Centrality Measures on Big Graphs: Exact, Approximated, and Distributed Algorithms

Francesco Bonchi
ISI Foundation
Turin, Italy
francescobonchi@acm.org

Gianmarco De Francisci Morales
Qatar Computing Research Institute
Doha, Qatar
gdfm@acm.org

Matteo Riondato*
Two Sigma Investments
New York, NY, USA
matteo@twosigma.com

ABSTRACT

Centrality measures allow to measure the relative importance of a node or an edge in a graph w.r.t. other nodes or edges. Several measures of centrality have been developed in the literature to capture different aspects of the informal concept of importance, and algorithms for these different measures have been proposed. In this tutorial, we survey the different definitions of centrality measures and the algorithms to compute them. We start from the most common measures, such as closeness centrality and betweenness centrality, and move to more complex ones such as spanning-edge centrality. In our presentation, we begin from exact algorithms and then progress to approximation algorithms, including sampling-based ones, and to highly-scalable MapReduce algorithms for huge graphs, both for exact computation and for keeping the measures up-to-date on dynamic graphs where edges are inserted or removed over time. Our goal is to show how advanced algorithmic techniques and scalable systems can be used to obtain efficient algorithms for an important graph mining task, and to encourage research in the area by highlighting open problems and possible directions.

Keywords

Centrality; Betweenness; Closeness; Tutorial

1. INTRODUCTION

Identifying the "important" nodes or edges in a graph is a fundamental task in network analysis, with many applications, from economics and biology to security and sociology. Several measures, known as *centrality indices*, have been proposed over the years, formalizing the concept of importance in different ways [16]. Centrality measures rely on graph properties to quantify importance. For example, betweenness centrality, one of the most commonly used centrality indices, counts the fraction of shortest paths going through a node, while the closeness centrality of a node is the average sum of the inverse of the distance to other nodes. Other centrality measures use eigenvectors, random walks, degrees, or more complex properties. For instance, the PageRank index of a node is a centrality measure, and centrality measures for sets of nodes have also been defined.

With the proliferation of huge networks with millions of nodes and billions of edges, the importance of having scalable algorithms for computing centrality indices has become more and more evident, and a number of contributions have been recently proposed, ranging from heuristics that perform extremely well in practice to approximation algorithms offering strong probabilistic guarantees, to scalable algorithms for the MapReduce platform. Moreover, the dynamic nature of many networks, i.e., the addition and removal of nodes or edges over time, dictates the need to keep the computed values of centrality up-to-date as the graph changes. These challenging problems have enjoyed enormous interest from the research community, with many relevant contributions proposed recently to tackle them.

Our tutorial presents, in a unified framework, some of the many measures of centrality, and discusses the algorithms to compute them, both in an exact and in an approximate way, both in-memory and in a distributed fashion in MapReduce. The goal of this unified presentation is to ease the comparison between different measures of centrality, the different quality guarantees offered by approximation algorithms, and the different trade-offs and scalability behaviors characterizing distributed algorithms. We believe this unity of presentation is beneficial both for newcomers and for experienced researchers in the field, who will be exposes to the material from a coherent point of view.

The graph analyst can now choose among a huge number of centrality indices, from the well-established ones originally developed in sociology, to the ones more recently introduced ones that capture other aspects of importance. At the same time, the original algorithms that could handle the relatively small networks for classic social science experiments have been superseded by important algorithmic contributions that exploit modern computational frameworks or obtain fast, high-quality approximations. It is our belief that the long history of centrality measures and the ever-increasing interest from computer scientists in analyzing larger and richer graphs create the need for an all-around organization of both old and new materials, and the desire to satisfy this need inspired us to develop this tutorial.

*Main contact.

WWW'16 Companion, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
http://dx.doi.org/10.1145/2872518.2891063.

2. OUTLINE

The tutorial is structured in three main technical parts, plus a concluding part where we discuss future research directions. All three technical parts will contains both theory and experimental results.

Part I: Definitions and Exact Algorithms.

In this first part, we introduce the different centrality measures, starting from important axioms that a good centrality measure should require. We then discuss the relationship between the different measures, including results highlighting the high correlation between many of them. After having laid these foundations, we move to present the algorithms for the exact computation of centrality measures, both in static and in dynamic graph. We discuss the state-of-the-art by presenting both algorithms with the best worst-case time complexity and heuristics that work extremely well in practice by exploiting different properties of real world graphs.

Exact computation of centrality measures becomes impractical on web-scale networks. Commonly, one of two alternative approaches is taken to speed up the computation: focus on obtaining an *approximation* of the measure of interest or use *parallel and distributed algorithms*. In the second part of our tutorial we explore the former approach, while in the third part we deal with the latter.

Part II: Approximation Algorithms.

Most approximation algorithms for centrality measures use various forms of sampling and more or less sophisticated analysis to derive a sample size sufficient to achieve the desired level of approximation with the desired level of confidence. In this part we present a number of these sampling based algorithms, from simple ones using the Hoeffding inequality to more complex ones using VC-dimension and Rademacher Averages. For each algorithm, we discuss its merits and drawbacks, and highlight the challenges for the algorithm designer. We also discuss the case of maintaining an approximation up-to-date in a dynamic graphs, presenting a number of contributions that recently appeared in the literature.

Part III: Highly-scalable Algorithms.

In the third part of our tutorial, we discuss parallel and distributed algorithms for the computation of centrality measures in static and dynamic graphs. Specifically we present an approach based on GPUs and one based on processing parallel/distributed data streams, together with experimental results.

List of topics with references.

The following is a preliminary list of topics we will cover in each part of the tutorial, with the respective references.

1. **Introduction: definitions and exact algorithms**
   (a) The axioms of centrality [3]
   (b) Definitions of centrality [16], including, but not limited to: betweenness, closeness, degree, eigenvector, harmonic, Katz, absorbing random-walk [14], and spanning-edge centrality [13].
   (c) Betweenness centrality: exact algorithm [4] and heuristically-faster exact algorithms for betweenness centrality [7, 21].
   (d) Exact algorithms for betweenness centrality in a dynamic graph [12, 15, 17].
   (e) Exact algorithms for closeness centrality in a dynamic graph [20].
2. **Approximation algorithms**
   (a) Sampling-based algorithm for closeness centrality [6].
   (b) Betweenness centrality: almost-linear-time approximation algorithm [22], basic sampling-based algorithm [5], refined estimators [8], VC-dimension bounds for betweenness centrality [18].
   (c) Approximation algorithms for betweenness centrality in dynamic graphs [1, 2, 9, 10].
3. **Highly-scalable algorithms**
   (a) GPU-based algorithms [19].
   (b) Exact parallel streaming algorithm for betweenness centrality in a dynamic graph [11].
4. **Challenges and directions for future research**

3. INTENDED AUDIENCE

The tutorial is aimed at researchers interested in the theory and the applications of algorithms for graph mining and social network analysis.

We do not require any specific existing knowledge. The tutorial is designed for an audience of computer scientists who have a general idea of the problems and challenges in graph analysis. We will present the material in such a way that any advanced undergraduate student would be able to productively follow our tutorial and we will actively engage with the audience and adapt our pace and style to ensure that every attendee can benefit from our tutorial. The tutorial starts from the basic definitions and progressively moves to more advanced algorithms, including sampling-based approximation algorithms and MapReduce algorithms, so that it will be of interest both to researchers new to the field and to a more experienced audience.

4. SUPPORT MATERIALS

We developed a mini-website for the tutorial at `http://matteo.rionda.to/centrtutorial/`. It contains the abstract of the tutorial, a detailed outline with short a description of each item of the outline, a full list of references with links to electronic editions, a list of software packages implementing the algorithms, and the slides used in the tutorial presentation.

5. INSTRUCTORS

This tutorial is developed by Francesco Bonchi, Gianmarco De Francisci Morales, and Matteo Riondato. All three instructors will attend the conference.

Francesco Bonchi is Research Leader at the ISI Foundation, Turin, Italy, where he leads the "Algorithmic Data Analytics" group. He is also Scientific Director for Data Mining at Eurecat (Technological Center of Catalunya), Barcelona. Before he was Director of Research at Yahoo Labs in Barcelona, Spain, leading the Web Mining Research group.

His recent research interests include mining query-logs, social networks, and social media, as well as the privacy issues related to mining these kinds of sensible data.

He will be PC Chair of the 16th IEEE International Conference on Data Mining (ICDM 2016) to be held in Barcelona in December 2016. He is member of the ECML PKDD Steering Committee, Associate Editor of the newly created IEEE Transactions on Big Data (TBD), of the IEEE Transactions on Knowledge and Data Engineering (TKDE), the ACM Transactions on Intelligent Systems and Technology (TIST), Knowledge and Information Systems (KAIS), and member of the Editorial Board of Data Mining and Knowledge Discovery (DMKD). He is co-editor of the book "Privacy-Aware Knowledge Discovery: Novel Applications and New Techniques" published by Chapman & Hall/CRC Press.

He presented a tutorial at ACM KDD'14.

Gianmarco De Francisci Morales is a Scientist at QCRI. Previously he worked as a Visiting Scientist at Aalto University in Helsinki, as a Research Scientist at Yahoo Labs in Barcelona, and as a Research Associate at ISTI-CNR in Pisa. He received his Ph.D. in Computer Science and Engineering from the IMT Institute for Advanced Studies of Lucca in 2012. His research focuses on scalable data mining, with an emphasis on Web mining and data-intensive scalable computing systems. He is an active member of the open source community of the Apache Software Foundation, working on the Hadoop ecosystem, and a committer for the Apache Pig project. He is one of the lead developers of Apache SAMOA, an open-source platform for mining big data streams. He commonly serves on the PC of several major conferences in the area of data mining, including WSDM, KDD, CIKM, and WWW. He co-organizes the workshop series on Social News on the Web (SNOW), co-located with the WWW conference. He presented a tutorial on stream mining at IEEE BigData'14.

Matteo Riondato is a Research Scientist in the Labs group at Two Sigma Investments. Previously he was a postdoc at Stanford and at Brown. His dissertation on sampling-based randomized algorithms for data and graph mining received the Best Student Poster Award at SIAM SDM'14. His research focuses on exploiting advanced theory in practical algorithms for time series analysis, pattern mining, and social network analysis. He presented tutorials at ACM KDD'15, ECML PKDD'15, and ACM CIKM'15.

6. REFERENCES

[1] E. Bergamini and H. Meyerhenke. Fully-dynamic approximation of betweenness centrality. *CoRR*, abs/1504.0709 (to appear in ESA'15), Apr. 2015.

[2] E. Bergamini, H. Meyerhenke, and C. L. Staudt. Approximating betweenness centrality in large evolving networks. *CoRR*, abs/1409.6241, Sept. 2014.

[3] P. Boldi and S. Vigna. Axioms for centrality. *Internet Mathematics*, 10(3-4):222–262, 2014.

[4] U. Brandes. A faster algorithm for betweenness centrality. *J. Math. Sociol.*, 25(2):163–177, 2001. doi: 10.1080/0022250X.2001.9990249.

[5] U. Brandes and C. Pich. Centrality estimation in large networks. *Int. J. Bifurcation and Chaos*, 17(7): 2303–2318, 2007. doi: 10.1142/S0218127407018403.

[6] D. Eppstein and J. Wang. Fast approximation of centrality. *J. Graph Algorithms Appl.*, 8(1):39–45, 2004.

[7] D. Erdős, V. Ishakian, A. Bestavros, and E. Terzi. A divide-and-conquer algorithm for betweenness centrality. In *SIAM Data Mining Conf.*, 2015.

[8] R. Geisberger, P. Sanders, and D. Schultes. Better approximation of betweenness centrality. In J. I. Munro and D. Wagner, editors, *Algorithm Eng. & Experiments (ALENEX'08)*, pages 90–100. SIAM, 2008.

[9] T. Hayashi, T. Akiba, and Y. Yoshida. Fully dynamic betweenness centrality maintenance on massive networks. *Proceedings of the VLDB Endowment*, 9(2): 48–59, 2015.

[10] M. Kas, M. Wachs, K. M. Carley, and L. R. Carley. Incremental algorithm for updating betweenness centrality in dynamically growing networks. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, pages 33–40, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2240-9. doi: 10.1145/2492517.2492533. URL `http://doi.acm.org/10.1145/2492517.2492533`.

[11] N. Kourtellis, G. De Francisci Morales, and F. Bonchi. Scalable online betweenness centrality in evolving graphs. *IEEE Trans. Knowl. Data Eng.*, 27(9): 2494–2506, 2015. doi: 10.1109/TKDE.2015.2419666.

[12] M.-J. Lee, J. Lee, J. Y. Park, R. H. Choi, and C.-W. Chung. QUBE: A quick algorithm for updating betweenness centrality. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 351–360, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1229-5. doi: 10.1145/2187836.2187884.

[13] C. Mavroforakis, R. Garcia-Lebron, I. Koutis, and E. Terzi. Spanning edge centrality: Large-scale computation and applications. In *Proceedings of the 24th International Conference on World Wide Web*, pages 732–742. International World Wide Web Conferences Steering Committee, 2015.

[14] C. Mavroforakis, M. Mathioudakis, and A. Gionis. Absorbing random-walk centrality: Theory and algorithms. *arXiv preprint arXiv:1509.02533*, 2015.

[15] M. Nasre, M. Pontecorvi, and V. Ramachandran. Betweenness centrality–incremental and faster. In *Mathematical Foundations of Computer Science 2014*, pages 577–588. Springer, 2014.

[16] M. E. J. Newman. *Networks – An Introduction*. Oxford University Press, 2010.

[17] M. Pontecorvi and V. Ramachandran. A faster algorithm for fully dynamic betweenness centrality. *arXiv preprint arXiv:1506.05783*, 2015.

[18] M. Riondato and E. M. Kornaropoulos. Fast approximation of betweenness centrality through sampling. *Data Mining Knowl. Disc.*, in press, 2015.

[19] A. E. Sarıyüce, K. Kaya, E. Saule, and Ü. V. Çatalyürek. Betweenness centrality on GPUs and heterogeneous architectures. In *Proceedings of the 6th Workshop on General Purpose Processor Using Graphics Processing Units*, pages 76–85. ACM, 2013.

[20] A. E. Sarıyüce, K. Kaya, E. Saule, and Ü. V. Çatalyürek. Incremental algorithms for closeness centrality. In *IEEE International Conference on BigData*, 2013.

[21] A. E. Sarıyüce, E. Saule, K. Kaya, and Ü. V. Çatalyürek. Shattering and compressing networks for betweenness centrality. In *SIAM Data Mining Conf.*, 2013.

[22] Y. Yoshida. Almost linear-time algorithms for adaptive betweenness centrality using hypergraph sketches. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1416–1425, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. doi: 10.1145/2623330.2623626.