Acta Technica 62 (2017), No. 5A, 459–468

Heterogeneous information integration mode in dynamic internet of thing

Wang jin yun[2,3], Hu Ruo[4]

Abstract. We have product a dynamic Internet of Thing Integration Mode based on Changeable Web-page Extended Mode (CWEM) called CWEM Dynamic (CWEMD). We compared CWEMD to CWEM, Keyword-classify, and another Mode we product, Dynamic Structure Classifying Mode for Web Heterogeneous Information (SCMWRI-D), to check its result and performance to find Heterogeneous information in Internet of Things when aim at other Internet of Thing Integration Modes. This CWEM Mode, checked on two real dynamic networks, increase efficiency that it would take CWEM to run, and creates common character, and in some factors fine, Internet of Things. In this study, we use an acquainted Internet of Thing Integration Mode to find Internet of Things in dynamic factor. Because of the expanding scale and dynamic nature of Internet of Thing, Modes for dynamic factor have turn into the main content of Internet of Thing searching. CWEMD proves to be effective than other Modes, and creates Internet of Things with fine framework for each Internet of Thing.

Key words. Internet of thing, dynamic factor, web-page, extended mode.

1. Introduction

Internet of Things can take on several different characteristics, ranging from small, such as a close-knit group of friends on Tencent, to large, such as millions of individuals following an international star on Sina, and being able to include several different types of people within the Internet of Thing. Other important aspect of these Internet of Things is their performance to overlap. Modes that detect Internet of Things in dynamic environment use several snapshots, or timestamps, of the network, taking into account the changes of the network that happen naturally between consecutive timestamps, and from there also create Internet of Things. Research has

[1]Acknowledgement - This paper is supported by The National social Science Foundation of China with project number: 17XMZ091,the Science and technology support program in NingXia province.

[2]Workshop - School of Management ,Hefei University of Technology, HeFei 230009, China

[3]Workshop - School of Business, North Minzu University, YinChuan 750021, China

[4]Workshop - Guangdong Polytechnic Normal University, GuangZhou 510665, China

created a variety of Modes to analyze these massive networking sites, creating opportunities to study and find the dynamic, habits, and general rules of Internet of Thing Internet of Things. What are commonly used to study these networks are Internet of Thing Integration Modes, which use various techniques to group and create collections of nodes that form Internet of Things. Internet of Things can be described as distinct groups of nodes that are very well connected between closely related nodes and only loosely connected between other groups of well-connected nodes inside the network. In the current world we live, interactions between individuals have turn into far easier than before due to online media. Massive Internet of Thing, such as Tencent, Sina, and Sohu, have allowed individuals to connect with one another on a vast level. These interactions can be modeled with networks by turning the individuals on the network into nodes, and showing their communication between other individuals as edges between nodes, as modeled by Figure 1. Networks can be described as either static or dynamic: a stationary structure or a continually dynamic structure over time. Nodes from these networks tend to group closely with each other, forming Internet of Things. A single node can belong to several different Internet of Things, such as a node on Tencent being friends with nodes from a workplace and nodes from a sports team, creating more dynamic interaction between Internet of Things than what would be allowed otherwise.

2. Relate Works

Another Mode, Keyword-classify, uses the idea of Web-page extended for its Internet of Thing searching, but instead of propagating Web-pages at random, Keyword-classify uses the performance that a node will receive a Web-page from a neighbor, which removes the randomness that is seen in TSM and CWEM. Due to this avoidance of randomness, Keyword-classify create more stable Internet of Thing structures than TSM and CWEM. Keyword-classify also use a stop criterion, which ensures a safe termination that also creates good results. The stop criterion, however, does not always guarantee the best runtime, but Keyword-classify is still very efficient despite this [3].

One Mode that uses ROM for Internet of Thing searching is ROM-Net, which uses a system of assigning all the values to generate (which serve as nodes) to form Internet of Things. ROM-Net starts by randomly generating genotypes for the network, and performing a check to make sure each individual can be regarded as safe, meaning there is an edge between the gene and the allele value assigned to it and protecting against improper grouping. ROM-Net then performs uniform crossover, which creates a random binary vector, then uses that vector to select genes from each parent to pass on to the child. Since both parents are assumed to be safe due to the check, the child can also be assumed to be safe. After this, the fitness function, in ROM-Net's factor Internet of Thing score, is computed for each member, and then a new population is created from the old members, eventually converging on a solution to the problem [5].

Another approach to Internet of Thing searching is based on Random Optimization Modes (ROM). ROM imitates evolution to find the optimal solution to a problem. ROM starts by generating a random population of solutions, measuring each solution strength based on a fitness function, then mutating population to form a new generation of solutions. The process repeats until either the maximum amount of Heterogeneous is reached, or an acceptable level for the fitness function has been reached, indicating an optimized solution to the problem at hand. Applying ROM to Internet of Thing searching would mean using this same process to develop a Internet of Thing structure that would satisfy a fitness function for Internet of Thing strength [5].

Keyword-classify update all of its nodes at the same time, which can be inefficient. A Mode based on Web-page-Rank, Keyword-classify, solves this problem by using local operations, working incrementally and only updating nodes that have changed in consecutive timestamps. This allows for greater efficiency, and being able to track a Internet of Thing through a change in time, due to its deterministic results [3].

To detect Internet of Things in dynamic networks efficiently, CFSMW (Dynamic FSMW) was created as an improved version of FSMW that, instead of updating the entire network with each Heterogeneous, finds the difference in edges between consecutive timestamps, and updates only the nodes involved in the changed edges. CFSMW also no longer requires a user-input for the ε threshold, instead finding an average for ε between the values of 0.2 and 0.6. CFSMW, when dealing with larger Internet of Things, tends to perform quicker than FSMW, while at the same time maintaining the same quality of FSMW in framework, a common benchmark for determining the strength of a Internet of Thing [4].

Another development for Internet of Thing searching in static networks is Structure Classifying Mode for Web (FSMW), which uses distance between nodes to determine Internet of Thing placement, FSMW uses a user-defined threshold ε to determine a node's neighbors, using this information to incorporate the node into a Internet of Thing. Nodes that share common character neighbors in their ε-neighborhood (the user-defined threshold) can be grouped and formed into a Internet of Thing. FSMW also allows for the searching of hubs and outliers, hubs being nodes that don't belong to a Internet of Thing, but connect two or more Internet of Things, and outliers being nodes outside of Internet of Things that have neighbors from only one Internet of Thing [4].

Internet of Thing S In the pursuit of Internet of Thing searching, there are several Modes present today that offer a variety of techniques to process networks and find Internet of Things. One such Mode is Web-page Extended Mode (TSM), which finds Internet of Things by allowing a random node to turn into a receiver, taking on the most popular Web-page of all its neighbors. If there is a tie between two or more Web-pages, which happens often in the first few Heterogeneous, a Web-page is selected at random from them. The process ends when each node's Web-page does not change, or if the Mode arrives at a user-set number of Heterogeneous, and the Internet of Thing membership is determined based on the Web-pages the nodes have received [1].

3. Searching in Dynamic Networks

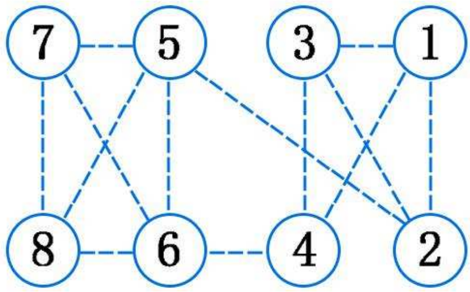

Fig. 1. A network of eight nodes in Cultural Network

Changes in a dynamic network from $\mathbf{G}_t$ to $\mathbf{G}_{t+1}$ are seen in the changes from $\mathbf{E}_t$ to $\mathbf{E}_{t+1}$. Any new or removed nodes in $\mathbf{G}_{t+1}$ will be connected to an edge, which will be found in $\mathbf{E}_t$ or $\mathbf{E}_{t+1}$; if a node does not contain any incident edges, then the Internet of Thing structure of a network would not be affected by the disjoint node. In a static network, let $\mathbf{G} = \{\mathbf{V}, \mathbf{E}\}$ be a network where $\mathbf{V}$ is the set of nodes and $\mathbf{E}$ is the set of edges consisting of node pairs $(u\mathbf{V}, v\mathbf{V})$. In a dynamic network, let $\mathbf{G}_t = \{\mathbf{V}_t, \mathbf{E}_t\}$ be a network at timestamp t with $\mathbf{V}_t$ and $\mathbf{E}_t$ being its corresponding node and edge sets respectively.

Our approach to Internet of Thing searching in dynamic networks focuses on this idea of updating a Internet of Thing structure around the nodes that have seen edge changes between two consecutive timestamps.

4. CWEMD

Since CWEMD only updates based on edge changes, a problem can arise from only one Internet of Thing involved in the edge change being updated, which can complicate finding a Heterogeneous information node between two Internet of Things if only one Internet of Thing of two in a possible overlap is updated. To combat this, we use a simple technique to repair Heterogeneous information in CWEMD, which works by finding the nodes attached to removed edges in the network, determining how many neighbors of each Internet of Thing the nodes have, and then deciding which Internet of Things the node belongs to.

Due to the large size of many networks, runtime can turn into a problem for many Modes. For some Modes, threading can be used to cut down on this problem. CWEMD allows for threading, where each node can independently ask its neighbors for Web-page's and obtain the maximum occurring Web-page of the received Web-pages. Threading can also handle the removal of low frequency Web-pages and the Internet of Thing repair of nodes. A potential problem for CWEMD is finding the right overlap for a node. If an edge between a node and its Internet of Thing disappears, its degree to the Internet of Thing decreases, which can possibly bring its degree down close to that of a different Internet of Thing.

Internet of Thing repair works by checking each node's neighbors and moving that node to the Internet of Thing with the highest neighbor count, effectively making sure that nodes belong to the same Internet of Thing that the majority of its neighbors belong to. One factor for the selected listener node is that the only speakers it can receive from must be within the node set that is being updated. In our Mode, along with the idea of the listener node receiving a random Web-page from a speaker, the speaker can also be able to send their highest frequency Web-page to the listener. One of the ways CWEMD helps to improve framework and tighten Internet of Thing structure is by using our Internet of Thing repair function.

The CWEM Mode is then run on these nodes, with $\mathbf{T}$ as number of Heterogeneous and r as the cutoff threshold. Essentially, CWEMD involves running CWEM on Internet of Things that change from one timestamp to the next. For each graph $\mathbf{G}_i$, CWEMD obtains the edges, $\mathbf{\Delta E}$, that have changed between two consecutive timestamps and acquires all nodes incident to them, $\mathbf{\Delta V}$. This set of nodes is then expanded to include all nodes belonging to Internet of Things of these nodes. This helps the CWEM Mode converge quicker and create more stable results when run multiple times. Before CWEM is run on the nodes that will be updated, two cleanup techniques must be performed. First, Web-pages of nodes that disappeared from the previous timestamp to the current are removed from all nodes in the network. This helps to clean out Web-pages of nodes that no longer exist, because these nodes can no longer propagate their Web-pages through the network. The final step before running CWEM is to reset the Web-pages of the nodes that are about to be updated. This is essentially the initialization phase of CWEM, but only on the nodes that will be updated.

We propose the Mode CWEMD (Changeable Web-page Extended Mode Dynamic) which uses CWEM's Web-page extended technique to propagate Web-pages only throughout portions of the network, reducing runtime. Mode 1 illustrates the pseudo-code for CWEMD. Our study is based upon the Mode CWEM and incorporates the performance for this Mode to handle dynamic networks.

5. Heterogeneous Information Internet of Thing Modes

We modified CFSMW to detect Heterogeneous information Internet of Things, calling the product CFSMW Heterogeneous information (CFSMWO). To detect nodes that connect Heterogeneous information Internet of Things, after CFSMW's Internet of Thing searching, SCMWRI-D checks every node and acquires each of their maximum common character of all incident edges. A user specified common character-decrease value defines the Heterogeneous information threshold. Any node with an edge that is incident to another Internet of Thing with a common character greater than the node's maximum common character minus the common character decrease turn into part of the other Internet of Thing and signifies overlap between the two Internet of Things.

And SCMWRI-D, we modified Keyword-classify to be able to detect Heterogeneous information Internet of Things. Essentially, when the highest performance Web-page is found, a threshold is introduced, and if any other Web-page is within the highest Web-page performance minus the threshold, the node takes on that Web-page as well. We gave Keyword-classify and SCMWRI-D very common character Heterogeneous information Internet of Thing searching modifications.

Due to how these two Modes form Internet of Things, performing Heterogeneous information find was a simple task. For a benchmark on Heterogeneous information Internet of Thing searching, we modified two disjoint Internet of Things Integration Modes and equipped them with the performance to find Heterogeneous information Internet of Thing structure.

6. Dynamic Network Database

The next network, the Things of Internet Call-ToI network, is much smaller than the physics network, being made up of only 68 nodes and 264 edges. In this Database, nodes are undergraduate students, and the edges between them represent calls or ToI Web-page messages between students. The time period it includes is between October 2007 and May 2010 [7].For evaluating the result of Internet of Things, we use framework [9]. Framework measures the difference between the numbers of edges found in a Internet of Thing to the expected number of edges in a randomly generated Internet of Thing structure. Structure can be described as:

$$Q = \frac{1}{2M} \sum_{u,v} (A_{u,v} - \frac{k_u k_v}{2M}) S_{uv} \tag{1}$$

Where **M** is the number of edges, **A** is the adjacency matrix, k_x is the degree of node x, and $\mathbf{S}_{uv}$ returns 1 if u and v are in the same Internet of Thing and 0 otherwise. Since our Modes can detect Heterogeneous information Internet of Things, we will also use **EQ** [1, 0]. **EQ** can be described as:

$$EQ = \frac{1}{2M} \sum_{u,v} (A_{u,v} - \frac{k_u k_v}{2M}) \frac{|l_u \bigcap l_v|}{|l_u|\,|l_v|} \tag{2}$$

where **M**, **A**, and k_x are the same, and l_x is the set of Internet of Things of x. **EQ** is framework that evaluates all common Internet of Things between nodes. When no Heterogeneous exists in a network timestamp, then **EQ** is framework. Both framework and **EQ** create values in the range of 0 to 1, where higher values indicate a fine Internet of Thing structure.

The final network we used is common character to the Call-ToI network described above. The Family-ToI Network tracks mobile phone usage between individuals. However, unlike the Call-ToI network above, this network only tracks ToI messaging. These interactions were recorded between March 2010 and June 2011 [8].

The first and largest, the arXiv HEP-TH physics network, is a citation graph that includes 26,680papers and 342,604 citations between papers. This data in the graph is comprised of papers written between January 1992 and April 2002 [6]. To check our new Mode aim at other commonly used Modes, we employed three real world networks ranging from small to large. Due to the gradual collection of the papers during the 10 years, we were able to generate timestamps for the graph to check with CWEMD.

7. Results

As noted before, CWEMD can be run as a threaded Mode. Features of CWEMD that can run synchronously are: nodes can receive Web-pages from its neighbors for one Heterogeneous at a time, removing low frequency Web-pages from each node, and our Internet of Thing repair function. In view of large networks, performing these operations synchronously greatly increases overall runtime. The comparison between CWEMD being threaded and unthreaded, or CWEMD-Threaded and CWEMD-Unthreaded, runtimes for each timestamp is showed on the physics network. As depicted, a low number of changes results in inefficiency from threading due to the memory overhead needed for threading, making CWEMD-Unthreaded ideal for small edge changes. Node changes reflect all nodes of Internet of Things that saw edge changes and will be updated. As the number of edge and node changes increases, the runtime for CWEMD-Unthreaded increases while CWEMD-Threaded has a slower increase in runtime. For a large number of edge changes, CWEMD runtime increases at a greater rate than CWEMD-Threaded. The number of edges changed and the number of nodes will need to be updated for each timestamp.

CWEMD performs comparably and effective than CWEM, Keyword-classify, and SCMWRI-D. CWEMD without Internet of Thing repair performs just below CWEM in framework, but with a lower runtime. When Internet of Thing repair was added to SLAPD, it remained effective than CWEM, and boosted results far above regular CWEMD and above CWEM. SCMWRI-D's runtime matches CWEMD's runtime in the first half of the timestamp, but as the network increases in size, CFSMW's runtime increase quicker than CWEMD's runtime. CWEMD was able to run effective than any Mode we checked. This is due to the complexity of CWEMD, which is no longer that of CWEM's $\mathbf{O(Tn)}$, but instead comes down to the nodes of Internet of Things that saw edge changes. Note that checks on CWEMD were averaged over ten runs to account for the fluctuating results due to randomness of the Mode. SCMWRI-D suffers in framework due to its searching of hubs and outliers, making framework comparison inefficient to compare SCMWRI-D to other Modes. So CWEMD has a complexity of $\mathbf{O(Tx)}$ where $x <= n$. CWEMD and Keyword-classify both have a fluctuating runtime curve, due to the efficiency of each Mode to only update segments of the network where changes occurred; the amount of changes from one timestamp to another is the cause of the fluctuating runtimes.

When comparing CWEMD and CWEM, we can also compare how they detect Heterogeneous information nodes. The common character in Heterogeneous information nodes is found by SLAPD and CWEM for each timestamp of the Physics network and the Call-ToI network respectively. We can see that CWEMD and CWEM are somewhat common character in Heterogeneous information node searching. It is important to note the beginning of both graphs; at first, both graphs are perfectly common character, which is caused by neither graph having found any Heterogeneous information nodes yet, and is followed by a sudden drop to zero, which occurs when one Mode finds Heterogeneous information nodes while the other still has not. The framework and EQ of the Internet of Thing structures found by these same Modes on the Call-ToI network. CWEMD created common character framework to CWEM and fine than Keyword-classify and SCMWRI-D, and generating common character EQ to CWEM, and SCMWRI-D. CWEMD with Internet of Thing repair did not improve CWEMD's Internet of Thing structure as significantly on the Call-ToI network as it did on the Physics network, due to CWEMD not needing as much repair for this network. SCMWRI-D is still hindered in both framework and EQ because of detecting hubs and outliers, which decrease these two measures. After these two irregularities, both Modes start detecting Heterogeneous information nodes, and normal results are created. Common character is defined as:

$$S = \frac{|A \bigcap B|}{\sqrt{|A|\,|B|}} \tag{3}$$

Where **A** and **B** are the sets of Heterogeneous information nodes being compared by two Modes on the same timestamp.

Because of these results, both CWEM and CWEMD will be assumed to use the frequent Web-page feature for the rest of the section. To check CWEMD, we ran it on the aforementioned networks along with CWEM, Keyword-classify, and SCMWRI-D. CWEMD ran effective than all three other Modes and Internet of Thing repair improved framework when CWEMD without Internet of Thing repair was not enough. All Modes checked detect Heterogeneous information nodes with a slight degree of common character. As noted previously, CWEMD provides the option for a listener to obtain either a random Web-page or the most frequent Web-page from a speaker. CWEMD using most frequent Web-page, called CWEMD-Frequent, outperforms CWEMD using random Web-page, called CWEMD-Random, on the Call-ToI network. We also ran this same check on the Friends and Family-ToI network, producing very common character results to the Call-ToI network. Each version of CWEMD was average over twenty runs, due to the randomness of the CWEMD Mode.

8. Conclusion

When searching Internet of Things in dynamic networks, CWEMD performs effective than any other Mode we checked and creates comparable, if not fine, results in terms of framework and EQ almost all the time. When running on the Call-ToI network, CWEMD created about equal Internet of Things of common character framework with both CWEM and Keyword-classify. CWEMD was also able to create fine framework with a shorter runtime than Keyword-classify on the physics network. On both the physics and Call-ToI networks, CWEMD was also able to detect a mildly common character amount of Heterogeneous information nodes when aim at CWEM. CWEMD proves to be a fast and reliable dynamic Internet of Thing Integration Mode that is capable of producing Internet of Things of common character or fine quality when aim at existing Modes.

References

[1] W. Q. Zheng, Bundell. G: *Contract based information system component measurement with UML modes.*International symposium on computer science and its applications (CSA '08)*29* (2008),83–102.

[2] R. Hu, H. Hu, H. Xu: *Big data-based intrusion behavior find in sensor networks, Journal of investigative medicine. 63* (2012).

[3] J. Gao, Shih, M. C: *A component feasibility mode for verification and measurement.* Proceedings of the 29th annual international computer information system and applications conference, Edinburgh (2005), No. 4, 555–563.

[4] R. Hu, H. Hu, Z. H. Xiao: *Research on integration mode of sensor network component based on demand concept set.* Journal of investigative medicine*63* (2015), Nos. 1–5, 198–210.

[5] R. Hu: *New network access control mode using intelligence agent technology.*Applied mathematics & information sciences(2013).

[6] J. P. Fu, M. Y. Lu: *Request-oriented mode of information system feasibility measurement.* Proceedings of the ITCS 2009 international conference on information technology and computer science *20*, (2009),77–80.

[7] Aharony, N: *Tracing mobile phone app installations in the Friends and Family study.* Proceedings of the 2010 workshop on information in networks (WIN'10).

[8] R. Hu: *Channel access controlling in wireless integrated information network using smart grid system.* Applied mathematics & information sciences (2012), No. 4, 813–820.

[9] Huda, M. Arya, Y. D. S,Khan, M. H: *Evaluating validity element of big data structure design: a feasibility perspective.* International journal of information system engineering & applications (IJSEA) (2015).

[10] R. Hu: *Stability analysis of internet of thing service via data stream methods.* applied mathematics & information sciences(2012).

Received November 16, 2017