Community Clustering Algorithm on Semantic Similarity in Complex Network

Laizhong Cui, Nan Lu, and Yuan Yuan Jin

Abstract—This paper proposes applying a WordNet based semantic similarity clustering algorithm on the cluster analysis of complex network community. By utilizing the semantic hierarchy of WordNet, the proposed method defines the key concept sets and the concept feature values for the community and constructs the semantic relations between concepts of the community nodes, which expands the application of clustering algorithms from text documents to complex network communities. The cluster structures derived from the proposed algorithm are concordance with peoples' judgments on a specific area, which will lead to the solution of the clustering problems in the complex network of different areas. Compared with VSM and k-MEANS, the proposed algorithm discovers more reasonable results and shows its effectiveness.

Index Terms—Semantic similarity, WordNet ontology, complex network, community structure, clustering algorithm, key concept set.

I. INTRODUCTION

Semantic similarity [1], [2] is a concept with various definitions according to different areas. Take the term "virus" as an example, the similarities of virus and its categories differ when putting it in the biological area and computer field. This is caused by different definitions in these two areas. Therefore, understand the definition other than the term itself becomes more important.

The WordNet ontology [3] is an online word sense mapping system containing concept word sets from different areas and relationships among them with a semantic network structure. Based on Wordnet, this paper extract words and construct a semantic IS-A relationship hierarchy. Mix concept word set into standard hierarchical structure.

In similarity calculation area, the most popular method is by using vector space model (VSM) [4], [5] to solve the problems of different weighted features and classification learning from long text and Web document processing. Most of these algorithms costly in computing power, Some algorithms need some background knowledge as well as manpower and therefore are difficult to be applied in unstructured complex network, Especially, in the community structure nodes with similar meanings yet few common words, VSM will heavily affect the effect of cluster finding and result in serious deviation from actual structure. For example, the "sports community" and "badminton clubs"

should belong to the topic of sports, yet the VSM will return a 0 in similarity due to no common words.

Complex networks are abstractions of complex systems and each node in the network represents the individual unit in the complex system. The edges between nodes are relationships in the network form according to certain rules. There are various types of complex network in the real world, such as social network, biological networks, etc. Finding community structure is not a random selection in a large number of nodes with same properties, but discrimination in nodes with different types, among which the nodes with same property are linked with more connections, while different types of nodes are sparsely connected. Finding community structures within a complex network is an important step towards clustering analysis and research of the network.

Clustering is an important method in founding community structure. Through the clustering, internal regulations and characteristics can be discovered. Clustering algorithms is capable of automatically generating category number without adding manual annotation and training classifier. As a unsupervised machine learning method, clustering has a higher flexibility and better automatic processing power. As the increasing tendency of community information dependence [6]-[9], people require intelligent information processing other than word patter or word sense processing. Therefore, semantic similarity computing becomes one of the ways to solve community clustering problem. It is crucial in improving the effectiveness and accuracy of clustering result, judging community structure correlations, classifying communities, and mining data.

This paper proposes a WordNet semantic network learning methods. By utilizing the semantic hierarchy of WordNet, the proposed method defines the key concept sets [10] and the concept feature values for the community and uses them to define the semantic similarity. Then according to the semantic similarity, a community clustering method in complex network is presented. The cluster structures derived from the proposed algorithm are concordance with peoples' judgments on a specific area, which will lead to the solution of the clustering problems in the complex network of different areas. Compared with VSM and k-MEANS [11], the proposed algorithm discovers more reasonable results and shows its effectiveness.

II. RELEVANT KNOWLEDGE

A. WordNet Ontology

WordNet is a widely used English words knowledge base and widely applied in natural language processing, semantic translation, has attracted many national attentions [12]. WordNet is organized by semantic relations. It uses synonym

Manuscript received January 6, 2014; revised April 1, 2014. This work was supported by the Shenzhen basic research under Grant NO.CYJ20130329102017840.

Laizhong Cui, Nan Lu, and Yuan Yuan Jin are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, Guangdong, PRC (e-mail: cuihz@szu.edu.cn, lunan@szu.edu.cn).

sets (synsets) to representative concepts. Keywords in synsets are bounded and the semantic relation between synsets is also kept in the hierarchy. One word can be mapped to several synsets and one synset contains several words, then it provides a way in representing semantic relationships into the relationships between the concept set or synsets. WordNet semantic relations mainly include: parent and child, synonymous, antonym, is-a-part-of and containment, attribute properties, "leading to" relationships and so on. Based on the English Wordnet, the Chinese WordNet is ontology of the Chinese words and concept word set, by using existing English-Chinese dictionary library to translation the word in English word into Chinese and get the knowledge base. It also has function of the concept word, same-word, and pan-word. The key concept word is the basic elements of Chinese WordNet, and use a number of relation types to connect these concept words, which leads to a key concept word set.

B. Similarity Calculation

The calculation of similarities between any two words starts from mapping the words into the concept word sets that they belong to, and then calculates the similarities between each pair. Finally, according to these similarities of concepts word sets, the word similarities are achieved.

1) Concept word and similarity

For the convenience of knowledge sharing and reuse, WordNet clearly defines concepts of different areas and their relationships. Since there is no formal standard, the descriptions of the same problem in different areas will differ. Even in the same area, different ontologies may also have some heterogeneity. This will significantly affect the use of WordNet. Ontology mapping is one of the ways to resolve this heterogeneous problem, while the similarity calculation is a key part of the ontology mapping.

Definition 1(concept word): Concept is an abstract describe of the objects in real world. A concept word is defined as a triad $Con=\{N, A, R\}$, where N is the concept name, A is the concept attribute set, R is the concept relationship set. The name and attribute describes internal characteristics of the concept, relationship set expresses the relation between concept and external environment, reflects the external characteristics of concept. Concept words can be represented by instances and therefore be more specific in concept meaning.

Definition 2 (similarity): By defining 1, it is known that the concept word has three important elements including name, attribute and relationship. So the similarity calculation including concept name similarity, attribute similarity and the relation similarity. WordNet is seen as a semantic tree organized according to the hierarchical relationships and concept word relationships. In this paper, the similarity calculation is based on the similar between two concept word of WordNet, which is the semantics distance between them (more specific, the path length of two semantics in the semantic tree). Similarity values range for 0 to 1. If two concept words are completely different, the similarity is 0. If they are identical, then the similarity is 1 by using following formula, where $len(w_1, w_2)$ is path length from the word w_1 to the word w_2 .

$$SIM(w_1, w_2) = \frac{1}{1+len(w_1+w_2)} \quad (1)$$

a) The concept name similarity SIM_{cn}

Usually, the concept word in WordNet is a compound word and it is to determine its semantic distance directly. Therefore, the first several steps will be word segmentation, stopping-word removal form a stopping-word table and key words extraction, then formula (1) is used to calculate key words similarities and get the summation to represent the similarity of concept name as follow:

$$SIM_{cn} = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m SIM(c_{1i}, c_{2j}) \quad (2)$$

among which, c_{1i} is the key word of concept c_1 , where $i \in [1, n]$. c_{2j} is the key word of concept c_2 , where $j \in [1, m]$.

b) Attribute similarity SIM_a

Attribute composes of attribute names (reflect attribute contents) and attribute domain (attributes' value ranges). Therefore, attribute similarity calculation must include attribute name similarity and attribute domain similarity:

$$SIM_a(a_1, a_2) = SIM_{an}(a_1, a_2) + \frac{1}{n} SIM_{ad}(a_1, a_2) \quad (3)$$

where $SIM_{an}(a_1, a_2)$ is the attribute name similarity, it can use a similar concept name similarity calculation method to calculate the result. $SIM_{ad}(a_1, a_2)$ is attribute domain similarity, it mates calculate.

c) Relationship similarity SIM_r

Relationship similarity reflects the connection degree between concept and external. The similarity calculation includes two parts: the relationship name similarity and relationship association concept similarity.

$$SIM_r(r_1, r_2) = \frac{1}{n} SIM_{rn}(r_1, r_2) + \frac{1}{n} SIM_{rc}(rc_1, rc_2) \quad (4)$$

Among them, $SIM_{rn}(r_1, r_2)$ is the relationship name similarity, achieved by similar calculation method as concept name similarity. $SIM_{rc}(rc_1, rc_2)$ is the association concept similarity, it is calculated from the concept name similarity of association concept, where rc_1 represents the concept of associate relation r_1 , rc_2 means concept of associate relation r_2 .

2) Semantic similarity

Semantic similarity calculation principle:

- 1) According to WordNet parent and child relation, The further the distance between any two concept word nodes, the smaller the semantic similarity.
- 2) The higher density the concept word node locates, The finer the local concepts are divided leading to a lower the similarity.

Two concept word nodes which have the same distance, the deeper the level it locates, the more specific it will be, then greater similarity is assigned.

The semantic similarity calculation formula is as follow:

$$SIM = \sigma + \alpha \times \bar{\varphi} + \beta \times \bar{\omega} \quad (5)$$

where α and β are the weights of depth factor and density factor respectively, σ is the distance factor, φ is the density factor and ω is the depth factor.

C. Community's Key ConceptWord Set and Concept Feature Value

In the community clustering process, different community

structure or different keyword density in community will lead to the distorted structure. Especially when the community data is in a high dimension, the quality, effect and the calculation speed of the cluster are significantly decreased. In order to improve the efficiency of cluster mining, the dimension reduction method is a better choice. At present, dimension reduction methods mainly includes (TF-IDF), information gain (IG), mutual information (MI), etc. [12], [13], which are based on the lexical frequency statistics information. For the convenience of clustering operation, a structuring process for nodes in complex network is required, which include: establishing community key concept word set and extracting concept feature values, and forming structured documents containing key concept words. Similar to text document processing, the words in structured documents can be divided into two classes: function words and content words. Function words are particle, which has no real meaning while content words are meaningful. According to the features, such as frequencies, positions and so on, of content words in the network, weights are assigned to these words to obtain the concept feature values. This value is propositional to the frequency of the concept associating with it and if a concept appears in the title of the network, the concept feature value of it will be increased. When a certain concept feature value is greater than a given threshold, then the words can be regarded as key concept word.

At present, for clustering purpose, a document is always transformed into a noun list yet ignores the contribution of words' frequencies to the content of the document [14], which will lead to a unsatisfying performance. This article utilizes the key concept word list illustrated in formula 7, which treat a social network as a two-dimension array include concept words and their frequencies, to meet the requirement of clustering in the complex network.

$$\sigma = \begin{cases} \sqrt{1 - \frac{\ln^2}{\theta^2}}, & \ln < \theta \\ 0, & \ln \geq \theta \end{cases} \quad \varphi = \frac{1}{\ln PN + 1}$$

$$\omega = \begin{cases} \frac{\sqrt{|dep - E_d|}}{E_d}, & dep \geq E_d \\ -\frac{\sqrt{|dep - E_d|}}{E_d}, & dep < E_d \end{cases} \quad (6)$$

where w_i is the i -th concept appears in community; f_i is the frequency associate with w_i . f_i is calculated by the frequency function, namely:

$$D = \{(w_1, f_1), (w_2, f_2), \dots, (w_n, f_n)\} \quad (7)$$

In (8), T_i is the feature value associate with the i -th concept word appears in community; TF_i is the times which the first i concept word appeared in the community; m_i is the number which communities containing the first i concept word; M is the total number of the concept word in community. From (7), it is obvious that the feature value of a concept word is proportional to the frequency that the concept word appears in sentence, and in inverse proportion to the number of communities containing the concept word.

$$T_i = TF_i \log \left(\frac{M}{m_i} \right) \quad (8)$$

III. HIGH PERFORMANCE CLUSTER ALGORITHM CACN

At present, there is no clustering algorithm that is generally

applicable to the complex network in revealing the complex structures that are represented by all kinds of multi-dimensional data sets. According to the clustering methods, clustering algorithms can be classified into partitioning clustering, hierarchical clustering and density based clustering. The classic partitioning algorithms are vector space model clustering and k-neighborhood clustering [15], they are efficient for large data set and applicable to Web document clustering applications. The hierarchical clustering algorithm uses association rules to split or cluster data in a hierarchical form to provide solution for hierarchical clustering. It is mostly applied in small data set.

To address the problems of predefined cluster number, initial value selection and local optimal issue, this paper proposes a clustering algorithm based on semantic similarity CACN (Clustering Algorithm of Complex Network) to efficiently solve community clustering problem in the complex network.

A. Basic Ideas

The basic idea of the proposed complex network community clustering algorithm is to define node distance between community structures and represents similarity between community structures by node semantic similarity. According to the similarities, the nodes are clustered one by one, and the closely related clusters will gather into a bigger cluster unit, which will grow in size gradually until all nodes form a cluster.

Fig. 1. Basic idea of CACN.

B. Algorithm Details

As the first step of CACN include network structuration and feature extraction. Using Wordnet concepts and the semantic relationships among concepts to generated key concept sets and the concept feature values representing the community structure. Then execute the clustering algorithm based on the key concept sets and the concept feature value. Finally, using the key concept set to express each clustered unit.

1) Key concept set extraction algorithm CACN-CSET

CACN-CSET(Clustering Algorithm of Complex Network for Concept Set) algorithm scans through the identification of each network nodes to extract semantics and map extracted nouns to concept through Wordnet. Each of the concept is initialized with the an interpretation weight.

Algorithm input: network node identification document set D.

Algorithm output: the key concept of every network node in the document set D $ConSET[i]$.

```

i=0; continue=true; // i index network node document
/*each network node document in circulation processing D */
do
    file=nextfile(D); //take a network node document in order
    if (isnull(file))
        continue = false;
    else {
        titlewords = gettitle(file);
        word=first(titlewords); //extract the first semantic word
        titlewords.remove(word);
        /* remove the semantic word which have taken out and
        update titlewords */
    }
    while(isnotnull(word)) {
        conceptnode = lookupindexword(word, noun);
        if (isnotnull(conceptnode))
            ConSET[i].add(conceptnode,1);
            /*if concept nodes exist,then join ConSET[i],
            the weight is 1*/
            word = first(titlewords);
            titlewords.remove(word);
        }
    i ++ ;
    while(continue != false)

```

2) Concept feature value extraction algorithm CACN-FeatVAL

CACN-FeatVAL(Clustering Algorithm of Complex Network for Feature Value) algorithm maps the semantic word to concept word through then synonym and parent-child relationships in WordNet. Then a small section of the concept words are selected to represent each document of the network structure.

Algorithm input: Semantic word concept feature value array $Feat$ from a normalization processing (word segmentation, stemming, stopping and word-frequency calculation) of the network structure document sets.

Algorithm output: Content feature value array $Feat[i]$ representing the content of each network structure in the document set D .

```

i = 0; // i index documents
/* circulation handling each network structure document */
do
    for each word in Feat[i] Do
        concept = mapintoconcept(word);
        if (isnotnull(concept)) {
            if (concept in Feat[i])
                add cf to the original concepts weight;
                /* cf is concept frequency, the original concept
                feature value add concept frequency */
            else
                Feat[i].add(concept, cf);
                hypernym = getdirecthypernym(concept)
            }
        if (isnotnull(hypernym))
            if (hypernym in Feat[i])
                add cf to the original hypernym 's weight;
                /* cf is concept frequency, the original
                concept feature value add concept frequency */
            else
                Feat[i].add(hypernym, cf);
            /* put the direct superior concept with concept
            feature value */

```

```

endfor
i ++;
while (I != Feat.length)
    i = 0; /* in network node index */
    /*circulation handle each network node document */
    do
        for each concept in Feat[i] Do
            hypernym = getdirecthypernym(concept)
            if (isnotnull(hypernym) && (hypernym in Feat[i]))
                /* reduce dimension */
                if (cf(hypernym) > cf(concept))
                    Feat[i].remove(concept, cf)
                else
                    Feat[i].remove(hypernym, cf)
            endfor
        i ++;
    while (I != Feat.length)

```

3) Clustering algorithm CACN

Algorithm input: the key concept set $ConSET[i]$ and feature value $FeatVAL[i]$ of each network structure in document set of network structure; Document number n ; Cluster number k ; weight coefficient of he , key concept set kc , weight coefficients of concept feature value cf .

Algorithm output: clustering results $Clusters$, as well as the explanation for cluster results $Results$.

```

P = callsimilarity(ConSET, FeatVAL, kc, cf);
Clusters.initialize(n); //cluster initialized to n cluster
Results.initialize(ConSET);
/* key concepts set ConSET initialized n cluster explain */
do
    findnearestcluster(c1, c2);
    Clusters.merge(c1, c2); // merger two cluster
    Results.merge(c1, c2);
    /* mark key concept set of explanation cluster and merger,
    and according to the similarity of concept in key concept
    set of cluster results */
    update(); // update similarity matrix
    n --;
    while(n>k) {
        callsimilarity(ConSET, FeatVAL, kc, cf)
        findnearestcluster(c1, c2)
    }

```

According to the key concept set and its concept feature vector and its weight coefficient to calculate similarity matrix; find two clusters which their similarity degree are the biggest in the clusters to $c1$ and $c2$.

IV. THE EXPERIMENT AND ANALYSIS

A. Experimental Data

The algorithm's experimental data are taken from 10 discussion groups of a BBS system. The discussion groups contain about more than 20000 discussion topics with a total number of 65000 entries.

B. Evaluation Standard

This paper adopts the analysis method (NMI, Number Mutual Information) based on mutual information within clusters or categories. as stated in [14]. This method can eliminate the influence on the final clustering result caused by the number of clusters. The closer the NMI value to one, the better the clustering result.

$$NMI = \frac{\sum_{h,l} n_{h,l} \lg \left(\frac{n_{h,l}}{n_h n_l} \right)}{\sqrt{\left(\sum_h n_h \lg \left(\frac{n_h}{n} \right) \right) \left(\sum_l n_l \lg \left(\frac{n_l}{n} \right) \right)}} \quad (9)$$

where n_h is the number of data sample in categories h , n_l is the number of data sample in categories l , $n_{h,l}$ is the same data samples in both of the categories h and l , n is the total number of sample data.

C. The Experimental Result

The experiment using the key concept set and concept feature value to calculate the similarity of the text. kc and cf are the parameters to adjust the key concept set and concept feature value, Fig. 2 shows the effects of clustering by computing NMI under different $kc:cf$ ratio.

Fig. 2. Effects of clustering by computing NMI.

Fig. 3. Effects of different clustering methods.

V. CONCLUSION

This paper studies community clustering algorithm based on semantic similarity under the complex network scenario and proposes a feature value using WordNet semantic words which can construct community key concept set to express the community concept. Compared with clustering methods using space vector model SVM, the proposed algorithm shows a better performance. By introducing semantic relations between concept word set (also called synonyms set) and concepts to describe network nodes, The algorithm reduces the dimension of feature lists representing community nodes, and therefore can be applied to clustering analysis of the community structure. The method proposed in this paper deserves more research on some problems. For example, the hierarchical relationships of the WordNet

ontology is still not yet fully utilized. Some fuzzy concept word sets are hard to be defined and better solution to improve the clustering accuracy.

REFERENCE

- [1] R. Zhang, "The clustering research on terminology definition," *Technology Terminology of China*, no. 1, pp. 14-19, 2011.
- [2] S. Q. Zhao, T. Liu, and S. Li, "A topical document clustering method," *Journal of Chinese Information Processing*, pp. 58- 61, no. 2, 2007.
- [3] H. F. Zhu, W. L. Zuo, F. L. He, T. Peng, and W. Y. Ji, "A novel text clustering method based on ontology," *Journal of Jilin University (Science Edition)*, vol. 48, no. 3, pp. 277-285, 2010.
- [4] Q. Y. Yao, G. S. Liu, and X. Li, "VSM-based text clustering algorithm," *Computer Engineering*, vol. 34, no. 9, pp. 39-43, 2008.
- [5] M. W. Yuan and P. Jiang, "Compression algorithm for ontology based vector space model computer engineering and applications," *Computer Engineering and Applications*, vol. 43, no. 10, pp. 12-17, 2007.
- [6] Y. Li, H. Wang, and J. Yang, "Web document clustering algorithm based on semantic similarity," *Journal of Hefei University of Technology*, vol. 32, no. 12, pp. 1846-1850, 2009.
- [7] B. L. Yang and K. Y. Shao, "Web document clustering algorithm based on high performance feature selecting function," *Application Research of Computers*, vol. 26, no. 2, pp. 546-551, 2009.
- [8] P. Zhang, F. Yang, and S. Lu, "Clustering-Based ontology block matching approach," *Journal of Jilin University (Science Edition)*, vol. 49, no. 5, pp. 493-499, 2011.
- [9] Y. J. Zhang, Y. P. Ren, L. C. Chen, and B. H. Xie, "Component clustering algorithm based on semantic similarity and optimization," *Computer Engineering and Design*, vol. 31, no. 11, pp. 2531-2537, 2010.
- [10] J. G. Sun, G. J. Huang, and J. L. Luo, "Modified concept similarity algorithm Computer Engineering and Applications," *Computer Engineering and Applications*, vol. 45, no. 5, pp.154-160, 2009.
- [11] T. Kanungo, D. M. Mount *et al.*, "A local search approximation algorithm for K-Means clustering," in *Proc. 18th Annual ACM Symposium on Computational Geometry*, no. 2-3, 2004, pp. 1-25.
- [12] S. Y. Wu and Y. Y. Wu, "Chinese and English word similarity measure based on Chinese WordNet," *Journal Zheng zhou Univ. (Nat. Sci. Ed.)*, vol. 42, no. 6, pp. 66-71, 2010.
- [13] T. Lu, H. Wang, and H. L. Yao, "K-nearest neighbor Chinese text categorization algorithm based on center documents," *Computer Engineering and Applications*, vol. 47, no. 2, pp. 127-132, 2011.
- [14] D. M. Deng, J. Z. Long, and X. Z. Yin, "A clustering algorithm based on structured Web document," *Journal of Central South University (Science and Technology)*, vol. 41, no.10, pp. 1871-1875, 2010.
- [15] Y. Z. Qu, W. Hu, and G. Cheng, "Constructing virtual document for ontology matching," in *Proc. 15th International Conference on World Wide Web*, New York: ACM, 2006, pp. 23-31.

Laizhong Cui was born in 1984. He received his B.S. degrees from Jilin University in 2007 and Ph.D. degrees from the Department of Computer Science and Technology, Tsinghua University in 2012. Since 2012, he has been a lecturer in College of Computer Science and Soft Engineering, Shenzhen University, P.R. China. His main research and engineering interests include P2P networks, distributed systems, network coding, cloud computing and social network.

Nan Lu is a professor at the college of computer and software of Shenzhen University. He is a doctor graduated from the college of computer of Jilin University. He is a member of China Computer Federation (CCF), and he is science and technology plan evaluation expert of Shenzhen, Guangdong province. He has published more than 20 articles at international conference and core journals, including SCI/EI 15 articles, and has three published books. His main research fields include modern network technology, computational intelligence, and intelligent information processing. His research interests include commerce intelligence, machine learning, complex network community structure, and trust mining of social network.