May - June 2020
ISSN: 0193-4120 Page No. 4995-4998

Community Detection using Keyword-based Search

[1]L A Lalitha, [2]Ganesh Gowda B C, [3]Harish Kumar SK, [4]B Keerthi Raj, [5]Hruthik C K

[1,2,3,4,5]School of C&IT, Reva University, Bengaluru, India

Article Info
Volume 83
Page Number: 4995-4998
Publication Issue:
May - June 2020

Article History
Article Received:*19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *16 May 2020*

Abstract
Social Networking platforms play a very important role these days in day to day human life. It has become a necessity among people to update everything they do and every place they visit. This action is a mere habit to people, but very essential data for data processing. One such social platform that is trending and provides data in abundance is Twitter. All kinds of data are readily available, it is to be mined and used at user's will. In this paper we discuss the possibility of implementing keyword search in XML, geospatial datasets and also Keyword- Based image search on Twitter data using Louvain Algorithm in an attempt to make the process more efficient.

Keywords: *Clustering, Dataset, Louvain, Keyword-Based search, Twitter*

1. Introduction

Basically, at present we implement an efficient algorithm over a twitter dataset to get expected output from the dataset. Data is formed into a tree containing clusters of data. These clusters are minimized to nodes based on their modularity. As in Louvain algorithm it works based on greedy method. Later on, conditions are implied to sort data according to need. Search, filtering, processing of data is later done through these clusters of data. The idea being proposed in this paper is to implement keyword search to this process. For instance, if we were to filter out Spam Tweets and Spam Accounts, we can list out certain keywords that can help in finding out spam entities. We can search for only those keywords in the dataset. This saves time and resources and also it is seen to be more efficient than the usual method. In our paper we have considered variety of possible datasets that are used such as XML dataset, geospatial dataset and also image annotation that help in keyword search so has to make it more accessible for all kinds of data.

A. Keyword-Based Image Annotation and Search

An Image Annotation is a kind of technique where we can label the image with certain outline and keywords. The purpose of this technique is to make the system recognizable and to train the machine. There are many techniques to describe the contents in image data with the set of figures we can describe the contents present in the image. There is another technique for image annotation which is based on the multi-dimensional model where each dimension is a tree structure taxonomy of concepts called Semantic Tags which is used to describe the image content.

In the scenique we can use Ostia Algorithm by explaining tags and data associated with an image its able to predict a high- quality set of semantic tags to that image. The purpose of using the algorithm is to improve the effectiveness of keyword search technique.

If we apply the image annotation to the community detection in twitter, we can analyze the image tweets which is been tweeted in the Social Media by the scenique technique, where we can retrieve the content from the image data and it can further be used for Community Detection.

Based on the image data which has been tweeted on the twitter, we can cluster the similar content which is retrieved from the data and form the community and then it can be analyzed based on one's interest[6].

B. Keyword search in XML documents

Dataset is a tree where each node associated with multiple keywords. We consider the nearest keyword search on XML documents. The distance between the two nodes as the number of edges in the path linking them. In nearest keyword search we pre-processed the data tree by building a separate distance structure for each unique keyword that appears in data tree. This is reminiscent of

May - June 2020
ISSN: 0193-4120 Page No. 4995-4998

the inverted index, which also has an inverted list dedicated to each keyword.

Instead of simple test, however our structure for keyword is a binary tree constructed in a more sophisticated manner. Nearest keyword queries can serve as the building bricks to tackle some important problems in XML databases and we use nearest keyword search in community detection in twitter.

In this paper our aim is to improve efficiency of path queries that can be processed by keyword search.[2]

2. Working

Implementing keyword search requires proficiency in query languages such as SQL, QUEL, etc. Performing keyword search over dataset is basically performing queries in query languages. These querying methods are then implemented through any data processing tool (such as R language) according to the ease of the programmer. The final product of this collaboration will be the tool to perform keyword search over any dataset. Here we have to also consider the nature of the dataset, as it is not simple as said to implement it on any other dataset available.

The working of this proposed algorithm starts with constructing tree structures on the datasets given. Each node based on its modularity in between each other forms singular nodes by combining for simplicity. This process is repeated until there is no further simpler form that can be achieved. At this stage we perform Community Detection based on our needs by also implementing keyword search over it.

C. The Louvain Algorithm

The Louvain algorithm is based on greedy method. It finds all the nodes that are similar in properties and scores them accordingly. This method of scoring is called as modularity. Based on the modularity score the cluster of nodes are formed to one single node. As such it goes on reducing the nodes into less complicated structure and therefore the huge graph with clustered data is reduced into simpler structure.[7]

Pseudo Code

```
CALL algo.beta.louvain(label:STRING,relationship:STRIN
G,{
write:BOOLEAN,
writeProperty:STRING
})
YIELD
nodes,communities,modularity,loadMillis,computeMillis
,writeMillis
```

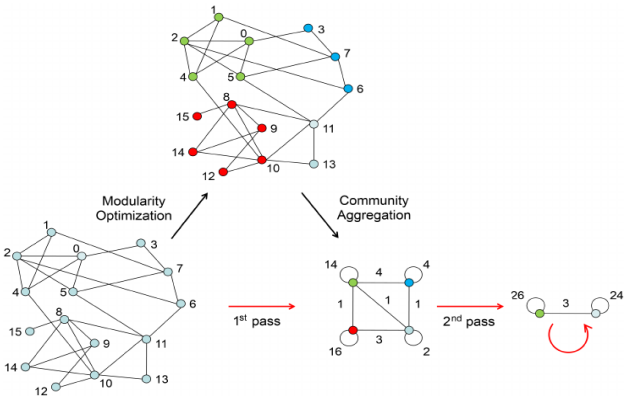


Figure 1: Overview of working of Louvain Algorithm

The above image represents the working of Louvain algorithm. The given dataset is formed into a tree and then based on its modularity the tree graph is optimized. These optimized graphs form into a single node and thus forming multiple communities. Then the communities are aggregated to form even more simpler structure. This loop of process is iterated until a simpler structure is obtained.

Table 1: Parameters of Louvain Algorithm

Name	*Type*	*Default*	*Optional*	*Description*
node label	*String*	*Null*	*yes*	*If null, load all nodes.*
relation	*String*	*Null*	*Yes*	*If null, load all relatioships.*
Configuration	*map*	*{}*	*Yes*	*Basic Configuration as stated by the software requirements.*

D. Algorithm: Keyword Attribute Structure

Input:

G=(V,E);

$V=(V_1, V_{2....}, V_n)$;

$E=(E_1, E_2, \ldots, E_m)$;

$E_i=\{atr_1(V_i), atr_2(V_i), \ldots., atr_j(V_i)\}$;

Output:

$I = \{C_1 : \{E_1, V_1, count(C_1)\}, C_2 : \{E_2, V_2, count(C_2)\}, \ldots, C_n : \{E_m, V_n, count(C_n)\}\}$;

1. Initialize i, j, k, count_node = Ø
2. for Vertices $v_i \in V$ do
3. for Class $C_j \in I$ do
4. k_core ← get K_core(G,v_i) value;
5. $atr_i \leftarrow E(v_i)$of V_i;
6. $atr_j \leftarrow E_j \in C_j$;
7. $Sim(atr_i, atr_j) \leftarrow |atr_i \cap atr_j| / |atr_i \cup atr_j|$;
8. $Avg\_weight(atr_i, atr_j) \leftarrow |W_e(atr_i \cap atr_j)| / |atr_i \cup atr_j|$;
9. if degree deg_vi >= k_core then

TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 4995-4998

10. if Sim(atr_i,atr_j)>=Θ_c then
11. if Avg_weight(atr_i,atr_j)>=Θ_w then
12. atr_i Uatr_j ←E_j {for $E(v_i)$,E_j}
13. V←v_i classify v_i in C;
14. count(v_i)++;
15. Merge attributes of v_i,$E(v_i)$←$E(v_i)$UC_j in class C_j;
16. end if
17. else
18. C_k← k+1;
19. V_k ← v_i;
20. E_k←$E(v_i)$;
21. count(C_k)←1;
22. Set I={C_k : {E_k,V_k,count(C_k)}};
23. Set $E(v_i)$ ←$E(v_i)$UC_k where $E(v_i)$Ɛ E;
24. end if
25. end if
26. end for
27. end for
28. return I

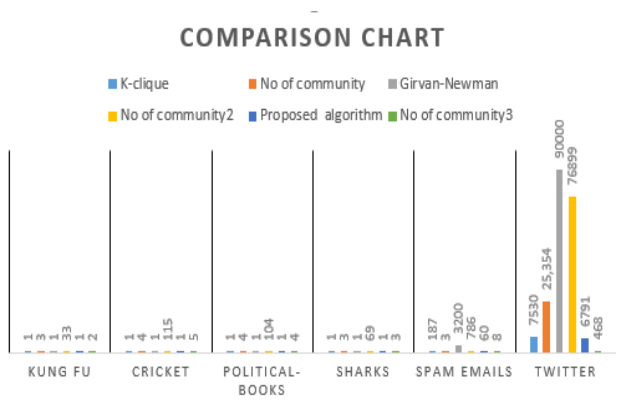
3. Result

On implementing the above keyword Attribute Structured Search algorithm over Louvain algorithm, it is very much anticipated to be more efficient than the current system and also fetch more accurate data from the tree structure. Clustered data is more effectively iterated through keyword search to perform community detection. The process time will be reduced to half and the time and resources are very much conserved compared at present. In the below table we have collected and listed random values of different charts to produce and compare their graphs. We have considered three different algorithms for comparison- K-clique, Newman-Girvan, and our proposed algorithm containing Keyword-Based Search. When applied on these different values we get the values entered in the table below.[7]

Table 2: Comparison between different algorithms

Graph	*K-clique*	*No of comm*	*Newman-Girvan*	*No of comm*	*Proposed algo*	*No of comm*
Kung Fu	***1***	***3***	***1***	***33***	***1***	***2***
Cricket	***1***	***4***	***1***	***115***	***1***	***5***
Political	***1***	***4***	***1***	***104***	***1***	***4***
Sharks	***1***	***3***	***1***	***69***	***1***	***3***
Spamemails	***187***	***3***	***3200***	***786***	***60***	***8***
Twitter	***7530***	***25154***	***90000***	***76899***	***6791***	***468***

Based on the table above we have drawn a graph for simpler understanding. In the below graph we can observe that the performance of the proposed algorithm is very liable, efficient and accurate as estimated. We can see that the bars in the proposed algorithm are low which indicates that implementation of Keyword Search over Community Detection results in a very efficient algorithm.



The above graph is a representation of the listed table. It can be observed that the performance of the proposed algorithm is efficient.[7]

4. Conclusion

It is guaranteed that by implementing the proposed idea in this paper we can make the considered algorithm more efficient and also the accuracy improves drastically along with it. The results from the task will be more to the point and junk data that stick along are removed along the way say as to provide a clean and neat data results to the client. It reduces the programmers work and also future works on automation can also be anticipated from the implementation of this method.

References

[1] J. Xiang, Z.Z. Wang, H.J. Li, Y. Zhang, S. Chen, C.C. Liu, …, L.J. Guo (2017) "Comparing local modularity optimization for detecting communities .", International Journal of Physics C Vol. 28, No. 6

[2] Yufei Tao, Stavros Papadopoulos, Cheng Sheng and Kostas Stefanidis (2017), "Nearest Keyword Search on XML Documents",Department of Computer Science and EngineeringChinese University of Hong Kong,New Territories, Hong Kong

[3] Kavita V.V Ganeshan, N.L.Sarda and Sanchit Gupta (2017), "Keyword Search on Geospatial

May - June 2020
ISSN: 0193-4120 Page No. 4995-4998

Database", GISE lab, Indian institute of technology, Bombay

[4] S. Ahajjam, M. El Haddad and H. Badir. (2018). "A new scalable leader-community detection approach for community detection in social networks". Social Networks, 54, 41-49

[5] Aggeliki Dimitriou and Dimitri Theodoratos (2018), "Efficient keyword search on large tree structured datasets", Department of Computer Science New Jersey Institute of technology USA

[6] Ilaria Bartolini and Paolo Ciaccia (2019), "Multi-dimensional keyword-based Image annotation and search", DEIS, University of Bologna, Italy

[7] Sanket Chobe and Justin Zhan (2019), "Advancing community detection using Keyword Attribute Search", Journal of Big Data volume 6, Article number:83