An Efficient Mapreduce-based System to Find Userlikeness on Social Networks

D. Ravikiran[1] and Dr. S.V.N Srinivasu[2]

[1] Acharya Nagarjuna University

Received: 6 December 2014 Accepted: 4 January 2015 Published: 15 January 2015

Abstract

Day to day Social network information growth pursues an exponential pattern, and Present DB management systems cannot manage efficiently such a huge volume of data. It is essential to employ a ?big data? solution for Social network problems. One of the most important problems in Social network is finding User likeness (ULi). Current methods for finding ULi are not flexible and do not sustain all data sources, nor can them accomplish user necessities for a query tool. In this paper, we propose a reliable and data available method to solve ULi problems over MapReduce design. RiDaULi supports storage and retrieval of all kinds of data sources in an appropriate manner. The dynamic nature of the proposed method helps users to define conditions on all entered fields. Our assessment shows that we can use this method as high confidence in less execution time.

Index terms— social networking, userlikeness, mapreduce, mapper.

1 Introduction

ow a days, with huge volume of user data contraption, common or frequent database management systems cannot effectively sustain data management and analysis in many fields, including meteorology, scientific instruments, social networks, and medical networks. In these and other fields we need a pattern shift to address our problems. Capturing, storing and retrieving information in a timely manner are vital issues in these systems. It is necessary to have available and reliable solutions for these kinds of problems because the prevalent single-node and parallel approaches are far from offering a timely solution. On the other hand, reliable and available resolutions have their own troubles, in particular network bottlenecks, low performance of hardware nodes, and necessities for other nodes' information. Social Network is one of the fields that need reliable and data available solutions, because current solutions cannot properly solve this area's problems. One of the most important problems in this area is identifying user's likeness, or ULi, defined as the rate of likeness between two or more users in terms of their like, interests, personal information, etc. The goal in ULi is to identify those Users who have the greatest amount of information in common in order to use their Preferences or recommendations for new users.

We have two main issues in ULi: the huge amount of information per users; and the fact that most of this data is nonstructured, lacking a predefined record structure that is common among all users. A large number of fields per users may add complexity to ULi problems as well. Given these characteristics, we have to use so-called "big data" solutions. One of the methods which can be used for reliable and data available solutions for big data is MapReduce. MapReduce is used to solve Social Network problems. But MapReduce and other data available solutions have problems such as data locality, network bottlenecks, hardware inefficiency etc. In this paper, we propose RiDaULi, a reliable and data available method for investigating user's likeness. In this method, a MapReduce-based method is used to solve ULi problems. Unlike other approaches, we do not use structured or semi-structured methods for user's information storage. RiDaULi can use different data sources with different data items. Even the same data source can have different data items for two users. Rather, RiDaULi uses a dynamic method to store user's information which can be easily dispersed over hardware nodes. In the proposed method hardware nodes can execute their tasks simultaneously, and none of the nodes needs information from other nodes which is the main problem of MapReduce-based methods. The structure of this paper is as follows. Section 2 investigates some preliminaries concerning MapReduce and Social Network problems. In Section 3, ULi-related literature is discussed. Section 4 focuses on the proposed method. Section 5 presents the evaluation of the proposed method. Section 6 provides the conclusion.

2 Ground Work

In this part, both MapReduce and the relationship between Social Network and big data are explained.

3 a) MapReduce

In this section, the literature related to MapReduce design is discussed, a decomposable algorithm, partitionable data, and sufficient small data partition are the main characteristics required for effective use of MapReduce. In [23], classic MapReduce was optimized to decrease the data transformation load. In the method described in [23], a shared area for information was considered. This type of design is suitable for solving problems, such as k-nn and top k queries. MPI (Message passing interface) was used for message passing in a MapReduce structure. The goal of that paper was to decrease the amount of data transferred in the MapReduce network. A method was developed for tackling workloads in hierarchical MapReduce architectures. Hadoop and uses a deduplication-based snapshot differential algorithm (D-SD) and update propagation. Haloop is another type of MapReduce structure suitable for iterative problems. iMapreduce also supports iterative processes. In [20], HDFS (Hadoop file system) was substituted with a concurrency optimized data storage layer based on the BlobSeer data management service. In [22], a model was presented to estimate I/O behavior of MapReduce applications. In [21], optimization over MapReduce structure was divided into five groups. Fig. 1 shows these groups

4 b) Social Network and big data

In this section, Social Network and its relation to big data are investigated. These days, users' information is generated at an exponential rate. This information has different formats and standards. According to [19], there are various standard data sources, As shown in Fig. 2, huge Volume of information is generated in Various formats with high Velocity; therefore, we have three Vs of Big data in Social Network networks. With ULi there is an additional challenge, namely Veracity, meaning that for many users we typically have doubtful or uncertain information. Social Network problems visible all of the V's, and therefore it is inevitable that we will use big data solutions to solve them but, according to [19], existing big data technologies do not effectively deal with the full spectrum of Social Network problems, so it is necessary to customize them for our purposes. According to high volume of information in Social Network big data is necessary for data analysis .Also costs are reduced by using big data analytics in Social Network. In a userscentered framework is proposed that Can personalize Social Network with a big data driven approach. In **??**35] big data is used to solve problems like the selection of appropriate recommendation paths or improvement of Social Network systems. AITION [37] proposed a reliable knowledge data discovery platform for big data Social Network.

5 Literature on Uli

In this section, literature specifically concerned with ULi is investigated. According to [1], finding ULi solutions can be divided into two parts. Fig. **??** shows this categorization. The first category is solutions that identify ULi relationships by machine learning algorithms [3][4][5]. These types of solutions are offline and they require a long time for the machine learning to take place. Also there are data mining methods which work on streaming data and they can be considered as online data mining methods. These methods can only work on a part of data. In other word they have methods like sliding window, sampling, synopsis etc. over stream data; therefore, this method is not appropriate for ULi problem because we need to analyze all data items **??**40]. The second category uses information retrieval techniques. Some techniques use simple search [6,7]; however, searching over limited keywords within a predefined structure may have severe limitations. Another information retrieval solution involves Using Entity Relationship Graphs (ERG) to investigate similarities between de-fined entities [8,9]. These types of solutions are expensive, and some are not online [8,9]. Some methods try to improve the ERG solution by unified search [10,11]. In [2] MapReduce is used to solve the problem. They tried to reduce algorithm execution time by distributing computation on hardware nodes. PARAMO [36] is a method which uses MapReduce to develop a predictive modeling platform in the Social Network analytics domain. Some methods used LSH [39] (Locality-Sensitive Hashing) for finding similarities **??**31]. In **??**31] LSH and MapReduce are used to extract user's likeness. LSH is not suitable for ULi problem because it works with predefined data structure and with ever changing data sources accuracy will reduced dramatically. According to our investigation, none of the above-mentioned methods are fully effective for solving ULi problems, because of the following considerations: ULi requires a dynamic structure to store users' information. Different users have different data items, and thus require a structure which can store data with different standards and different data formats with no default assumptions.

6 Fig. 3 : Finding user likes

? In the ULi data retrieval phase, the proposed method has to accept all types of input data items and be able to dynamically create queries over all users' data fields. ? ULi implementation time is very important; the method has to implement in a appropriate manner and with high precision. Offline and long-time query execution is not satisfactory. ? Given the huge volume of data generation, distributed solutions are necessary. In this paper we introduce RiDaULi, a reliable and data available method that uses dynamic data structure to store users' data items from data sources with different formats. It can also retrieve data items by dynamic query generation. In this connection our system achieves reliable and data available architecture of RiDaULi, acceptable query execution time is achieved. To the best of our knowledge, RiDaULi is unique in being able to offer a solution to the ULi problem.

IV.

7 Proposed Method

With our proposed method we illustrated RiDaULi is a reliable and data available method which is based on MapReduce. In this method, users' input data is converted to a integrated format as explained below. This adaptation has two main primitive advantages. First, varying in input data does not affect the RiDaULi format; therefore, we can allow any data format without any changes in our format. Second, this format is suitable for MapReduce architecture and helps us to dispense data over nodes. Moreover, each node can do its tasks without the need for other nodes' information.

8 Global Journal of C omp uter S cience and T echnology

Volume XV Issue VII Version I Year 2015

9 (C)

Because of these advantages, we can easily solve ULi problems over distributed nodes. Users' records in various formats can be stored, and efficiency can be achieved by autonomous calculations.

10 a) Data allocation

Because of the unified data format of RiDaULi, data can be distributed over different nodes. Processing power and memory of each hardware node can be important factors to allocate data items to each node.

11 b) Query execution

To execute queries over MapReduce architecture, the queries first have to be converted to an appropriate format for RiDaULi. Then each converted query is sent to the nodes separately for execution, and the RowIDs of the results are returned. Finally, the extracted RowIDs are sent to the Phase 2 Mappers, and users' information is retrieved.

As shown in Fig. **??**, each Phase 1 Mapper sends its results as triples. In the Phase 1 Reducer, aggregation is done on Score based on RowID, and the final Score per RowID is calculated. In the Phase 2 Mapper, other fields with corresponding RowIDs are extracted. The resulting formats of Phase 2 Mappers areas. In Phase 2 Reducer, results of Phase 2 Mappers are aggregated. Also, Phase 1 Reducer results are sent directly to thelikeness Ranker, which sorts RowIDs according to their scores; then, when a RowID is selected by the user, other related information is extracted.

-An Efficient Mapreduce-based System to Find Userlikeness on Social Networks Also to identify equal fields on different data sources it is necessary to have the RiDaULiEqual table. Table **??** shows RiDaULiEqual. Then all rows that are equal to extracted ColumnIDs are retrieved from the RiDaULiFact table. Emit function execute queries and put results into the specified table on the specified server. If the specified table does not exist it creates a table with the specified name. For the Score calculation, many algorithms can be used. Here we use a simple algorithm, in which input users data items are compared with the same data items of existing users. If the data item value of the existing users is exactly equal to the input user's data item value, then its Score is equal to two. Otherwise, if the user's data item value is partially similar to an existing user's data item value, then the Score is equal to one. If there is no likeness between the input data item value and the existing data item values then the Score is equal to zero. In the data sources there are many misspellings, imprecise terms, colloquial terms, etc. To solve these problems we use metadata to create associations between columns. In the Query builder phase, we can define column groups which contain the main term together with its colloquial terms, imprecise terms and prevalent misspellings. When an input column is used in a query, all other Fig. **??** : RiDaULi Process to execute query Group members are considered and their related information is gathered. If there is a bottleneck in the Reducer phase, we remove these via combiners. Fig. 5 shows the RiDaULi architecture with combiners. In this We used data from different Social Network systems, which in turn have different standards for storing data, by Using RiDaULi, we found that we could easily achieve the required results on a reliable and data available structure. As shown in Fig. **??**, twentyone servers were used in Phase 1 and twenty-one for Phase 2. For thirty seven different queries we achieved an average time of 9.42 seconds. As shown in Fig. 5, we then added five combiner servers with the same specifications to each of the two phases, for a total of 52 servers. The average execution time for thirty seven queries improved about 60%, decreasing to 5.65 seconds. and 5. Also we used the LSH algorithm over MapReduce for evaluation. 52 servers with the Table **??** specification were used. For thirty seven different queries we achieved an average time of 63.11 seconds. Fig. 7 shows the results.

12 VI.

13 Conclusion

In this paper, we propose RiDaULi, a reliable and data available method to solve user likeness (ULi) problems over Social network. Previously, the standard methods were based on Machine Learning (ML) or Information Retrieval (IR). ML methods need a long time to execute, and are offline. Standard IR methods have -An Efficient Mapreduce-based System to Find Userlikeness on Social Networks V.

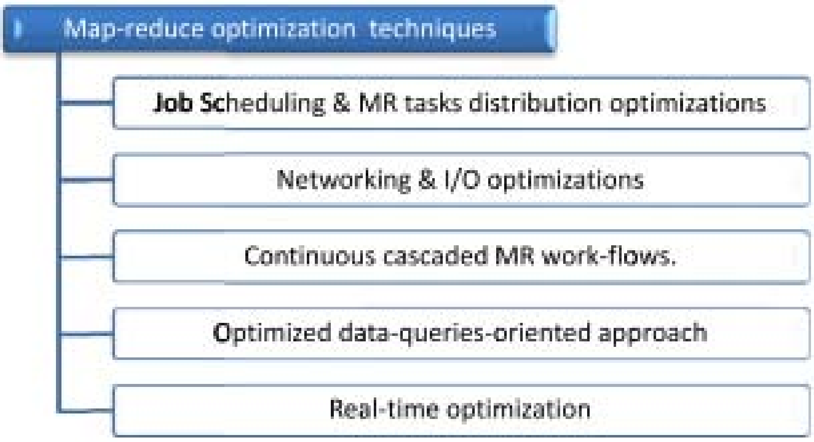
14 Evaluation

In this section we evaluate RiDaULi from two views. First the execution time of the proposed method is evaluated, and second the accuracy of RiDaULi is calculated. As per illustration we producing sample Expected results. many limitations for information storing and query processing; they support only a basic user interface, and limit the kinds of queries that can be built. Online data mining methods have good performance with predefined data sources and are not suitable for dynamic data sources. Also there are some methods like LSH that can properly work over distributed environments but their performances are decreased when there are many changes in input data sources. RiDaULi is an IR method which supports different data formats. All of these formats can be retrieved by data unification. In this method all fields need not be completed, and for each user only the existing fields are entered. This feature allows for data storage size to be considerably reduced. Our evaluation shows that RiDaULi can solve ULi problems effectively. Because of the reliable and data available nature of RiDaULi, it can utilize hardware effectively in order to solve problems involving huge amounts of data.

15 Global

1

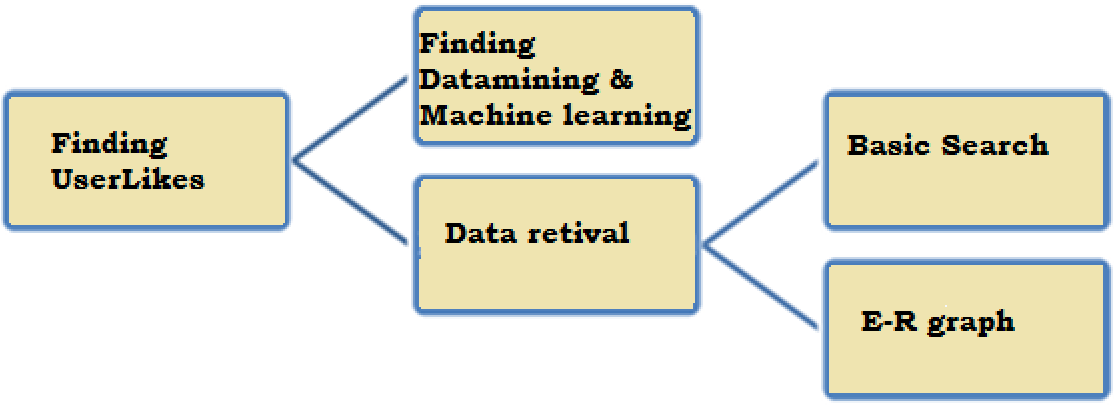
Figure 1: Fig 1 :



2

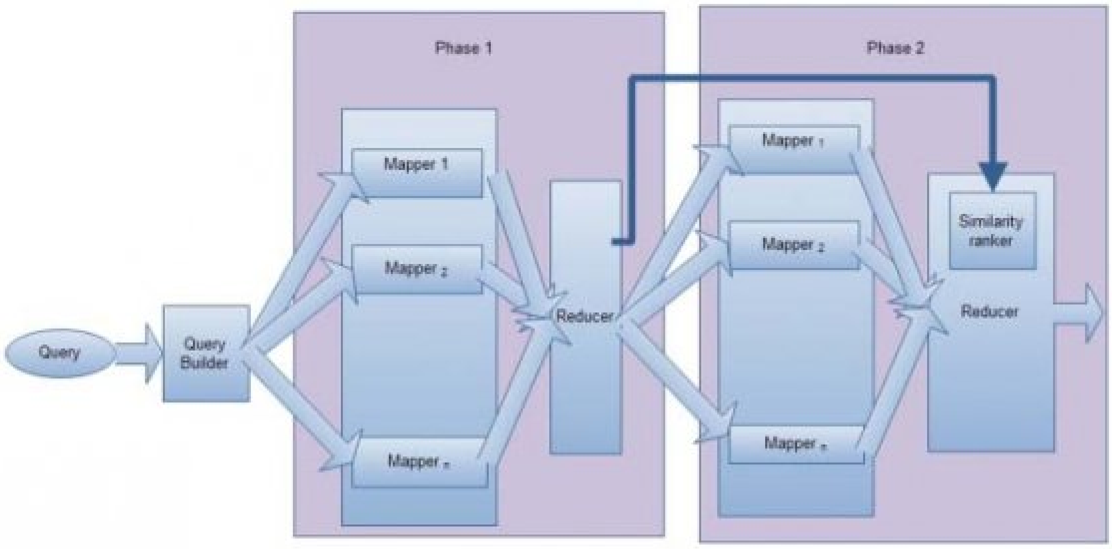
Figure 2: Fig. 2 :

Figure 3: Global



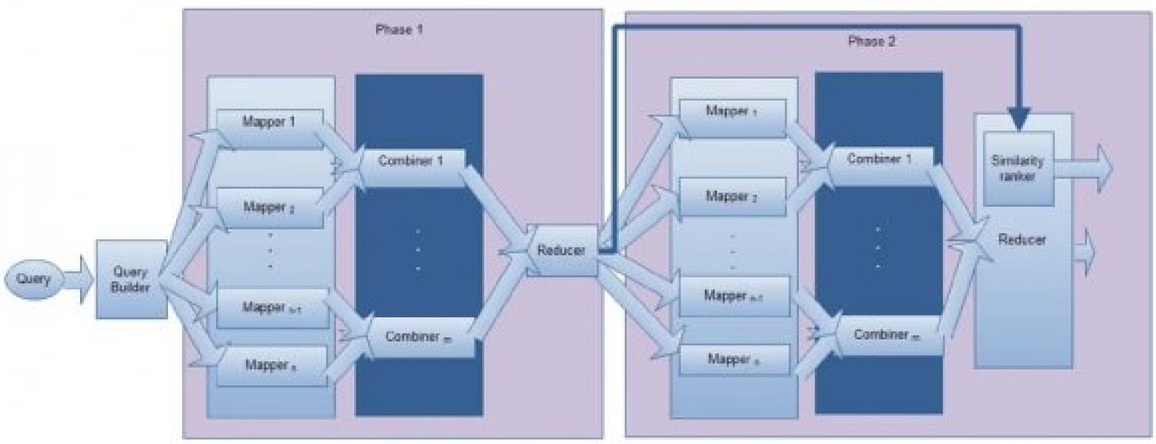
5

Figure 4: Fig. 5 :



6

Figure 5: Fig . 6 :



7

Figure 6: Fig 7 :

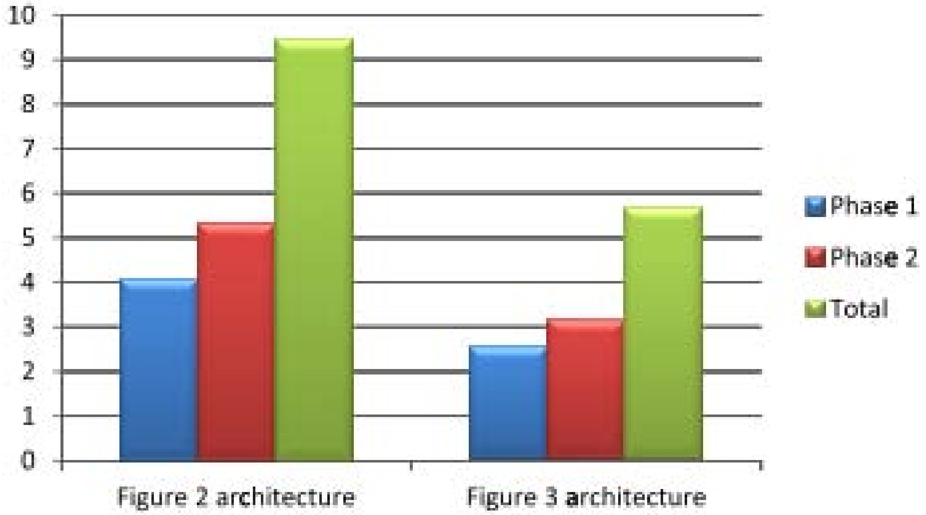


Figure 7:

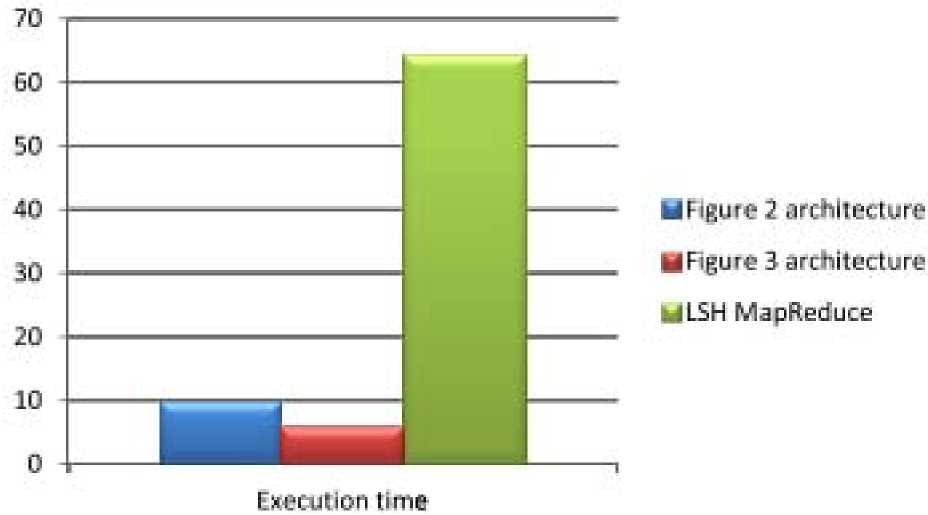


Figure 8:

1

Source ID	Source name
1	Facebook
2	Twitter
3	Linkedin
?.	??.

Figure 9: Table 1 :

2

Id	Name	age	Gender	habits	Likes1	Likes2
1211	sai	20	Male	Reading books	Spiritual	fiction
1212	ram	40	Male	Watching Movies	Action	comedy
1213	seetha	35	Female	Listening Music	melody	devotional

Figure 10: Table 2 :

3

Column Id	Column name	Data Source ID
1	ID	1
2	name	1
3	age	1

Figure 11: Table 3 :

4

Column Id	Row Id	Value
1	1211	sai
2	1211	20
3	1211	male

Figure 12: Table 4 :

1

Figure 13: Table 1 shows

4

has several
advantages:

? Dynamic columns definition

? Completion of all fields is not necessary

? Unified data format

? Data storage size reduction

The proposed data format is suitable for the MapReduce structure, and allows us to execute queries simultaneously on different nodes. There are several steps to Using RiDaULi:

? ETL (Extract/Transform/Load): First, information from different data sources is gathered, and the metadata table (like Table 3) and data table (like Table 4) are created.

GetColumnID function retrieves ColumnID of a specific field from the RiDaULiColumn table. Input parameters are DataSourceID and ColumnName.

Figure 14: Table 4

179 [1] [2] [3]

[1] © 2015 Global Journals Inc. (US) 1

[2] © 2015 Global Journals Inc. (US)

[3] An Efficient Mapreduce-based System to Find Userlikeness on Social Networks © 2015 Global Journals Inc. (US) 1

[Facebook Data] , Tracker Facebook Data . http://www.checkfacebook.com

[The Osn Data and Set] , The Osn Data , Set . http://current.cs.ucsb.edu/facebook/index.html

[Nat. Acad. Sci. USA (2002)] , *Nat. Acad. Sci. USA* Jun. 2002. 99 (12) p. .

[Nguyen et al. ()] 'BAdaptive algorithms for detecting community structure in dynamic social networks'. N P Nguyen , T N Dinh , Y Xuan , M T Thai . *Proc. IEEE INFOCOM*, (IEEE INFOCOM) 2011. p. .

[Kernighan and Lin ()] 'BAn efficient heuristic procedure for partitioning graphs'. B W Kernighan , S Lin . *Bell Syst. Tech. J* 1970. 49 (1) p. .

[Zachary ()] 'BAn information flow model for conflict and fission in small groups'. W W Zachary . *J. Anthropol. Res* 1977. 33 (4) p. .

[Fortunato ()] 'BCommunity detection in graphs'. S Fortunato . *Phys. Rep* 2010. 486 (3-5) p. .

[Girvan and Newman] *BCommunity structure in social and biological networks*, M Girvan , M E J Newman .

[Lin and Schatz ()] 'BDesign patterns for efficient graph algorithms in MapReduce'. J Lin , M Schatz . *Proc. ACM 8th Workshop Mining Learn. Graphs*, (ACM 8th Workshop Mining Learn. Graphs) 2010. p. .

[Newman (2004)] 'BDetecting community structure in networks'. M E J Newman . *Eur. Phys. J. BVCondens. Matter Complex Syst* Mar. 2004. 38 (2) p. .

[Tyler et al. ()] 'BE-mail as spectroscopy: Automated discovery of community structure within organizations'. J R Tyler , D M Wilkinson , B A Huberman . *Inform. Soc* 2005. 21 (2) p. .

[Newman and Girvan (2004)] 'BFinding and evaluating community structure in networks'. M E J Newman , M Girvan . *Phys. Rev. E* Feb. 2004. 69 (2) p. .

[Schaeffer (2007)] 'BGraph clustering'. S E Schaeffer . *Comput. Sci. Rev* Aug. 2007. 1 (1) p. .

[Dean and Ghemawat (2008)] 'BMapReduce: Simplified data processing on large clusters'. J Dean , S Ghemawat . *Commun. ACM* Jan. 2008. 51 (1) p. .

[Mislove et al. (2007)] 'BMeasurement and analysis of online social networks'. A Mislove , M Marcon , K P Gummadi , P Druschel , B Bhattacharjee . *Proc. 7th ACM SIGCOMM Conf. Internet Meas*, (7th ACM SIGCOMM Conf. Internet MeasSan Diego, CA, USA) Oct. 2007. p. .

[Newman (2006)] 'BModularity and community structure in networks'. M E J Newman . *Proc. Nat. Acad. Sci. USA*, (Nat. Acad. Sci. USA) Jun. 2006. 103 p. .

[Brandes et al. (2008)] 'BOn modularity clustering'. U Brandes , D Delling , M Gaertler , R Go¨ Rke , M Hoefer , Z Nikoloski , D Wagner . *IEEE Trans. Knowl. Data Eng* Feb. 2008. 20 (2) p. .

[Brin and Page (1998)] 'BThe anatomy of a large-scale hypertextual Web search engine1'. S Brin , L Page . *Comput. Netw. ISDN Syst* Apr. 1998. 30 (1-7) p. .

[Wilson et al. (2009)] 'BUser interactions in social networks and their implications'. C Wilson , B Boe , A Sala , K P N Puttaswamy , B Y Zhao . *Proc. 4th ACM Eur. Conf. Comput. Syst*, (4th ACM Eur. Conf. Comput. SystNuremberg, Germany) Mar. 2009. p. .

[Rattigan et al. ()] 'BUsing structure indices for efficient approximation of network properties'. M J Rattigan , M Maier , D Jensen . *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, (12th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining) 2006. p. .

[Xue et al. ()] 'BX-RIME: Cloud-based large scale social network analysis'. W Xue , J Shi , B Yang . *Proc. IEEE Int. Conf. Services Comput*, (IEEE Int. Conf. Services Comput) 2010. p. .