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

Proposal for a Half-day Tutorial at ACM WSDM'16

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

Gianmarco De Francisci Morales
Aalto University
Helsinki, Finland
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. Several measures of centrality are available in the literature, each capturing different aspects of the informal concept of importance, paired with several algorithms to compute them.

In this tutorial, we survey the different definitions of centrality measures and the algorithms to compute them. We start from the most common measures (e.g., closeness, betweenness) and move to more complex ones, such as spanning-edge centrality. In our presentation of the algorithms, we begin from exact ones, and progress to approximation algorithms, including sampling-based ones, and to scalable MapReduce algorithms for huge graphs, both for exact computation and for keeping the measures up-to-date on dynamic graphs, where edges change 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 tasks, and to encourage research in the area by highlighting open problems and possible directions.

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 plan to present the material in such a way that any advanced undergraduate student would be able to productively follow our tutorial.

We start from the basic definitions and progressively move to more advanced algorithms, including sampling-based approximation algorithms and MapReduce algorithms. Therefore, our tutorial will be of interest both to researchers new to the field and to a more experienced audience.

Duration: Half-day.

Previous editions of the tutorial

The tutorial was not previously offered. We did not find any tutorial covering similar topics in the programs of recent WSDM conferences and of other top conferences.

*Main contact person

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 Visiting Scientist at Aalto University. Previously he worked as a Research Scientist at Yahoo Labs Barcelona, and as a Research Associate at ISTI-CNR in Pisa. His research focuses on scalable data mining, with an emphasis on Web mining and data-intensive scalable computing systems. He is one of the lead developers of Apache SAMOA, an open-source platform for mining big data streams. 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.

EXTENDED ABSTRACT

Motivation

Identifying the "important" nodes or edges in a graph is a fundamental task in network analysis, with many applications. Many measures, known as *centrality indices*, have been proposed over the years, formalizing the concept of importance in different ways [15]. Centrality measures rely on graph properties to quantify importance. For example,betweenness centrality, one of the most commonly used centrality indices, counts the number 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. The PageRank index of a node is also a centrality measure, and centrality measures for sets of nodes are also possible.

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 and/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 discuss the algorithms to compute them, both in an exact and in an approximate way, both in-memory and in a parallel/distributed fashion for the MapReduce framework of computation. This is done with an effort 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 parallel/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 unified 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 to 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 and/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 is the desire to satisfy this need that inspired us to develop this tutorial.

Outline

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

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

Links to related resources

Previous tutorials presented by the instructors:

Big Data Stream Mining (IEEE BigData'14): `https://sites.google.com/site/bigdatastreamminingtutorial`.

VC-Dimension and Rademacher Averages: from Statistical Learning Theory to Sampling Algorithms (ACM KDD'15, ECML PKDD'15, ACM CIKM'15) `http://bigdata.cs.brown.edu/vctutorial/`.

Correlation Clustering: from Theory to Practice (ACM KDD'14) `http://www.francescobonchi.com/CCtuto_kdd14.pdf`.

Support materials

We are developing a mini-website for the tutorial at `http://matteo.rionda.to/centrtutorial/`. The website will contain the abstract of the tutorial, a more detailed outline with short a description of each item of the outline, a full list of references complete with links to electronic editions, a list of links to software packages implementing the algorithms we present, and naturally the slides used in the tutorial. A preliminary version of the website will be available 15 days after the tutorial is accepted. We plan to work on it and enrich the contents continuously. A preliminary version of the slides will be available 30 days before the conference, or in any case by any deadline given to us by the conference organizers, and the final version will be available 15 days before the conference.

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. doi: 10.1080/15427951.2013.865686. URL `http://dx.doi.org/10.1080/15427951.2013.865686`.

[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] 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`.

[10] N. Kourtellis, G. D. F. 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.

[11] 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.

[12] 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.

[13] C. Mavroforakis, M. Mathioudakis, and A. Gionis. Absorbing random-walk centrality: Theory and algorithms. sep 2015. URL `http://arxiv.org/abs/1509.02533`.

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

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

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

[17] M. Riondato and E. M. Kornaropoulos. Fast approximation of betweenness centrality through sampling. In C. Castillo and D. Metzler, editors, *Proc. 7th ACM Conf. Web Search Data Mining*, WSDM'14. ACM, 2014.

[18] M. Riondato and E. M. Kornaropoulos. Fast approximation of betweenness centrality through sampling. *Data Mining Knowl. Disc.*, (to appear), 2015.

[19] A. E. Sariyüce, 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. Sariyuce, K. Kaya, E. Saule, and U. V. Catalyurek. Incremental algorithms for closeness centrality. In *Big Data, 2013 IEEE International Conference on*, pages 487–492. IEEE, 2013.

[21] A. E. Sarıyüce, E. Saule, K. Kaya, and U. 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.