Usability.** The archive should not be an enumeration of available datasets without any context. Instead, we envision an interactive system that allows users to browse and search the large collection of available datasets.

(R5) **Synthetic datasets.** The archive should provide access to synthetic datasets, even created based on users' demands. Although many generators are publicly available, deploying and using them correctly and efficiently is not always trivial.

(R6) **Provenance and impact.** An archive should mention where the datasets originate from (*provenance*) and how they are used in research (*impact*). This allows users to assess the value of a dataset, and enables the community to report relevant results.

4 Design of GraphPedia

In this section, we present the design of GraphPedia as the first representative of the new generation of graph archives. We explain how GraphPedia addresses the requirements listed in Sect. 3, which ultimately define its architecture.

4.1 Data Model of Graphs

To address requirement (R1), GraphPedia uses a generic data model that can be used to represent many different types of graphs. Each graph consists of a set of vertices, each uniquely identified by an integer, and a set of edges, each consisting of the identifiers of its endpoints. Edges can either be *directed* (i.e., edges are uni-directional) or *undirected* (i.e., edges are bidirectional). Multiple edges are allowed between two vertices, thus enabling multi-edge graphs. Additionally, both vertices and edges can have a list of named properties to store data such as timestamps (temporal graphs), weights (weighted graphs), or labels (bipartite graphs). A similar data model is used in the Graphalytics benchmark [11].

4.2 Virtual Meeting Space

To address requirements (R1) and (R2), GraphPedia allows its users to add new datasets to the archive. These datasets are added after (semi-)automated validation by a GraphPedia moderator, to avoid storing incorrect, irrelevant, or simply duplicate datasets. The possibility to share graphs benefits both the contributors and the users of the archive. Contributors benefit since it helps them gain recognition of their work and it allows them to share knowledge with peers. Users benefit because continuously extending the archive increases both the volume and the variety of the archive over time. Ideally, this dynamic interaction will also enable interdisciplinary interactions.

4.3 Storage Formats

To tackle requirement (R3), GraphPedia enables access to every dataset in many different storage formats. Datasets are internally stored *once* using a single unified format. Whenever a user requests a different format, the dataset is *either* retrieved from a cache, *or* it is being converted into the appropriate format on-the-fly. This approach keeps the required storage capacity under control, while potentially offering large number of formats.

4.4 Network Metrics

To address requirement (R4), GraphPedia presents many graph metrics for each dataset [4]. Users can therefore quickly gain valuable insight, and decide whether a dataset is useful for their application.

Overall, metrics can be classified into three categories.

- *Basic metrics* describe the basic struture and are light-weight. Examples are number of vertices, density, average degree, and number of components.
- *Complex metrics* describe more complex characteristics. Examples are the average clustering coefficient, spectral norm, diameter, and Lorenz curve.
- *Property metrics* describe the distribution of the vertex/edge properties. Basic statistics can be given for these properties, such as mean, minimum/maximum, and standard deviation.

Clearly, an initial selection of metrics to offer needs to be made, but the design must be flexible enough to add more such metrics on-demand.

4.5 User-Interface

Also addressing requirement (R4), GraphPedia allows users to quickly select the relevant datasets *for their application*. This is an essential feature, since the archive grows over time (e.g., due to user contributions, but not only). GraphPedia will include advanced searching to allow users to select, filter, and sort datasets based on their domain, description, and graph characteristics. Note that the graph metrics play a fundamental role here, since they enable characteristics-driven search within the archive.

4.6 Generated Graphs

To cover requirement (R5), GraphPedia does not only offer static real-world datasets, but also provides a service to generate synthetic graphs. Multiple graph generators will be integrated into GraphPedia. Users can obtain a synthetic graph by specifying the type of generator and the corresponding parameters. If this graph is already present in the archive *or* in its cache, it can be downloaded immediately. Otherwise, the graph will be generated and cached. Once a graph is demanded multiple times, a GraphPedia moderator will decide whether it should be made a permanent member of the archive.

In addition to traditional generators, GraphPedia must also offer the ability to "replicate" an existing real-world graph at different scale. Thus, users can "shrink" a graph that is appropriate but too large, or "expand" a small real-world graph to a larger scale, for example to test functionality or study performance at different scales.

4.7 Provenance and Impact

An added value of a centralized archive is the possiblity to study provenance and impact of its items. The archive should contain, as much as possible, datasets with a full "pedigree": source of data, time of collection, extraction procedure, etc. For users providing new data, evidence must be provided (publications, lab reports, raw data, etc.) for the ownership and open nature of the data. This provenance meta-data will be published (annonymized if needed) togehter with the data. In terms of impact, the archive should list, for every dataset, the publications that use it. This information enables researchers to assess how often datasets are used, and which research communities favor the use of particular graphs. It also facilitates a fair comparison and the reproducibility of experimental results.

4.8 Architecture

Figure 1 depicts a high-level overview of the GraphPedia architecture. Users can access the archive via its web-based *frontend* (1). The *backend* of the architecture consists of a number of components: a database for datasets' meta-data (2), separate storage, cached, for the raw datasets themselves (3) (implemented, for example, as a fast (distributed) file system), and a processing platform to handle the processing jobs (4) (e.g., format conversion, synthetic graph generation, or metrics computation).

The web-interface provides four actions: *search* datasets, *download* datasets, *upload* datasets, and *generate* datasets. Searching for datasets is performed using the data from the meta-data storage. When downloading a graph, the *format converter* (5) fetches the raw dataset from the dataset storage, and converts it

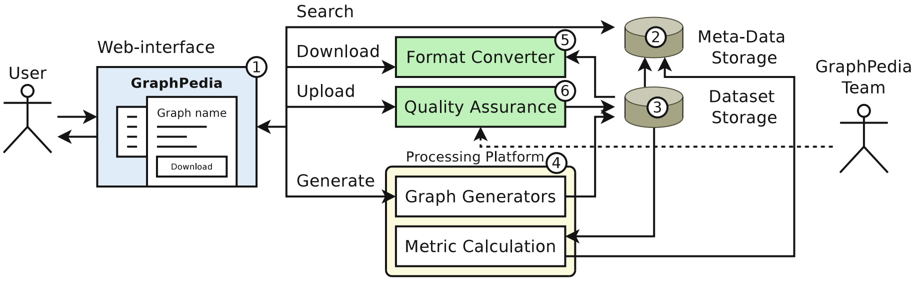

Fig. 1. High-level overview of GraphPedia architecture.

into the appropriate format. We note that for most edge-based formats, this can be done in a streaming fashion, reducing storage requirement and conversion time. When uploading a graph, the graph is submitted to *quality assurance* (6) for approval. Once the submission is successfully checked and approved, it is added to the dataset storage. When generating a graph, a new job is submitted to the processing platform and the resulting graph is added to storage once the job completes. For every new dataset (both uploaded and generated graphs), its meta-data is also added to the meta-data storage. Additionally, jobs are submitted to the processing platform to calculate graph metrics. The results are also saved in the meta-data storage.

5 Open Challenges

There are several challenges in realizing the implementation of GraphPedia.

First challenge is **efficiency**. Since GraphPedia is not just a static collection of datasets, but its provides interactive services that can convert and generate graphs, efficiency is essential to successfully build a large, diverse, dynamic, yet still usable archive. For example, a single user that submits very large conversion and generation jobs, should not prevent other users from submitting smaller jobs. The same job submitted frequently should avoid repeated reprocessing, but its results should be cached and reused More research is required to design and deploy solutions that enable (and measure) the overall efficiency of such a system.

Second, to measure the **impact** of different datasets, GraphPedia must discover all publications that use each dataset. Finding these publications manually is virtually impossible. Thus, automated tools are needed to periodically scan all relevant published work, eventually extracting the ones that use the GraphPedia datasets. Research in information retrieval is required to build this tool.

Third, the topic of **licensing** needs to be thoroughly studied. It cannot be assumed that all datasets are in the Public Domain and users should be able choose a suitable license for their work, which must be respected by the archive and its users.

Finally, although synthetic graph generation is a well-studied research topic, **shrinking** and **expanding** are less known. Research is required to find efficient techniques and tools that can be integrated into GraphPedia while preserving its efficiency. Alternative approaches, such as generating new graphs to mimic existing graphs following non-standard distributions, also require additional research. In particular, capturing and reproducing accurately the characteristics of any type of graph is still an open challenge.

6 Conclusion

Relevant network datasets are increasingly needed, both by data scientists developing and deploying methods to extract meaningful information and by computer scientists developing and tuning systems that enable processing diverse and large-scale network data. Addressing this need, our work proposes GraphPedia, a next generation archive for network data.

Key to our design, we do not see GraphPedia as a static collection of datasets, but as a virtual meeting space that allows researchers to meet and share their data. Additionally, GraphPedia offers many novel features such as rich metadata, advanced searching and filtering, different storage formats, and synthetic graphs on-demand. Overall, GraphPedia will benefit many different research communities, including graph algorithm designers, graph system researchers, and performance engineers.

We are currently tackling practical concerns in implementing GraphPedia, including increasing efficiency and providing a variety of graph storage formats. We will further focus on maintaining the community and continuously supporting emerging topics in graphs.

References

1. Bader, D.A., Kintali, S., Madduri, K., Mihail, M.: Approximating betweenness centrality. In: Bonato, A., Chung, F.R.K. (eds.) WAW 2007. LNCS, vol. 4863, pp. 124–137. Springer, Heidelberg (2007). doi:10.1007/978-3-540-77004-6_10
2. Brandes, U., Eiglsperger, M., Lerner, J., Pich, C.: Graph markup language (GraphML). Citeseer (2010)
3. Chakrabarti, D., Faloutsos, C.: Graph mining: laws, generators, and algorithms. ACM Comput. Surv. **38**(1), 2 (2006)
4. Chebotarev, P.: Studying new classes of graph metrics. In: Nielsen, F., Barbaresco, F. (eds.) GSI 2013. LNCS, vol. 8085, pp. 207–214. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40020-9_21
5. Davis, T.A., Hu, Y.: The University of Florida sparse matrix collection. ACM Trans. Math. Softw. (TOMS) **38**(1), 1 (2011)
6. Erdős, P., Rényi, A.: On random graphs i. Publ. Math. Debrecen **6**, 290–297 (1959)
7. Erling, O., Averbuch, A., Larriba-Pey, J., Chafi, H., Gubichev, A., Prat, A., Pham, M.D., Boncz, P.: The LDBC social network benchmark: interactive workload. In: SIGMOD International Conference on Management of Data. ACM (2015)
8. Gonzalez, J.E., Low, Y., Gu, H., Bickson, D., Guestrin, C.: Powergraph: distributed graph-parallel computation on natural graphs. In: USENIX Symposium on Operating Systems Design and Implementation (2012)
9. Guo, Y., Iosup, A.: The game trace archive. In: 11th Annual Workshop on Network and Systems Support for Games (NetGames) (2012)
10. Hong, S., Depner, S., Manhardt, T., Van Der Lugt, J., Verstraaten, M., Chafi, H.: PGX.D: a fast distributed graph processing engine. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM (2015)
11. Iosup, A., Hegeman, T., Ngai, W., Heldens, S., Prat, A., Manhardt, T., Chafi, H., Capota, M., Sundaram, N., Anderson, M., et al.: LDBC graphalytics: a benchmark for large-scale graph analysis on parallel and distributed platforms. Proc. VLDB Endow. **9**(12), 1317–1328 (2016)
12. Klyne, G., Carroll, J.J.: Resource description framework (RDF): concepts and abstract syntax. Technical report, W3C (2006). http://www.w3.org/TR/rdf-concepts/