Alexandria Engineering Journal (2021) 60, 4819–4828

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

Module structure detection of oracle characters with similar semantics

Qingju Jiao [a,b,c,*], **Yuanyuan Jin** [a], **Yongge Liu** [a,b,c], **Shengwei Han** [a,c], **Guoying Liu** [a,b,c], **Nan Wang** [a,b], **Bang Li** [a,c], **Feng Gao** [a,b,c]

[a] *School of Computer & Information Engineering, Anyang Normal University, Anyang 455000, China*
[b] *Key Laboratory of Oracle Bone Inscriptions Information Processing, Ministry of Education of China, Anyang 455000, China*
[c] *Key Laboratory of Oracle Information Processing in Henan Province, Anyang 455000, China*

Received 16 December 2020; revised 2 March 2021; accepted 28 March 2021
Available online 8 April 2021

KEYWORDS

Complex network;
Module structure;
Oracle character;
Semantics

Abstract It is a highly productive approach to analyze human languages with complex networks, because the language network can effectively predict unknown features. One of the classical features in complex networks is module (or community) structure, which features dense intra-group connections and sparse inter-group connections. This work aims to identify oracle characters, the oldest characters in China, with similar semantic through module structure detection because about two-thirds of all oracle characters remain unknown. Firstly, two oracle character networks are constructed based on the context and shape of oracle rubbings and characters. Then, three module structure detection methods are employed to mine the modules from the oracle character networks. Through the experiments on the oracle character networks, three significant results are obtained. The first one is that oracle characters (or oracle variant characters) with similar semantics can be discovered by module structure, and the second one is that context and shape are of equal importance to the semantic formation of oracle characters. The last result is that modules have strong local topology of the network.

1. Introduction

Oracle bone inscriptions are the earliest mature writing system discovered in China [1]. Oracle characters, which appeared in Shang Dynasty (circa 1,250–1,046BC) are the oldest characters in China. These characters are usually carved on cattle bones or turtle shells (Fig. 1). The research of oracle characters has a great significance to archeology and philology. Therefore, some computer technologies have been adopted to study oracle characters. These researches primarily focus on oracle character recognition, structure of oracle character analysis, oracle character dataset collection and oracle bone fragments rejoining.

Compared with modern Chinese characters, oracle characters also have structural characteristics. Therefore, some literatures have been appeared to recognize oracle characters [1–11] or study structures of oracle characters [12,13]. In 2015,

* Corresponding author.
E-mail address: qjjiao@aynu.edu.cn (Q. Jiao).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2021.03.072

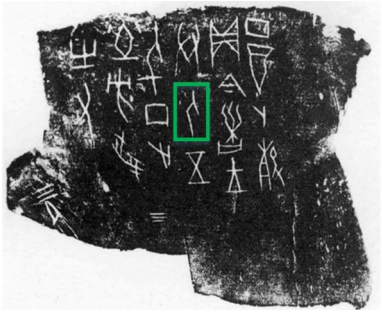

Fig. 1 An example of oracle rubbing, the character in the green rectangle is an oracle character that represents 'people' in modern Chinese.

Guo et al. [2] combined a Gabor-related low-level representation with a sparse-encoder-related mid-level representation into a novel hierarchical representation to recognize oracle characters. In 2018, Liu and Gao [3] extracted image features by 5 convolutional layers with small kernel size, obtained the final description of oracle characters with 2 fully-connected layers, and recognized oracle characters with the aid of convolutional neural network (CNN). In 2019, Meng et al. [4] first applied the pre-processing of binarization, changing brightness and contrast, and rotation in oracle bone images, and then employed single shot multibox detector to recognize oracle characters. In fact, getting oracle characters from the rubbings of oracle bones is the first step for recognizing them. To solve the problem, Liu et al [5] proposed an image segmentation method based on fully convolutional networks (FCN) to obtain oracle characters from images of oracle rubbings. In 2020, Chen et al. [6] proposed an encoding-based oracle bone script recognition (OBSR) system that applies image preprocessing techniques to encode oracle images into small matrices and recognize oracle characters in the encoding space, and the method achieved a high accuracy rate of 99% within a time range of milliseconds on the oracle bones from the Yin Ruins in XiaoTun village. Sun et al. [7] introduced a dual-view oracle bone script recognition system by combining the character recognition algorithm and temporal-spatial psychovisual modulation. The system recognized not only oracle bone inscription images but also handwriting oracle character images.

In order to recognize oracle variant characters which are different shapes of the same characters [1], Gao et al. [1] presented a two-stage method to distinguish them. In the method, the oracle variant characters were identified by computer related methods in the first stage, and then the unrecognized oracle variant characters in the first stage were further identified by multi-domain methods combining a priori knowledge. Likewise, Liu et al. [8] proposed an image retrieval method combining deep neural networks (DNN) and clustering technology to divide the oracle characters into different subsets of variants.

Like Arabic [14], the huge challenge for recognizing oracle characters are data limitation and imbalance [11]. Therefore, Yang et al. [9] used gaussian pyramid (GP) to preprocess oracle rubbing images, and analyze different generated models based on generative adversarial networks (GANs). Zhang et al. [10] used a convolutional neural network to map the character images to an euclidean space, and then nearest neighbor classification was performed for oracle character recognition. Han et al. [11] proposed a novel data augmentation approach, named Orc-Bert augmentor pre-trained by self-supervised learning, for few-shot oracle character recognition.

In the aspect of structure of oracle character analysis, Dress et al. [12] analyzed oracle characters for animals from the cognitive perspective, and applied the SplitsTree method as encoded in the NeighborNet algorithms, creating a family of dissimilarity-based networks. Yang et al. [13] used convolutional neural networks to analyze the evolution of the oracle characters based on oracle radicals. For oracle character dataset collection, Huang et al. [15] constructed a large dataset of oracle characters called OBC306, and evaluated the dataset based on the standard deep CNN, which serves as the benchmark model for oracle recognition. Li et al. [16] established a handwriting oracle character database called HWOBC, containing 83,245 character-level samples which are grouped into 3881-character categories. Both OBC306 and HWOBC are downloaded (http://jgw.aynu.edu.cn/DownPage) from the database 'Qin Qi Wen Yuan' (http://jgw.aynu.edu.cn). The database of 'Qin Qi Wen Yuan' is a specific database model (like the literature [17]) for oracle bone inscriptions, and can be performed semantic image retrieval [18,19] between images of oracle rubbings and oracle characters. For oracle bone fragments rejoining, Chen et al [20] presented a multiregional convolutional neural network to classify the oracle rubbings, and these results made contributions to the progress of the study of oracle bone morphology and oracle bone fragments rejoining. Zhang et al. [21] developed a software tools named AI-powered OBI for the annotation of oracle bone inscriptions and rejoining of oracle bone fragments.

In fact, the biggest challenge to oracle bone inscriptions is that two-thirds of oracle characters have not been recognized. That is, the semantics of these oracle characters remain unknown. To study the semantics of unknown oracle characters, this paper abstracts the relationships among oracle characters as a language network, and classifies oracle characters with similar semantics by module structure. Actually, the human language, as a complex system, can be modeled and analyzed as a complex network [22] or a graph [23]. Many related words or other linguistic components can be constructed and examined within the context of semantics, syntactics, and co-occurrence.

In 2001, Cancho and Sole [24] first constructed the co-occurrence network of the English language, and discovered the small-world effect and scale-free degree distribution in the co-occurrence network. In 2004, Cancho [25] constructed a syntactical network for three languages, and statistically analyzed the network from the aspects of degree distribution, hierarchical organization, and clustering coefficient. Steyvers and Tenenbaum [26] investigated the large-scale structure of three types of semantic networks, namely, word associations, WordNet, and Roget's Thesaurus, and found that, similar to other natural complex networks, these semantic networks have a small-world structure, characterized by sparse connectivity, short average path lengths between words, and strong local clustering. In 2009, Cech and Macutek [27] compared two syntactic dependency networks based on the same Czech language dataset, and obtained some interesting results. Rather than the structural features of language networks, Arbesman et al [28] explored the difference between English and Spanish in word

formation, using language networks constructed by small components called islands, and drew the following conclusions: Spanish words in the islands tend to be phonologically and semantically similar, but English words in the islands only have phonological similarity. In 2017, Dautriche et al. [29] constructed and analyzed four language networks, and discovered that all the networks own minimal pairs, average Levenshtein distance, and several network properties. In 2019, Liang and Wang [30] constructed 206 co-occurrence networks of Chinese characters and words, discussed the relationships among the statistical parameters in these networks, and found the spectral behavior of the modern Chinese linguistic topology is consistent over time unless something having a major impact on Chinese language happen. To represent texts as network, Arruda et al. [31] proposed a network model based on the similarity between the content of the paragraphs in the text, and found that real texts tend to have a more well-defined community structure.

Although artificial intelligence algorithms have been applied to research oracle bone inscriptions and achieved some meaningful results, few algorithms were used to predict or analyze semantics of oracle characters. As an effective tool for abstracting complex systems, complex networks are widely used to study human language. For example, based on the concept of the module (or community) structure in the network, Siew [32] digged out 17 modules of different scales in the established phonological network, and found that larger communities tend to consist of short, frequent words of high degree and low age of acquisition ratings, and smaller communities tend to consist of longer, less frequent words of low degree and high age of acquisition ratings. From the results of literature [32], we can see that the words with similar attributes may tend to gather and form modular structure in language networks. According to this assumption, we employ module structure in oracle character network to analyze oracle characters with similar semantics.

This work firstly constructs two networks for oracle characters, including context network and shape network (the flow chart see Fig. 2). Then, two subnetworks in which nodes represent known oracle characters are abstracted from context network and shape network. Third, three module detection methods are employed to mine modules from each subnetwork, and the predicted modules are compared with metadata by normalization mutual information (NMI) [33]. At last, the factors including the context and shape of oracle characters are analyzed in predicting the semantics of oracle characters by fusing context and shape subnetworks. The research results can provide vital data support to the design of semantic prediction algorithms for unknown oracle characters.

2. Construction of oracle character context network

Co-occurrence network is constructed by modeling the linear ordering of words in a corpus [34]. In a co-occurrence network, every node represents a word, and each pair of adjacent words are connected by an edge [31]. To generate a co-occurrence network, it is necessary to select a vital parameter window m_n. The co-occurrence window m_n of size n can be defined as a set of n subsequent words from a text [35]. Within a window, the edges need to be established between the first word and the n-1 subsequent words. Words are also linked according to the optional usage of specified delimiters. In the co-occurrence network, the parameter l is the length of sentence, and the edge is limited to the sentence borders. In general, the m_n is set as 2 or l. Fig. 3 shows an example for constructing a co-occurrence network, where, $w_1,\ldots,w_6$ are words in a sentence.

To build a co-occurrence network (named context network) for oracle characters, an oracle character network is established based on 72,151 pieces of oracle rubbings, and modeled into an oracle character network. The oracle characters in the same rubbing are treated as the words in a sentence of modern Chinese. Considering the serious damage of the oracle rubbings over 3,000 years, only the 71,455 oracle rubbings with oracle characters are selected. In total, these oracle rubbings contain 6,199 oracle characters, including 1,602 known semantics characters and 4,597 unknown semantics characters. Then, the weight (link) between two oracle characters is defined. The weight calculation depends on oracle rubbings, because each oracle rubbing depicts a complete semantic unit (e.g. a war or a hunting). In the same oracle rubbing with t oracle characters $\mathrm{r} = [O_1, O_2, \cdots, O_t]$, the weight between two oracle characters can be obtained by Eqs. (1) and (2) [36].

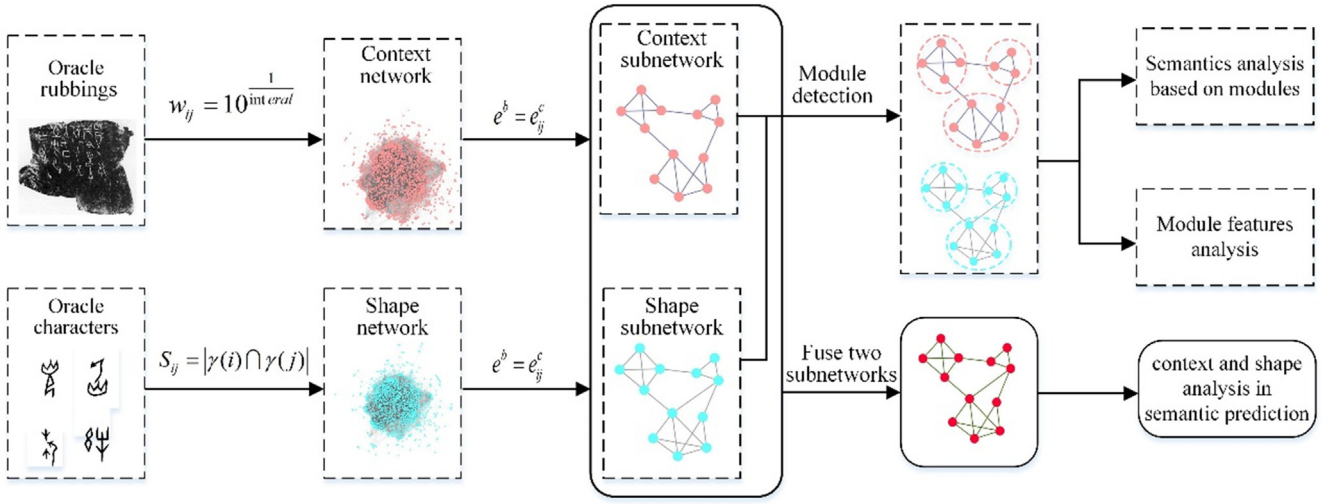

Fig. 2 The Flow chart for detecting and analyzing modules in oracle character networks.

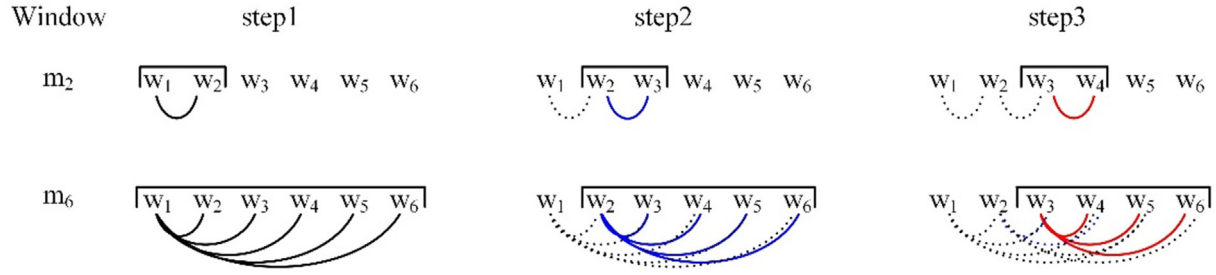

Fig. 3 An example of three steps in a network construction with a co-occurrence window m_n of sizes $n = 2$ and $n = 6$ [35].

$$w_{ij} = 10^{\frac{1}{interal}} \tag{1}$$

$$interal = \begin{cases} l_j - l_i \\ \beta + (l_j - l_i) \end{cases} \tag{2}$$

where, w_{ij} is the weight between oracle characters i and j, int*eral* is a parameter defined in formula (2); l_j and l_i are locations of oracle characters i and j in the same oracle rubbing, i.e., $l_j > l_i$ and (i, j) ∈ r; β is a user-defined variable. If fragmentary oracle character(s) exist between oracle characters i and j, the value of int*eral* includes two parts: l_j-l_i, and β. The value of β is set to 2 through experiments.

Following formulas (1) and (2), a weighted oracle character network (named context network, see Table 1) is constructed based on 71,455 oracle rubbings. The weighted network can be represented by a matrix **M** with 6,199 nodes. The network construction involves the following steps (see Fig. 4). Step 1, calculate the weight $w_{ij}^{r^1}$ between oracle characters i and j in the same rubbing r^1 by formulas (1) and (2). Step 2, set the value of $\boldsymbol{M}_{ij}^{r^1}$ as $w_{ij}^{r^1}$. Step 3, if oracle characters i and j appear simultaneously in another oracle rubbing r^{m}, calculate the weight $w_{ij}^{r^m}$ between them by formulas (1) and (2). Then, update the value of $\boldsymbol{M}_{ij}^{r^1}$ to $\boldsymbol{M}_{ij}^{r^m}$ ($\boldsymbol{M}_{ij}^{r^m} = \boldsymbol{M}_{ij}^{r^m} + w_{ij}^{r^1}$). Step 4, repeat Steps 3, calculate all the weights between oracle characters i and j for the 71,455 rubbings, to get the final value of $\boldsymbol{M}_{ij}$. All the elements of matrix **M** can be obtained by the same method. Fig. 4 illustrates the calculation of elements in matrix **M**. The two oracle rubbings r^1 and r^2 in Fig. 4 have 6 and 5 oracle characters, respectively. Based on the two oracle rubbings, a matrix **M** can be constructed with 9 nodes. For example, the value of w_{24} can be calculated by formulas (1) and (2): $w_{24} = 10^{\frac{1}{interal}} = 10^{\frac{1}{l_4 - l_2}} = 10^{\frac{1}{2}}$. But the calculation of w_{56} is more complex than that of w_{24}. First, the weight of $w_{56}^{r^1}$ is computed by oracle rubbing r^1: $w_{56} = 10^{\frac{1}{interal}} = 10^{\frac{1}{l_6 - l_5}} = 10^{\frac{1}{1}}$. Then, $w_{56}^{r^2}$ is also calculated by oracle rubbing r^2. After that, the value of $\mathbf{M}_{56}$ is obtained as the sum of $w_{56}^{r^1}$ and $w_{56}^{r^2}$.

3. Construction of oracle character shape network

The oracle characters are written by hand. Some of them are hieroglyphs. Therefore, the semantics of an oracle character can be reflected by its shape. To classify oracle characters with similar semantics, it is necessary to construct an oracle character shape network. First, 1,806 small components of oracle characters are collected. Each oracle character i can be represented by some small components: $\gamma(i)$: g_1, g_2, g_3…. Let $\gamma(i)$ and $\gamma(j)$ be the sets of small components of oracle characters i and j, respectively. Then, the weight between i and j can be calculated based on the hypothesis that two oracle characters are more likely to have similar semantics if they have many common small components. There are many similarity indexes to calculate the weight between i and j [37], here we adopt a simple index common neighbors (CN, formula 3 and Fig. 5) [37]. Based on the weights obtained by formula 3, an oracle character shape network (named shape network, see Table 1) of 5,890 nodes and 1,198,652 edges is established.

$$S_{ij} = |\gamma(i) \cap \gamma(j)| \tag{3}$$

4. Abstraction of subnetworks

To verify the semantics of the oracle characters in the same module, a subnetwork of oracle context network is extracted in the following steps. First, select 1,602 oracle characters with known semantics (i.e. known oracle characters). Second, retain each edge in the context network, if the two nodes connected by the edge are known oracle characters (formula 4). Third, repeat the second step until all edges are processed. In this way, a subnetwork (named context subnetwork, see Table 1)

Table 1 Basic properties of four oracle character networks.

Name	Nodes	Edges	Class	Link density	Diameter	Average path length	Average degree	Clustering coefficient
Context network	6199	232,695	—	0.0121	5	1.5819	75.0629	0.8235
Context subnetwork	1,392	41,821	135	0.0432	5	1.4335	60.0876	0.7175
Shape network	5,890	1,198,652	—	0.0691	6	1.0972	386.7243	0.6827
Shape subnetwork	1,428	57,985	132	0.0569	8	1.0749	81.2115	0.7211

Note that, if the network is disconnected, the diameter is the maximum of diameters of all connected components, and average path length is the mean of average path lengths of all connected components.

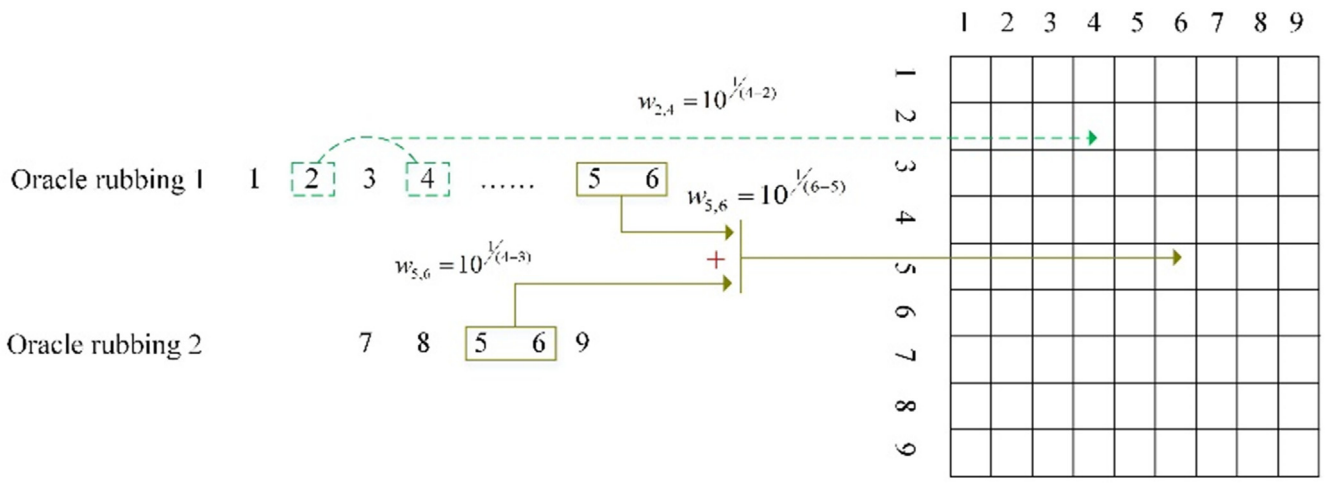

Fig. 4 An example of illustrates the calculation of elements in matrix **M** [36], '.....' represents fragmentary oracle character(s).

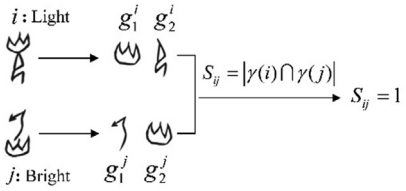

Fig. 5 An example for calculating common neighbors between two oracle characters.

can be obtained with 1,392 nodes and 41,821 weighted edges by deleting the edges whose weight is smaller than 5. According to the scenes described by oracle characters, the 1,392 oracle characters in the context subnetwork are divided into 135 classes, creating a standard metadata to measure modules mined by other methods. Similarly, a subnetwork with 1,428 nodes and 57,985 weighted edges were abstracted from the oracle character shape network (named shape subnetwork, see Table 1). Then, the shape subnetwork are divided into 132 classes by semantics, providing the metadata to measure the modules detected by various methods.

$$e^b = \begin{cases} e^c_{ij} \ subjec\ to\ (i,j) \in U \\ \quad null\ otherwise \end{cases} \tag{4}$$

where, e^b is the set of edges of subnetwork, e^c is the total edges in context network. U is the set of known oracle characters, and *null* denotes that e^b is null.

In order to further understand the four oracle character networks constructed by this work, their basic properties [38] including link density (formula 5), diameter, average path length, average degree and clustering coefficient are calculated, and the results are shown in Table 1. From link density, we can see that the context network is sparser than other three networks, possibly because many isolated nodes exist in context network. The diameter and average path length can reflect the small-world effect [39] of network. Comparing other real-world networks [38], these four oracle characters also have small-world effect. As the diameter shows, the four networks may have the phenomenon of six degrees of separation [40]. For the average degree, the shape network is the largest in four networks. The values of link density and average degree imply that the shape network is the most densely network because different oracle characters have more overlapped small components. Unlike the properties mentioned above, clustering coefficient has been used to measure the local topology of network, that is the larger the clustering coefficient value, the stronger the local of the network. As the table shows, the clustering coefficients of four networks are quite large, which provides data support for designing module detection algorithms.

$$LD = \frac{2E}{N \times (N-1)} \tag{5}$$

where E and N are total number of edges and nodes in network, respectively.

5. Results and discussion

5.1. Modules in context and shape subnetworks

In oracle bone inscriptions, the oracle characters with similar semantics tend to form a class. In order to mine the oracle characters with similar semantics, the oracle character subnetworks are divided into classes, and the predicted classes are evaluated against real classes using an index. Firstly, the modules are mined from the subnetworks. Module is a classic feature of complex network. As shown in Fig. 6 [41], each module is a subnetwork with dense intra-group connections and sparse inter-group connections. Module structure has been applied to many fields, ranging from biology, computer science, to sociology. Based on module structure, it is possible to predict new attributes of an object (or a node in a network). Suppose the attribute of node 12 in Fig. 6 is unknown, while that of nodes 8–11 are known. Then, node 12 is highly likely to have the same attribute as nodes 8–11, because nodes in the same module tend to have the same attribute. Next, the NMI [33] is employed to evaluate the classes predicted by different module detection methods. Let $\chi = (X_1, X_2, \cdots, X_{n_X})$ and $\gamma = (Y_1, Y_2, \cdots, Y_{n_Y})$ be two partitions of a network, and n_X and n_Y be the number of modules in the two partitions, respectively. Then, the NMI can be defined as:

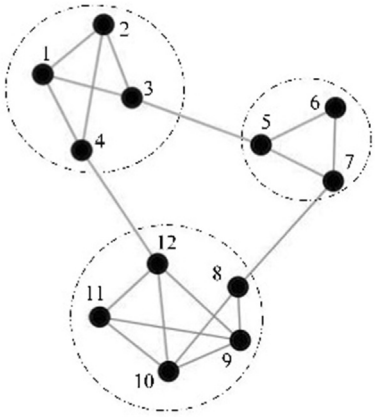

Fig. 6 A 12-node 19-edge small network divided into 3 modules [41]

$$NMI(\chi,\gamma)=\frac{-2\sum_{i=1}^{n_X}\sum_{j=1}^{n_Y}n_{ij}^{XY}\log\left(\frac{n_{ij}^{XY}\cdot N}{n_i^X\cdot n_j^Y}\right)}{\sum_{i=1}^{n_X}n_i^X\log\left(\frac{n_i^X}{N}\right)+\sum_{j=1}^{n_Y}n_j^Y\log\left(\frac{n_j^Y}{N}\right)} \tag{6}$$

where, N is the number of nodes in network; n_i^X and n_j^Y are the number of nodes in modules X_i and Y_j, respectively; n_{ij}^{XY} is the number of nodes shared by modules X_i and Y_j: $n_{ij}^{XY}=|X_i\cap Y_j|$. The larger the NMI, the better the partitioning of modules.

In order to detect the modules in the oracle character subnetworks, three methods are adopted to detect the modules, namely, ClusterONE method [42], GCDanon method [43] and Moduarity method [44]. The ClusterONE method centers on a concept called the cohesiveness score (see formula 7). This method searches for modules in the network by a greedy growth process. It can detect partly overlapped modules from unweighted and weighted networks. Here, the ClusterONE method embedded in Cytoscape [45] is selected to detect the modules in oracle character networks. The parameter of minimum size in ClusterONE method is set to 3, the haircut threshold is set to 0.1, and the other parameters are as default. Both GCDanon and Modularity methods aim to optimize the modularity Q (see formula 8). The GCDanon method is extended from Newman's algorithm for community detection, which treats communities of different sizes on an equal footing. The Modularity method, which is based on global structure of network, is immensely popular.

$$f(V)=\frac{w^{in}(V)}{w^{in}(V)+w^{bound}(V)+p|V|} \tag{7}$$

$$Q=\sum_{V=1}^{n_V}\left[\frac{W_V}{W}-\left(\frac{S_V}{2W}\right)^2\right] \tag{8}$$

In formulas (7) and (8), V is a module, $w^{in}(V)$ is the total weight of edges in V, $w^{bound}(V)$ is the total weight of edges that connect the module with the rest of the network, $p|V|$ is a penalty. n_V is the number of modules, W is the total weight of edges in the network, W_V is the total weight of the internal edges in V and S_V is the total strength of the nodes in V.

Using the three methods, the modules in context and shape subnetworks are detected. Then, the detected modules are compared against the metadata. Finally, the NMIs between the modules and metadata are calculated (see Fig. 7). As shown in Fig. 7, the highest NMIs of modules in context and shape subnetworks are discovered by ClusterONE method with 0.4642 and 0.5011. This means the module structure in the network can reveal the oracle characters with similar semantics. The NMIs of GCDanon and Modularity methods are lower than those of ClusterONE method, they are 0.3544 and 0.3006 in context subnetwork, and 0.3294 and 0.3032 in shape subnetwork. A possible reason is that ClusterONE is defined by the local structure of network, while Modularity and GCDanon are defined by the global structure of network.

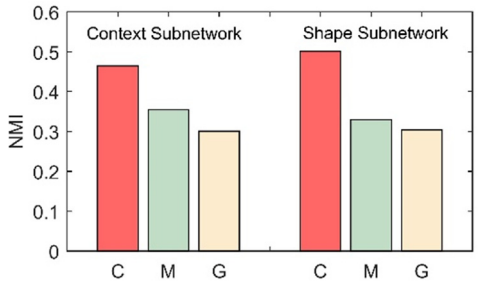

Fig. 7 The NMIs of the three methods on context and shape subnetworks, C, M and G represent ClusterONE, Moduarity and GCDanon methods, respectively.

Next is a detailed analysis on the semantics of oracle characters in the same modules. To measure the matching degree between the detected modules and real semantic units, a match degree index (C) can be defined as formula 9, and can be calculated as following steps. First, the i-th detected module is compared with all the real semantic units one by one, and then the maximum value of the intersection is the C value of i-th detected module. In the context subnetwork, the first example module 18 (see Fig. 8) with 0.5333 of C value contains 15 characters representing digits (e.g. 3, 4, 5, 6, 7, and 8) and digit-related oracle characters. Module 55 with 0.6829 of C value contains 82 oracle characters. These oracle characters describe

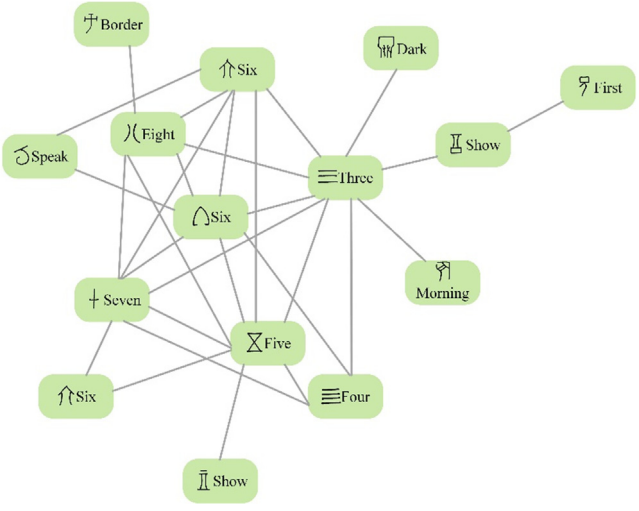

Fig. 8 The structure of module 18 with 15 oracle characters including two oracle variant characters 'Six' and 'Show', note that the 'Dark' and 'Show' represent the six-related word and the two-related word in oracle bone inscriptions, respectively.

a cross-shaped things, such as a cross-shaped weapon. Module 61 contains 10 characters and its C value is 0.5, these oracle characters depict edible plants like wheat and rice. The results show that the context is critical to the formation of semantics of non-hieroglyphs, and the semantics of unknown non-hieroglyphs can be predicted by context.

$$C_i = \max_{1 \leqslant j \leqslant n} \frac{|P_i \cap R_j|}{|P_i|} \tag{9}$$

where, P_i and R_i are the detected module and real semantic unit, respectively.

In the shape subnetwork, 101 modules are discovered by ClusterONE method. Among them, 43 modules have a C value greater than 0.5, revealing the importance of shape to the formation of semantics of oracle characters. For example, an interesting module is module 50 (Fig. 9), the module includes 12 oracle characters and its C values is 1. The results means that all the oracle characters in the module describe the door-related things or motions or some scenes, for example, some oracle characters have the meaning of protection mainly because the strong door can defend against enemies or beasts in ancient China. Likewise, some oracle characters in module 50 also describe warehouse because the things in the warehouse with solid door cannot be stolen. Module 8 (C value is 0.667) includes 24 characters, two-thirds of which depict the things related to the house, such as family (the people in the same family should live the same house in ancient China), room, and place. In addition, an oracle character in this module describes safeness because people living in a solid house means safeness in ancient China. Module 5 with C = 0.6721 has 61 oracle characters, most of which describe things or people related to woman, e.g. sister, and pregnancy. Module 23 contains 12 characters and its C value is 0.75. Most of them describe the things associated with money, e.g. greed, stored things, and jewelry. The other big module is module 36, this module contains 58 modules with C value of 0.5833. The oracle characters in the module describes foot-related things, motions, or scenes, for example, they describe the scenes that people run away from house or some children play or chase each other. The last instance of module is module 89, the module includes 8 oracle characters with high C value of 0.875. The oracle characters in the modules principally describe some fruits of cereals.

The above results demonstrate that the shape of oracle characters plays a more vital role in the formation of semantics than the context. In fact, many oracle characters are hieroglyphs that convey semantics in shape. Hence, it is possible to predict the fuzzy semantics of an unknown oracle characters by shape, and then infer accurate semantics by context.

5.2. Analysis of context and shape in semantic prediction

To compare the importance of context and shape to semantic prediction of unknown oracle characters, the context subnetwork is fused with the shape subnetwork in the following steps. First, only the edges e^s_{ij} stratifying the condition that its two nodes i and j are overlapped with the nodes in the context subnetwork are selected from the shape subnetwork. Next, the weight (w^f_{ij}) of edge between nodes i and j in the fusion subnetwork is calculated by equation (10). In this way, we can obtained a fusion subnetwork with 1,379 nodes and 83,066 edges.

$$w^f_{ij} = \alpha \frac{w^c_{ij}}{\max(w^c)} + (1-\alpha)\frac{w^s_{ij}}{\max(w^s)} \tag{10}$$

where, α is a parameter in the range of (0, 1), w^c_{ij} is the weight between nodes i and j in the context subnetwork, $\max(w^c)$ is the maximum weight in the context subnetwork. w^s_{ij} is the weight between nodes i and j in the shape subnetwork and $\max(w^s)$ is the maximum weight in shape subnetwork.

Because of its excellence in module detection, ClusterONE method is adopted to detect the modules in the fusion subnetwork. The detected modules are compared with metadata, and subject to NMI calculation. Further, parameter α is adjusted into α_1, α_2, …, α_n to obtain different network modules P_1, P_2, …, P_n. These modules are also compared with the metadata, and subject to NMI calculation (NMI_1, NMI_2, …, NMI_n).

Fig. 10 illustrates the relationship between α, NMI, and the number of modules. It can be seen that, as α increased from

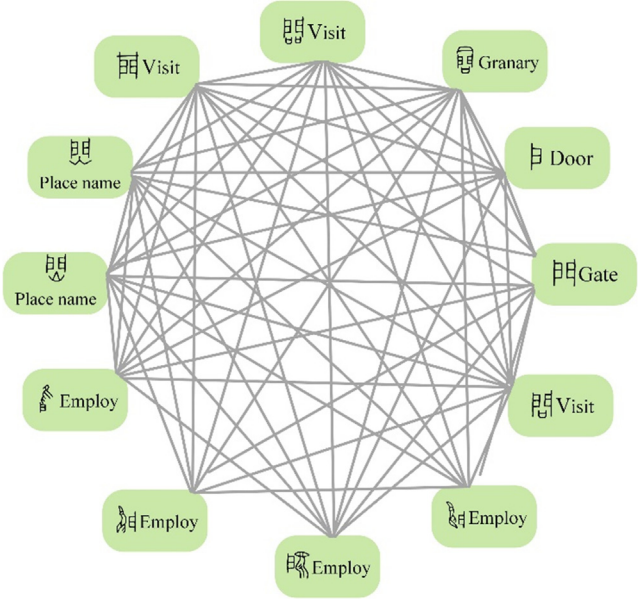

Fig. 9 The structure of module 50 with 12 oracle characters including three oracle variant characters 'visit', 'employ' and 'place name'

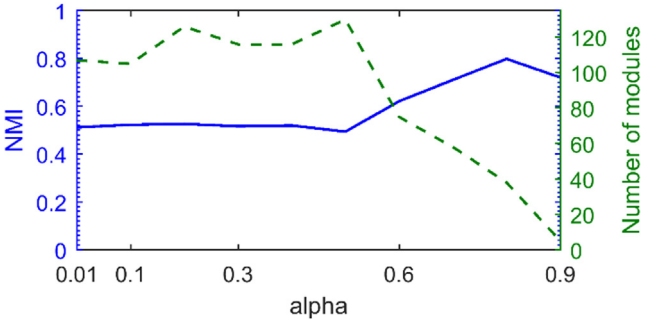

Fig. 10 The relationship between α, NMI, and the number of modules.

0.01 to 0.5, Both the number of modules and NMIs in the fusion subnetwork remained stable, suggesting that context plays the main role in predicting the semantics of non-hieroglyphic oracle characters, and that the property of module structure for context subnetwork is not disturbed by adding shape factor. When α increased from 0.6 to 0.9, the NMIs continued to grow, indicating that NMI accuracy can be improved by adding context information into the shape subnetwork. But the NMI accuracy was not good by adding shape information into the context subnetwork. Different from the values of NMI, the number of module is decreased rapidly when α is set from 0.6 to 0.9 because of two possible reasons. The first one is that the property of module structure for shape subnetwork become very weak by adding context factor. The other one is that the ClusterONE method may discard some nodes without module structure. Therefore, a reasonable way is to predict the semantics of unknown oracle characters by shape information, and then validate the semantics by context information. This step-by-step semantic prediction approach is consonant with the traditional way to infer semantics of unknown oracle characters. Specifically, shape and context information are not fused to predict semantics. For example, when α equaled 0.5, the NMI (0.4941) reached the minimum.

5.3. Module features in oracle character networks

The module features in oracle character networks provide data and theoretical supports for algorithm design. Here, the multi-scale algorithm is introduced to divide oracle character networks into modules with different sizes. The modules are mined from oracle character networks through spectral factorization [46]. The greatest merit of the method is the flexible setting of the number of modules.

For context and shape subnetworks, the number of modules was adjusted from 1 to 200 for spectral factorization. Then, the two subnetworks are divided into different modules (1–200). Finally, the obtained modules are evaluated against metadata with NMIs. Fig. 11 shows the NMIs of the different number of modules (1–200) mined by spectral factorization. In general, when the number of modules mined is small (the size of module is large), the NMIs are relatively small for the two subnetworks mainly because the size of real modules in oracle character networks is small. In fact, only several oracle characters can describe an entire semantic scene in oracle bone inscriptions. The largest NMIs (0.5059 and 0.6094) appeared, when the number of modules reached 193 and 195 for context and shape subnetworks, respectively. Overall, the modules in oracle character networks have two main features: the modules reflect the local topology of the network, and the module accuracy depends on the quality of module detection method.

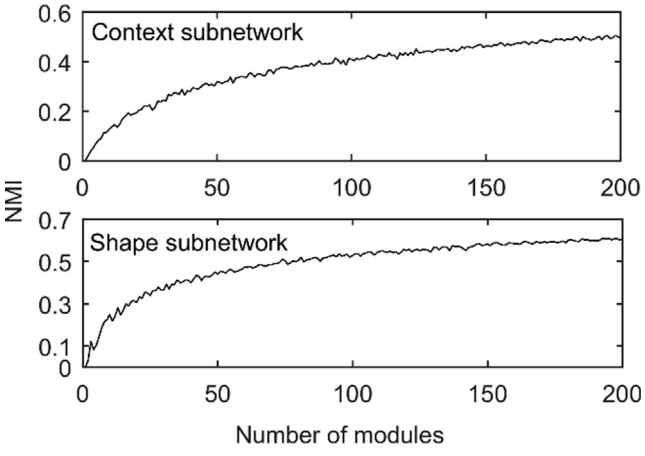

Fig. 11 The NMIs of modules mined by spectral method with different number (1 ~ 200) of modules.

6. Conclusions

Oracle bone inscription is the oldest character system in China. In oracle bone inscriptions, the semantics of about two-thirds of oracle characters are still unknown. Inferring the semantics of unknown oracle characters would greatly promote the research into oracle bone inscriptions and the understanding of the ancient history of China. Considering the significance of context and shape to sematic formation of oracle characters, this work abstracts the context of oracle characters as a complex network based on oracle rubbings, and abstracts the shape of oracle characters as complex network based on the components in such characters. Next, two subnetworks, namely, context subnetwork and shape subnetwork, with known oracle characters are separately extracted from the context and shape networks. After that, three module detection methods are employed to mine modules from the context and shape subnetworks, and the modules with oracle characters of similar semantics are analyzed in details. Then, the roles of context and shape in the semantic prediction of oracle characters are investigated by fusing the two subnetworks. Finally, some features of the modules in oracle character network are examined, revealing that the modules reflect the local topology of the network, and the module accuracy depends on the quality of module detection method. The above results show that oracle characters, despite their long history, have laws of complex system like modern Chinese characters. Therefore, oracle bone inscriptions can be studied by computer-based methods.

Albeit the interesting results, there is ample room for improving the community detection methods in this work. In recent years, as an effective method of artificial intelligence, deep learning [47] is applied to different fields. Such as disease diagnosis [48,49], intrusion detection [50] and image recognition [51]. Likewise, deep learning has become a new trend in community detection, because it can encode feature representations of high-dimensional data [52] and learn the pattern of nodes, neighborhoods, and subgraphs [53], and so on [54]. Several community detection methods have emerged based on deep learning. For example, Xin et al. [55] designed a structured deep convolutional neural network to detect communities from topologically incomplete networks (TIN). To deal with highly sparse matrices, Sperli [56] described a novel community detection method based on a deep learning. Apart from CNN, Generative adversarial network and graph neural network (GNN) have also been introduced to community detection. Jia et al. [57] first define an embedding that indicates the membership strength of vertices to communities. Second, a specifically designed GAN is adopted to optimize such embedding, and to mine communities. As with deep neural network, graph neural network is also adopted to detect communities. Chen et al. [58] propose a novel family of GNNs for solving community detection problems in a supervised learning

setting. Jin et al. [59] solve the problem of semi-supervised community detection in attributed networks with semantic information by combining graph convolutional networks (GCNs) [60] and markov random fields (MRF) [61]. Likewise, Shchur and Gunnemann [62] propose a GNN-based model for overlapping community detection. Since the semantics of about a third oracle characters are known, it is possible to combine deep learning (especially GCNs) with the prior knowledge like the semantics of known oracle characters to improve the accuracy of semantic prediction.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors acknowledge the supports from the National Natural Science Foundation of China (No. 61806007, U1804153), the National Social Science Fund Major Entrusted Project of China (No. 16@ZH017A3), the Program for Changjiang Scholars and Innovative Research Team in University (No. 2017PT35).

References

[1] J.H. Gao, X. Liang, Distinguishing oracle variants based on the isomorphism and symmetry invariances of oracle-bone inscriptions, IEEE Access 8 (2020) 152258–152275.
[2] J. Guo, C.H. Wang, E. Roman-Rangel, H.Y. Chao, Y. Rui, Building hierarchical representations for oracle character and sketch recognition, IEEE Trans. Image Process. 25 (1) (2015) 104–118.
[3] G.Y. Liu, F. Gao, Oracle-Bone Inscription Recognition Based on Deep Convolutional Neural Network, J. Image Graph. 8 (4) (2020) 1442–1450.
[4] L. Meng, N. Kamitoku, X.B. Kong, K. Yamazaki, Deep Learning based Ancient Literature Recognition and Preservation, in: 2019 58th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), 2019, pp. 473–476.
[5] G.Y. Liu, X. Song, W.Y. Ge, H.Y. Zhou, J. Lv, Oracle-bone-inscription image segmentation based on simple fully convolutional networks, Eleventh International Symposium on Multispectral Image Processing and Pattern Recognition (MIPPR2019), 2019.
[6] T.Z. Chen, Y.Y. Qian, J.Y. Pei, S.T. Wu, J. Wu, L. Li, J.Y. Tu, A study on encoding-based oracle bone script recognition, J. Chin. Writing Syst. 4 (4) (2020) 281–290.
[7] W.J. Sun, G.T. Zhai, Z.P. Gao, T.Z. Chen, Y.C. Zhu, Z.D. Wang, Dual-View oracle bone script recognition system via temporal-spatial psychovisual modulation, IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) 2020 (2020) 193–198.
[8] G.Y. Liu, Y.G. Wang, Oracle character image retrieval by combining deep neural networks and clustering technology, IAENG Int. J. Comput. Sci. 47 (2) (2020) 199–206.
[9] Z. Yang, Q.Q. Zhu, Z.Y. Huang, Z.J. Yin, F. Yang, Generative adversarial networks for oracle generation and discrimitation, in: 2019 IEEE 19th International Conference on Communication Technology (ICCT), 2019, pp. 1616–1620.
[10] Y.K. Zhang, H. Zhang, Y.G. Liu, Q. Yang, C.L. Liu, Oracle character recognition by nearest neighbor classification with deep metric learning, in: International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 309–314.
[11] W.H. Han, X.L. Ren, H.Y. Lin, Y.W. Fu, X.Y. Xue, Self-supervised learning of Orc-Bert augmentator for recognizing Few-Shot oracle characters, Proceedings of the Asian Conference on Computer Vision (ACCV), 2020.
[12] A. Dress, S. Grünewald, Z.B. Zeng, A cognitive network for oracle bone characters related to animals, Int. J. Mod. Phys. B 30 (4) (2016) 1630001.
[13] Z. Yang, G.L. Xu, F. Yang, Z.J. Yin, Semantic analysis of the Oracle Radicals using Similarity Strategy based on the Yolov2 network, in: 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE), (2019) pp. 1975–1978.
[14] B. Ait Ben Ali, S. Mihi, I. El Bazi, N. Laachfoubi, A recent survey of Arabic named entity recognition on social media, Revue d'Intelligence Artificielle, 34(2) (2020) 125–135.
[15] S.P. Huang, H.B. Wang, Y.G. Liu, X.S. Shi, L.W. Jin, OBC306: A Large-Scale Oracle Bone Character Recognition Dataset, in: 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 681–688.
[16] B. Li, Q.W. Dai, F. Gao, W.Y. Zhu, Q. Li, Y.G. Liu, HWOBC-A handwriting oracle bone character recognition database, in: The 2020 second International Conference on Artificial Intelligence Technologies and Application (ICAITA), 2020, p. 012050.
[17] H. Bais, M. Machkour, Method and apparatus for querying relational and XML database using French language, Revue d'Intelligence Artificielle 33 (6) (2019) 393–401.
[18] B.X. Jia, B. Meng, W.N. Zhang, J. Liu, Query rewriting and semantic annotation in semantic-based image retrieval under heterogeneous ontologies of big data, Traitement du Signal 37 (1) (2020) 101–105.
[19] F. Gandon, A survey of the first 20 years of research on semantic web and linked data, Ingénierie des Systèmes d'Information 23 (3–4) (2018) 11–56.
[20] S.X. Chen, X. Han, W.Z. Gao, X.X. Liu, B.F. Mo, A classification method of oracle materials based on local convolutional neural network framework, IEEE Comput. Graphics Appl. 40 (3) (2020) 32–44.
[21] C.S. Zhang, R.X. Zong, S. Cao, Y. Men, B.F. Mo, AI-Powered Oracle Bone Inscriptions Recognition and Fragments Rejoining. Proceedings of the Twenty-Ninth International Joint Conference on Artifificial Intelligence (IJCAI-20), (2020) pp. 5309–5311.
[22] J. Cong, H.T. Liu, Approaching human language with complex networks, Phys. Life Rev. 11 (4) (2014) 598–618.
[23] S.F. Zhang, Classification of urban land use based on graph theory and geographic information system, Ingénierie des Systèmes d'Information 24 (6) (2019) 633–639.
[24] R.F.I. Cancho, R.V. Solé, The small world of human language, Proc. R. Soc. Lond. B Biol. Sci. 268 (1482) (2001) 2261–2265.
[25] R.F.I. Cancho, Euclidean distance between syntactically linked words, Phys. Rev. E 70 (5) (2004) 056135.
[26] M. Steyvers, J.B. Tenenbaum, The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth, Cogn. Sci. 29 (1) (2005) 41–78.
[27] R. Čech, J. Mačutek, Word form and lemma syntactic dependency networks in Czech: A comparative study, Glottometrics 19 (2009) 85–98.
[28] S. Arbesman, S.H. Strogatz, M.S. Vitevitch, Comparative analysis of networks of phonologically similar words in English and Spanish, Entropy 12 (3) (2010) 327–337.
[29] I. Dautriche, K. Mahowald, E. Gibson, A. Christophe, S.T. Piantadosi, Words cluster phonetically beyond phonotactic regularities, Cognition 163 (2017) 128–145.

[30] W. Liang, K.P. Wang, Relationships among the statistical parameters in evolving modern Chinese linguistic co-occurrence networks, Physica A 524 (2019) 532–539.
[31] H.F.D. Arruda, V.Q. Marinho, L.D.F. Costa, D.R. Amancio, Paragraph-based representation of texts: A complex networks approach, Inf. Process. Manage. 56 (3) (2019) 479–494.
[32] C.S.Q. Siew, Community structure in the phonological network, Front. Psychol. 4 (2013) 553.
[33] A. Strehl, J. Ghosh, Cluster ensembles–-a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (Dec) (2002) 583–617.
[34] W.P. Goh, K.K. Luke, S.A. Cheong, Functional shortcuts in language co-occurrence networks, PLoS ONE 13 (9) (2018) e0203025.
[35] D. Margan, A. Meštrović, LaNCoA: a python toolkit for language networks construction and analysis, in: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015, pp. 1628–1633.
[36] Q.J. Jiao, F. Gao, Y.Y. Jin, J. Xiong, Y.G. Liu, Construction and analysis of rubbing-oriented oracle character network, J. Chin. Inform. Process. 32 (7) (2020) 137–142.
[37] L.Y. Lu, T. Zhou, Link prediction in complex networks: A survey, Physica A 390 (2011) (2011) 1150–1170.
[38] M.E.J. Newman, The structure and function of complex networks, SIAM Rev. 45 (2) (2003) 167–256.
[39] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks, Nature 393 (6684) (1998) 440–442.
[40] S. Milgram, The small world problem, Psychol. Today 1 (1) (1976) 60–67.
[41] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (3–5) (2010) 75–174.
[42] T. Nepusz, H.Y. Yu, A. Paccanaro, Detecting overlapping protein complexes in protein-protein interaction networks, Nat. Methods 9 (5) (2012) 471–475.
[43] L. Danon, A. Diaz-Guilera, A. Arenas, The effect of size heterogeneity on community identification in complex networks, J. Stat. Mech: Theory Exp. 2006 (11) (2006) P11010.
[44] M.E.J. Newman, Analysis of weighted networks, Phys. Rev. E 70 (5) (2004) 056131.
[45] R. Saito, M.E. Smoot, K. Ono, J. Ruscheinski, P.L. Wang, S. Lotia, A.R. Pico, G.D. Bader, T. Ideker, A travel guide to Cytoscape plugins, Nat. Methods 9 (11) (2012) 1069.
[46] J.P. Hespanha, An efficient matlab algorithm for graph partitioning. Technical Report, USA: University of California (2004).
[47] Y. LeCun, Y. Bengio, G.E. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[48] M. Yildirim, A. Çinar, Classification of white blood cells by deep learning methods for diagnosing disease, Revue d'Intelligence Artificielle 33 (5) (2019) 335–340.
[49] M. Mohammedhasan, H. Uguz, A new early stage diabetic retinopathy diagnosis model using deep convolutional neural networks and principal component analysis, Traitement du Signal 37 (5) (2020) 711–722.
[50] G. Ketepalli, P. Bulla, Review on generative deep learning models and datasets for intrusion detection systems, Revue d'Intelligence Artificielle 34 (2) (2020) 215–226.
[51] Y. Li, D.L. Shi, F.J. Bu, Automatic recognition of rock images based on convolutional neural network and discrete cosine transform, Traitement du Signal 36 (5) (2019) 463–469.
[52] X.L. Zhang, Multilayer bootstrap networks, Neural Networks 103 (2018) 29–43.
[53] Z.H. Wu, S.R. Pan, F.W. Chen, G.D. Long, C.Q. Zhang, P.S. Yu, A comprehensive survey on graph neural networks, IEEE Trans. Neural Networks Learn. Syst. 99 (2020) 1–21.
[54] F.Z. Liu, S. Xue, J. Wu, C. Zhou, W.B. Hu, C. Paris, S. Nepal, J. Yang, P.S. Yu, Deep learning for community detection: progress, challenges and opportunities. Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20), (2020) pp. 4981–4987.
[55] X. Xin, C.K. Wang, X. Ying, B.Y. Wang, Deep community detection in topologically incomplete networks, Physica A 469 (2017) 342–352.
[56] G. Sperlí, A deep learning based community detection approach, in: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, 2019, pp. 1107–1110.
[57] Y.T. Jia, Q.Q. Zhang, W.N. Zhang, X.B. Wang, CommunityGAN: Community detection with generative adversarial nets, The World Wide Web Conference (2019) 784–794.
[58] Z.D. Chen, L.S. Li, J. Bruna, Supervised community detection with line graph neural networks, in: International Conference on Learning Representations, 2019, pp. 1–21.
[59] D. Jin, Z.Y. Liu, W.H. Li, D.X. He, W.X. Zhang, Graph convolutional networks meet markov random fields: Semi-supervised community detection in attribute networks, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-19), vol. 33 (2019), pp. 152–159.
[60] T. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks. International Conference on Learning Representations, 2017 (2017).
[61] G.R. Cross, A.K. Jain, Markov random field texture models, IEEE Trans. Pattern Anal. Mach. Intell. 5 (1) (1983) 25–39.
[62] O. Shchur, S. Günnemann, Overlapping community detection with graph neural networks. Deep Learning on Graphs Workshop, KDD, 2019 (2019).