Efficient Virtual Social Media for Location Based Services over Mobile Peer-to-Peer Networks

Byoungyup Lee, He Li, Yeonwoo Kim, YosopPae, Hyoungrak Choi,Jongtae Lim, Kyungsoo Bok,Jaesoo Yoo

Abstract Background/Objectives: Recently, there have been done studies on contents sharing and searching among mobile devices on social media.

Methods/Statistical analysis: However, the previous social media are not suitable for mobile peer-to-peer networks due to the high cost of graph management and the low data search success rate. In order to alleviate such problems, an efficient virtual social media for location based services over the mobile peer-to-peer networks is proposed.

Findings: We generally build social relationships by considering similarities between different nodes. The similarities are measured by computing the common interests and the recent positions. In order to reduce the graph management overhead, each peer only manages the peers that have common interests and nearby to it. This reduces graph management overhead and improves navigation performance.

Improvements/Applications: In order to demonstrate the excellence of the virtual social media, the proposed method is compared with conventional techniques in a variety of environments in terms of the retrieval success rate

Keywords: Virtual social media, Mobile user, Peer-to-peer network, Network topology, Location-based Service.

I. INTRODUCTION

The fast evolution of data transmission technologies and smart devices has made the Internet accessible anytime, anywhere. Because most mobile devices have GPS capabilities, mobile users can obtain location information easily. These days, as mobile users increase, mobile social media services are developing rapidly. Location-based mobile retrievals are becoming so critical for users who use them. Unlike conventional Web services, the mobile social retrieval uses user's preference information collected from a variety of social media to find the best results for users. In a mobile environment, how to manage recent location information and how to order retrieval results using positions are important issues. Unlike traditional computing environments, a mobile computing environment needs to see retrieval results on mobile devices that screen sizes are small. As a result, effective refining techniques are required for mobile social retrievals [1-3].

These days, information sharing among social media like YouTube and Twitter are popular. With the advancement of the personal mobile devices and especially the rapid growth of the smart phones, a large of personal data is created every day. Users can use social media services to share data with other users. Using a short range of wireless technologies, self-organizing applications can be performed on ad-hoc networks of mobile phones[4-5]. In general, the nearby users can connect in a dynamic way and create their self-organizing networks. For example, users can share some useful contents in a conference, bar or office.

Recently, with the advance of the personal smart instruments and especially the rapid growth of the smart phones, a large of personal data is created every day. Users can share these data with other users by using the social network services. The availability of short range wireless technology such as WiFi and Bluetooth makes it possible to build a new self-organizing applications running in ad hoc networks of mobile phones[6-10]. In general, a group of users with common interests can relate, and the nearby users can connect in a dynamic way and create their self-organizing networks[11-15]. For example, users can share some useful contents in a conference, bar or office.

In recent years, location based services on mobile ad hoc networks have been focused. Various social media are being utilized for information sharing and exchange on mobile peer-to-peer networks. In [1], it proposed an effective social P2P query processing method by using core words for retrieving resources. But the method adopted the existing transmission techniques for transmitting request messages across the network. And it cannot guarantee the success rate of contents discovery for large scale mobile networks.

In [2], a dynamic social grouping based routing algorithm in mobile Ad-Hoc networks was introduced. It forms a set of social clusters by considering the connection patterns of users. Another method to calculate the user similarity by using the semantics of the locations was proposed in [5]. However, for both these two methods the connection frequency and history

Revised Manuscript Received on May 22, 2019.

[1]**Byoungyup Lee,** Department of Cyber Security, Paichai University, 155-40,Baejae-ro(street), Seo-Gu, Daejeon, 35345, South Korea

[2]**He Li,** School of Software in Xidian University, Xi'an, China,266 Xinglong Section of Xifeng Road, Xi'an, Shaanxi, 710126, China

[3]**Yeonwoo Kim,** [5]**Hyoungrak Choi,** [6]**Jongtae Lim,** [7]**Kyungsoo Bok,** [*8]**Jaesoo Yoo** School of Information and Communication Engineering, Chungbuk National University, 1 Chungdae-ro(street), Seowon-gu, Cheongju, 28644, South Korea

[4]**YosopPae,** Department of Big Data, Chungbuk National University,1 Chungdae-ro(street), Seowon-gu, Cheongju, 28644, South Korea

* E-mail : yjs@chungbuk.ac.kr

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

locations cannot be easily collected in mobile peer-to-peer networks. Additionally, only considering connection frequency or history locations for contents sharing and searching may lead to low efficiency.

[3] presented the information sharing and exchange method based on the node mobility in peer-to-peer environments. The method clustered the moving nodes into a variety of groups by using the interests of the moving nodes. Then, in each group, it chooses the secure node and assigns it as the name node that maintains an index of all data in the group. It also assigns the nodes with high mobility as the transmission nodes. But it requires a big graph management burden as the name servers must keep all data across the network.

In this paper, we propose a location based virtual social media on mobile networks. The proposed scheme creates the social relationship by taking into account the similarities among various nodes. The similarities are measured by computing the common interests and the current position. Each node manages the nodes that have the similar characteristics and nearby to it for decreasing the graph management overhead. By doing this, the proposed scheme reduces the graph management overhead and improves the retrieval success rate. It is shown through various performance evaluations that the proposed method is more efficient than the previous methods.

The remainder of this paper is organized as follows. Section 2 presents the details of the proposed method. Section 3explainsvarious performance evaluation results in order to show the excellence of our virtual social media. Finally, Section 4 summarizes this paper.

II. THE PROPOSED METHOD

2.1. *Characteristics*

Mobile virtual social media is represented as a dynamic graph. It is very important to deal with the dynamic graph efficiently. Dynamic graphs have much common subgraph, causing wasted space due to duplicate storage. We present a new a historical graph management scheme that minimizes the storage of common subgraphs and efficiently access snapshot graphs. The accessibility of past graphs can be ensured while minimizing common subgraph by dividing and managing data into intersection and delta snapshots. An intersection snapshot (IS) is generated with common subgraph according to a common subgraph ratio. The common subgraphs are managed in the intersection snapshot in an integrated manner to reduce storage space wastage and the remaining subgraphs except for the common graphs are stored in the delta snapshot (DS). A single snapshot includes tables of out-going and in-coming vertices. An out-going vertex represents an edge between two vertices by the number of vertices connected with an offset that indicates a starting position in the corresponding in-coming vertex table. Furthermore, the CSR technique is utilized to reduce the amount of information in graph data.

A IS are generated by detecting common subgraphs sequentially by time from the first graph data. A IS is generated for which the Common SubGraph Ratio (CSGR) of the historical graphs is bigger than the reference value. The DS is generated to include subgraphs at each time that is not included in the detected common subgraphs in the graph data up until the time generated by the IS and the generated DS is connected to the corresponding IS. The proposed overall structure is constructed by iterating the above process.

2.2 Intersection snapshot

Dynamic graphs will likely have partial changes over time rather than changes to the entire graph and the subgraphs without changes will be duplicated over time. The common subgraphs wastes storage space in an integrated graph requires ISs. The ISs are individually stored to reduce storage space and search performance. The IS only stores common subgraph within a time interval in contrast with snapshots in existing schemes. Since the generated IS does not store common subgraphs, it can minimize the waste of storage space. Each IS_i stores $<TI_i, NS_i, CSG_i>$. Here, TI_i is a usage time interval $[ST_i, ET_i]$ of IS_i, where the ST_i and ET_i are the first changed time and last changed time of graph G_n connected to IS_i. Equation (1) and (2) are the ST_i and ET_i, where $T(G_n)$ is a changed time of G_n. The NS_i is a the number of historical graphs connected to IS_i and CSG_i is common subgraphs contained in IS_i. If a graph G_n is a first graph connected in IS_i, CSG_i is calculated by Equation (3), where '$\cap$' is an intersection operation that stores common subgraph of graphs.

$$ST_i = MIN_{n=j}^{j+NS_i-1}(T(G_n)) \quad (1)$$

$$ET_i = MAX_{n=j}^{j+NS_i-1}(T(G_n)) \quad (2)$$

$$CSG_i = G_j \cap G_{j+1} \cap G_{j+2} \cap ... \cap G_{j+NS_i-1} \quad (3)$$

To generate the IS, the common subgraphs in the graphs over time are sequentially detected. When there are k consecutive graphs, the common subgraphs and the change histories are analyzed by time. If a CSGR is lower than a threshold value after analysis, the IS for k historical graphs is generated. Otherwise, the CSGR for the next historical graph is calculated. We continuously compare the consecutive graphs, and the Common SubGraph (CSG) and Provenance Information (PI) is stored in a Graph Pattern Table (GPT) over time. The CSG stores a common edges among consecutive graphs. In dynamic graph, vertices and edges are inserted and deleted over time. Originally, provenance are metadata that represent the source information or changing history of data [33-35]. The provenance can be used to track the data changes and usage histories. The proposed scheme should detect the changed subgraph over the time for detect the CSG. Therefore, the proposed scheme uses the provenance to track the update operation of graphs over time. The PI stores the changed information of the current graph through comparison with the previous graph by simplifying the original provenance representation. In PI, each changed information UI_j is represented by $< OP_j, OB_j, T_j >$, where a OP_j is a update operation such as insert (i) and delete (d), a OB_j is a changed object such as vertex and edge, and T_j is a changed time of a OB_j .

2.3 Delta snapshot

If k historical graphs are stored separately during generating IS, it requires a lot of storage space and search cost. The proposed scheme manages k historical graphs as a provenance graph with the PI to support a historical search. However, dynamic graphs are continuously changed and then the size of the provenance graph are continuously increased. Therefore, we generates a DS which stores the subgraphs that are not contained in the IS for the k historical graph and each DS_n is connect to the corresponding IS_i . The DS_n of a graph G_n manages $< T_n, SG_n >$, where a T_n is a changed time of G_n and SG_n is a subgraph of G_n that are not contained in IS_i . The provenance graph stores the original graph as well as the changing history. Since IS_i stores a common subgraph CSG_i , we first remove the CSG_i included in the IS_i from the provenance graph to generates a DS_n . If the G_j is the first graph connected in IS_i , the DS_n is generated by removing all the changed histories. If the G_n is not the first graph connected to the IS_i , the DS_n is generated by reflecting the changing history information. If the PG_i is a provenance graph and the PI_n is the provenance information of graph SG_n , the SG_n is calculated by Equation (5), where ‘ $-$ ’ is an operation that removes certain subgraphs or withdraws update operations, and ‘ $+$ ’ is an operation that reflects the update operations.

$$SG_n = \begin{cases} PG_i - CSG_i - \sum_{n=j}^{n+k-1} PI_n & ,\text{if } SG_n \text{ is the first graph cooneted to } IS_j \\ SG_{n-1} + PI_n & ,\text{otherwise} \end{cases} \quad (4)$$

2.4 Network Representation

Generally, the existence of edges that connect vertices is represented by a sparse matrix in graph data. However, if graph data are represented by a sparse matrix, it requires a matrix size that includes the total number of vertices. Furthermore, most values in a matrix whose edges do not exist are represented by zero. Thus, a form of the CSR representation method is used to store the sparse matrix representation of graph data more efficiently. CSR representation does not represent an unnecessary matrix whose edges are not present but connects only existing data according to the order of rows. The proposed scheme modifies the CSR method to connect snapshots. The existing CSR representation technique has the drawback that it represents all common subgraph in the dynamic environment. The proposed scheme represents graph data inside each snapshot by modifying the CSR representation method. The proposed scheme modifies the CSR technique to reduce the amount of information in graph data that is duplicated in the dynamic environment and connects common subgraph with hierarchical snapshots. The proposed scheme can represent the information of vertices and edges in the internal structure of a snapshot.

2.5 Graph search

The graph search in the proposed scheme is more efficient when for searching histories over a certain period of time, since it manages common subgraph in an integrated manner. For a graph search, the corresponding IS is accessed to read common graphs and DSs in the lower end to access overall graphs. Assuming that the i -th IS and n -th DS are represented by IS_i and DS_n , a graph at time t_n can be expressed by Equation (6). Here, ‘ $\cup$ ’ is a union operation to merge the two graphs.

$$G_n = IS_i \cup DS_n \quad (5)$$

It is an example that accesses G_3 through the overall structure in [Figure 1]. First, whether DS_3 is connected with IS_2 is verified. Once access to IS_2 connected to the DS at the corresponding time is complete, access to DS_3 is accessed to access each vertex in the same manner as in the IS. Since there is no new information about the creation of edges among edges connected to out-going vertex v_1 in the DS, offset (0, 2) of the in-coming vertex table of IS_2 that represents the information of edges connected with vertex v_1 in the out-going vertex table of IS_2 is stored to read the common subgraph. IS_2 refers to the offset (0, 2) that points to the in-coming vertex table v_2 and v_3 connected to v_1 in the out-going vertex table; this represents an edge that is a connection between two vertices. Offset 0 refers to the location of the in-coming vertex table, and offset 2 refers to the number of in-coming vertices connected to out-going vertex v_1 . In the case of the next out-going vertex v_2 of the DS, a new edge e_{26} is generated. Thus, the in-coming vertex table pointed to by vertex v_2 represents vertex v_6 , which indicates new connection information, and the remainder of the common information is indicated by the in-coming vertex table offset of IS_2 and can be accessed. Finally, all information in the DS is read through the above procedure to respond to the historical graph query at the corresponding time.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

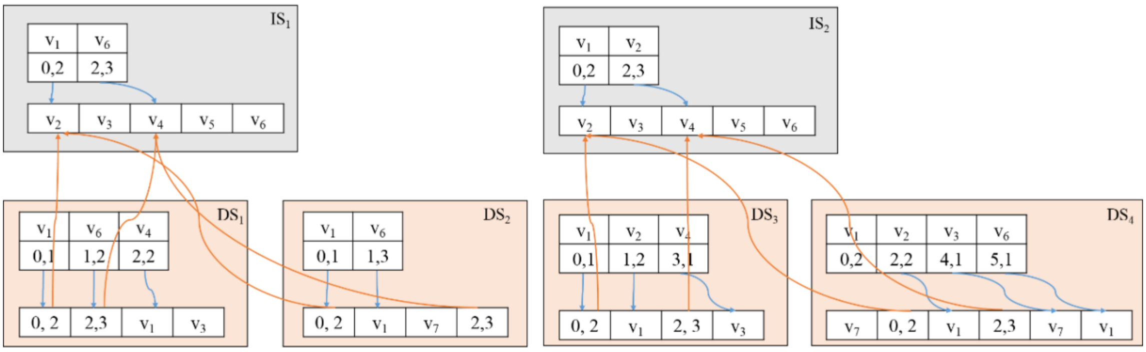

Figure 1. Historical graph search

III. EXPERIMENTAL RESULTS

In order to show the excellence of the proposed scheme, we compare it the MANET environment of the method proposed in [3] in terms of the graph management burden and the retrieval success rate. The performance evaluations were executed in every 400s. We create the moving objects by using the networked-based generator [6]. The experimental space is set to 1,000m 1,000m. Each mobile device has a transmission range of 30 meters. We wrote simulation codes in Java.

The graph management burdens of the proposed virtual social media and the existing technique are first investigated. The management overhead is defined as the change times of the graph structure within any specific experiment time. [Figure2] shows the experimental results that the proposed method outperforms the previous method when moving speeds of the mobile peers vary from 1m/s to 5m/s. It is shown through the experiments that the existing method is not suitable for the mobile peers with fast speed as all of the peers with the similar characteristics across the network should be managed by the directory peers. Since the proposed method adopts the current position and common interest of each peer to form the network, and keeps only the near similar peers, it is very clear that its graph management burden significantly decreases.

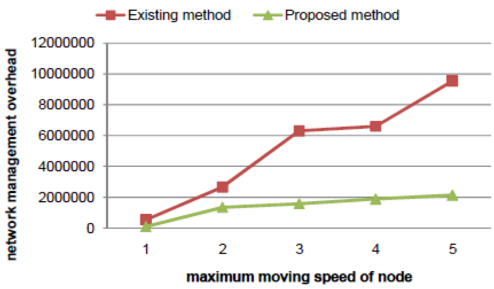

Figure 2. The Graph Processing Overhead

[Figure3] represents the success rates of data searching according to various peer speeds. The success rate of data searching means the ratio of success to total number of quests. In the performance evaluation, we randomly generated 400 queries from the participating mobile devices. The experiments are performed when the speeds of peers are 1m/s, 2m/s, and 3m/s. It is shown through the experimental results that the search success rate is reduced when the average moving speeds of each node increase. The proposed method outperforms the existing method because it considers the current location of each node.

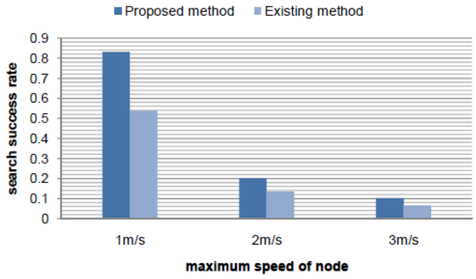

Figure 3. The Search Success Rate

IV. CONCLUSION

This paper has proposed an efficient virtual mobile social network over MANET. The proposed method takes into account both common interests and current positions as social relations. Each peer only manages the peers with common interests and close positions in order to reduce the graph management overhead. And the proposed method accesses the other peers by using friend relationship in the social media. Therefore, our virtual social media decreases graph management burdens and improves the retrieval success rate. It was shown through the various experimental evaluations that the proposed method is more efficient than the conventional methods. In the near future, we will apply the proposed method to real systems.

ACKNOWLEDGMENT

This research was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2017S1A5B8059946) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. NRF-2017R1A2B1003678).

REFERENCES

1. HawR,HongCS,KangCH. A Social P2P Networking Based on Interesting Keywords. Proc. International Conference on Information Networking,2011:509-12.
2. CabanissR,MadriaS,RushG,TrottaA,Vulli SS. Dynamic Social Grouping Based Routing in a Mobile Ad-Hoc Network. Proc. International Conference on Mobile Data Management, 2010:295-6.
3. ZhangH,ShenH. A Social Network based File Sharing System in Mobile Peer to Peer Networks. Proc.International Conference on Computer Communications and Networks, 2009:1-6.
4. LiH, BokK, YooJ. An unstructured Peer to Peer architecture over mobile ad hoc networks.International

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Information Institute, 2012;15:3085-106.
5. LeeMJ,ChungC. A User Similarity Calculation Based on the Location for Social Network Services. Proc.International Conference on Database Systems for Advanced Applications, 2011:38-52.
6. BrinkhoffT.A Framework for Generating Network-based Moving Objects. GeoInformatica, 2002;6(2):153-80.
7. MagalhaesJ,HolandaM. EIKO: A Social Mobile Network for MANET. Proc. Information Systems and Technologies, 2011:15-8.
8. EmreS, OrianaR, PatrickS, GustavoA. Enabling social networking in ad hoc networks of mobile phones. VLDB Endowment, 2009;2(2):1634-9.
9. CarretonAL,Harnie D, Boix EG, Scholliers C, Cutsem TV, Meuter WD. Urbiflock: An experiment in Dynamic Group Management in Pervasive Social Applications. Proc. IEEE International Conference on Pervasive Computing and Communications Workshops, 2010:250- 5.
10. Bae JH, Lee SH, Kim SW, VegaNet: A Peer to Peer Overlay Network for Mobile Social Applications. Proc. IEEE International Symposium on Consumer Electronics, 885-889.
11. Haw R, Hong CS, Kang CH. A social P2P Networking Based on Interesting Keywords. Proc. International Conference on Information Networking, 2011:509-12.
12. Li J,Khan SU. MobiSN: Semantics-based Mobile Ad Hoc Social Network Framework. Proc. IEEE Conference on Global Telecommunications,2009:883-8.
13. Zhang H, Shen H. A social network based file sharing system in Mobile Peer to Peer networks. Proc. International Conference on Computer Communications and Networks, 2009:1-6.
14. Chen K, Shen HY, Zhang HB. Leveraging Social Networks for P2P Content-based File Sharing in Mobile Ad Hoc Networks. Proc. IEEE International Conference on Mobile Ad-Hoc and Sensor Systems, 2011:112- 21.
15. Roy C, Sanjay M, George R, Abbey T, Srinivasa SV. Dynamic Social Grouping Based Routing in a Mobile Ad-Hoc Network. Proc. International Conference on Mobile Data Management, 2010:295-6.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication