Physica A 540 (2020) 123022

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Likelihood category game model for knowledge consensus

Zhong-Yan Fan [a,*], Ying-Cheng Lai [b], Wallace Kit-Sang Tang [a]

[a] Department of Electronic Engineering, City University of Hong Kong, Tat Chee Avenus, Kowloon, Hong Kong
[b] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287, USA

ARTICLE INFO

Article history:
Received 23 January 2019
Received in revised form 28 September 2019
Available online 3 October 2019

Keywords:
Naming game
Complex networks
Categorization

ABSTRACT

To reach consensus among interacting agents is a problem of interest for social, economical, and political systems. To investigate consensus dynamics, naming game, as a computational and mathematical framework, is commonly used. Existing works mainly focus on the consensus process of vocabulary evolution in a population of agents. However, in real-world cases, naming is not an independent process but relies on perception and categorization. In order to name an object, agents must first distinguish the object according to its features. We thus articulate a likelihood category game model (LCGM) to integrate feature learning and the naming process. In the LCGM, self-organized agents can define category based on acquired knowledge through learning and use likelihood estimation to distinguish objects. The information communicated among the agents is no longer simply in some form of absolute answer, but involves one's self perception and determination. With its distinguished features, LCGM allows coexistence of multiple categories for an observation. It also provides quantitative explanation that consensus is hard to be reached among serious agents who have a more complex knowledge formation. The proposed LCGM and this study are able to provide new insights into the emergence and evolution of consensus in complex systems.

1. Introduction

To reach consensus among a population of agents is a problem with significant applications in social, economical, and political systems. A mathematical and computational paradigm to describe, characterize, and understand consensus dynamics is naming game (NG) and its variants [1–21]. In an NG, agents attempt to reach consensus about a certain object or event via interactions. Most existing works emphasize on the consensus of vocabulary among the agents [1–20]. While vocabulary is often related to objects and plays an important role in the development of natural language, for intelligent agents, another determining factor in NG is the content or the feature of the object. In fact, a pivotal cognitive ability of human being is categorization which recognizes, characterizes, and eventually names objects according to their features. In our daily activities, majority of the naming actions are based on a category rather than a specific term or vocabulary. For example, the word "cat" refers to a category in which objects have same features rather than a specific cat "Tom" or "Kitty". To better describe consensus dynamics in the real world, it is essential to incorporate categorization through learning and knowledge growth into the model. While there were previous works on categorization games that deal with feature recognition and analysis [22–26], the process of categorization itself remains to be an outstanding topic of research, due to its high complexity. It is thus a challenging task to incorporate perception and categorization into NGs, while our aim is to introduce a new kind of category game model to address this problem.

* Corresponding author.
E-mail address: zyfan4-c@my.cityu.edu.hk (Z.-Y. Fan).

https://doi.org/10.1016/j.physa.2019.123022
0378-4371/

In previous works on NGs, consensus is achieved through some learning processes solely governed by the interactions between agents [1–3]. Therefore, topological features of the agents' interaction network, such as degree, clustering coefficient, and path distance, have significant influences, and it is concluded that higher degrees, lower clustering and shorter network distance tend to promote consensus [4–8]. The knowledge transfer in NGs is not limited between two agents, for example, Li et al. [11] suggested a model with multiple hearers while Gao et al. [13] considered negotiation to take place between multiple indirectly connected agents. An agent can also play the roles of a speaker and a hearer simultaneously [13]. More general settings were studied in which agents possess different propensities such as commitment [14,16] and stubbornness [9,12]. It is concluded that the learning behaviors of these agents can significantly affect the consensus dynamics over the whole population. In addition, other issues for more realistic agents, such as learning errors during interactions [15], memory loss [17], multi-word/language [18,19] and negative reference [20] have been investigated. These works provide great insights into consensus dynamics. However, since only vocabulary is focused, significant feature-based cognitive abilities such as categorization were not taken into account which, as natural intuition would suggest, would play a more significant role in the emergence and evolution of consensus.

There were a few proposals on categorization games. For example, discrimination and guessing games were designed to accomplish the task of categorization [22]. In a discrimination game, a speaker is trained with ground truth so that it can relate an object to the actual category using a classifier. Guessing game is similar to a typical NG, however, the hearer not only acquires the name but also updates its classifier for the category as instructed by the speaker. Based on these two types of games, the problem of color categorization was studied [22], and a category game model (CGM) was developed [23]. In a game defined by CGM, a pair of objects are presented to a speaker and a hearer, and the target topic (the object to be learned) is selected by the speaker. If both objects are distinguishable to the speaker, one is randomly chosen as the target topic. Otherwise, the speaker must discriminate the two objects by creating a new boundary between them before selecting one as the target topic. The interaction between the speaker and the hearer then makes it feasible to learn and name the target topic. In the CGM framework, factors such as language aging [24], persistence [25], and individual biases [26] can be studied. A feature common to both discrimination game and CGM is that certain pre-requisites are needed. Specifically, in a discrimination game, it is necessary to relate the object to an actual category, while in CGM, the two presented objects are assumed to belong to two different categories. In the real world, it often occurs that a population can reach an agreement without being given any ground truth, and this has consequences. For example, different language systems can give different color categorization, animals swarm in a consistent way under a distributed agreement [27], and the consensus of opinions among practitioners in a financial market can trigger a herd behavior that leads to the fat-tail distributions in prices and returns [28].

To understand and exploit consensus dynamics as realistically as possible requires the development of a more comprehensive type of game models centered about the object features. The base of our model is the recently proposed domain learning naming game (DLNG) [21], which aims to solve the color categorization problem through elimination of pre-requisites in the learning process. The color perception process in DLNG follows a variant algorithm based on the majority rule. Agents in DLNG have numerous sensors uniformly distributed in the domain, and one sensor can be dominated by at most one category. Then, an object is deemed as in a category if the majority of sensors near the object belong to that category. However, the original DLNG is a kind of coarse-grained model in which a determined category may contain distinct objects. For example, intelligent agents such as humans are generally capable of assessing that crimson is much redder than magenta, although both colors belong to the same category of red.

To overcome this difficulty, here we propose a likelihood category game model (LCGM), in which self-organized agents define category based on acquired knowledge and use likelihood estimation to distinguish objects in the same category. The information communicated among the agents is no longer simply some absolute answer but involves agents' perception. That is, the agents are able to classify objects in terms of distinct likelihoods as determined by their knowledge, which can be further updated through learning new objects. For the special case where the learning domain is highly localized, our model reduces to a variant of the minimum NG. The primary contribution of this paper is the proposed LCGM, with a new concept of likelihood to allow agents assess an object via perception, which better reflects our real experiences. Some conclusive remarks related to consensus characteristics are provided. It is noticed that short distance and heterogeneity can facilitate the consensus. Moreover, LCGM clearly reflects more complex knowledge makes consensus harder to be reached, while agents with larger degree are more "intelligent", contributing more to the knowledge formation and consensus. Our work in LCGM provides novel insights into the process of consensus, with potential applications to predicting and controlling consensus, related to problems with implications to social, economical, and even political systems where achieving consensus is often a desired objective.

2. Likelihood category game

In this paper, a new game model, called likelihood category game model (LCGM), is proposed to illustrate how agents categorize a class of objects O and interact with each other. Here, an object $o_i \in O$ is assumed to be characterized by a point in an N-dimensional feature domain [23], that is $x_i \equiv \left(x_{i,1}, x_{i,2}, \ldots, x_{i,N}\right) \in D \subseteq \Re^N$.

2.1. Agent model

An agent, Agent_j, has memory which maintains a set of categories $\mathcal{C}_j$. Each category, $c_k \in \mathcal{C}_j$, is described by a weight ω_k and N's normal distributions, $\mathcal{N}(\mu_{k,n}, \sigma_{k,n}^2)$, each corresponds to the nth dimension feature with $n = 1, 2, \ldots, N$.

Agent_j can then classify an object o_i based on a likelihood score which reflects how likely o_i belongs to category c_k. The score is defined as

$$LS_j(o_i, c_k) = g(\omega_k) \prod_{n=1}^{N} \left[f\left(\frac{x_{i,n} - \mu_{k,n}}{\sigma_{k,n}} \right) / f(0) \right] \tag{1}$$

where $f(\cdot)$ is the probability density function (PDF) of the standardized normal distribution and $g(\cdot)$ is a scale function specified by

$$g(\omega) = \tanh(\alpha\omega), \tag{2}$$

with a constant α. Here, α is a parameter adjustable to the difficulty of the knowledge.

As shown in Eq. (1), the likelihood score consists of two components, namely the prior and the likelihood components. The prior component, represented by $g(\omega_k)$, indicates the probability that category c_k is selected for an arbitrary object. In our simulations, we set $\alpha = 2.5$, and $\omega = 1$ when a new category is created. Consequently, based on (2), we have $g(1) = \tanh(2.5) = 0.987 \approx 1.0$. It reflects the case that agent can learn a category very well via only one game. In addition, further learning of the same object will not improve the category much, as tanh is a saturation function.

The likelihood component is specified by the product term in Eq. (1). The normalized value $f\left(\frac{x_{i,n}-\mu_{k,n}}{\sigma_{k,n}}\right)/f(0)$ measures the likelihood of o_i belonging to category c_k along the nth dimension. There are two considerations to apply normalization. Firstly, in the LCGM, $\sigma_{k,n}$ is non-decreasing and is likely increased after incorporating new knowledge from an encountered object. Therefore, the distribution $\mathcal{N}(\mu_{k,n}, \sigma_{k,n}^2)$ would be dispersed, making $f(\cdot)$ in Eq. (1) small. This eventually affects the LS score, violating the general understanding of learning. To resolve this problem, normalization is thus adopted and thus it is divided by $f(0)$ that is the maximum value of $f\left(\frac{x_{i,n}-\mu_{k,n}}{\sigma_{k,n}}\right)$. Secondly, based on Eq. (1), the LS score is always bounded from 0 to 1, making the comparison of score and the determination of threshold LST easier (see Fig. 1). Also, if the nth feature value of an object equals to $\mu_{k,n}$ for a category, it should be assigned the highest score, which is one, as specified by Eq. (1).

It is also remarked that, in the absence of any correlation among different dimensions, the product term gives the joint probability value.

In LCGM, each agent can execute four possible activities: object identification, category updating, category creation, and category deletion, which are described in the followings.

2.1.1. Object identification activity (OIA)

Given an object o_i, Agent_j will return a null name ($\text{name}_a = \text{null}$) and a zero likelihood score $[LS_j(o_i, c_a) = 0]$ if its memory is empty, i.e. $\mathcal{C}_j = \emptyset$. Otherwise, it returns $LS_j(o_i, c_a)$ and the corresponding name_a, where $c_a = \arg\max_{c_x \in \mathcal{C}_j} LS_j(o_i, c_x)$ is the category having the highest score, providing that $LS_j(o_i, c_a)$ is larger than or equal to a specified threshold LST. In case, $LS_j(o_i, c_a) < LST$, implying that the agent has very weak confidence to categorize the object, the agent will again return a null name and a zero likelihood score.

An illustrative example of OIA for an one-dimensional case is given in Fig. 1. Suppose the agent has two categories: "Blue" and "Red". With regard to objects "a", "b", "c" and "d", there are four cases (a, b, c and d, respectively). In case a, since $LS(a, \text{Blue}) > LS(a, \text{Red})$ and $LS(a, \text{Blue}) > LST$, the agent returns the name of category "Blue" and $LS(a, \text{Blue})$. In case b, $LS(b, \text{Red}) > LS(b, \text{Blue})$ and $LS(b, \text{Red}) > LST$, the name of category "Red" and $LS(b, \text{Red})$ are returned. In case c, $LS(c, \text{Blue})$ is higher but still lower than LST, so the agent returns name $=$ null and $LS = 0$. In case d, $LS(d, \text{Red}) = LS(d, \text{Blue}) > LST$. The category "Red" or "Blue" is randomly picked. However, it should be remarked that this case is rare as it occurs only at the intersection point of the two curves.

2.1.2. Category updating activity (CUA)

If an object o_i is assigned to category c_a, the features ($\mu_{a,1}$, $\sigma_{a,1}, \ldots, \mu_{a,N}$, $\sigma_{a,N}$ and ω_a) of c_a will be updated by including the information of o_i. Since o_i is solely represented by the point x_i in the domain D, by using the same feature representation, o_i can be presented by having $\omega_i = 1$, $\mu_{i,k} = x_{i,k}$ and $\sigma_{i,k} = 0$ with $k = 1, 2, \ldots, N$.

The purpose of CUA is then to merge the features of the new object o_i to the category c_a, forming an updated category c_a^*. Therefore, the new weight of c_a^* can be computed by

$$\omega_a^* = \omega_a + \omega_i = \omega_a + 1 \tag{3}$$

The process in forming an updated category can then be illustrated by the use of weighted distribution curves, as shown in Fig. 2 (for simplicity, only one dimension is shown). The areas under the curves for original category, the object, and the updated category are ω_a, ω_i and ω_a^*, respectively.

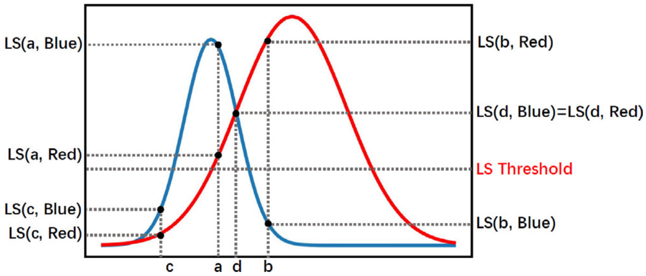

Fig. 1. An illustrative example of object identification based on one dimension cases.

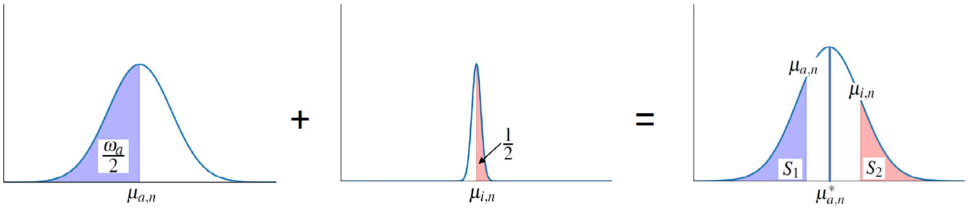

Fig. 2. An illustrative example of σ updating.

For each nth dimension, the updated mean, $\mu^*_{a,n}$, is the weighted average of the category c_a and the point x_i, which can be obtained through

$$\mu^*_{a,n} = \frac{\omega_a \times \mu_{a,n} + \omega_i \times \mu_{i,n}}{\omega^*_a} = \frac{\omega_a \times \mu_{a,n} + x_{i,n}}{\omega^*_a} \tag{4}$$

We further assume an exception condition for those in original category c_a under the updated category c^*_a, which is given by

$$Pr\left(x_{j,n}\left|\left[|x_{j,n} - \mu^*_{a,n}| > |\mu_{a,n} - \mu^*_{a,n}|\right]\right.\right) < \frac{\omega_a}{\omega^*_a} \tag{5}$$

where $Pr(\cdot\,|\cdot\,)$ stands for the conditional probability. Similarly, the following condition can be derived in related to the object o_i:

$$Pr\left(x_{j,n}\left|\left[|x_{j,n} - \mu^*_{a,n}| > |\mu_{i,n} - \mu^*_{a,n}|\right]\right.\right) < \frac{\omega_i}{\omega^*_a} \tag{6}$$

The above definitions specify the exception without the need for additional parameters. Moreover, it differentiates a newly encountered object ($\omega_i = 1$) and a category with ω_a times of updates.

Since an object, which is not an exception in the original category, should not be considered as an exception in the updated category, based on (6), one requires

$$Pr\left(x_{i,n}\left|\left[|x_{i,n} - \mu^*_{a,n}| > |x_{i,n} - \mu^*_{a,n}|\right]\right.\right) \geq \frac{1}{\omega^*_a} \tag{7}$$

which eventually has

$$\sigma^*_{a,n} \geq \frac{-|x_{i,n} - \mu^*_{a.n}|}{F^{-1}\left(\frac{1}{2\omega^*_a}\right)} \tag{8}$$

with $F(.)$ denoting the CDF of $f(\cdot)$. Similarly, by considering (5), one also has

$$\sigma^*_{a,n} \geq \frac{-|\mu_{a,n} - \mu^*_{a.n}|}{F^{-1}\left(\frac{\omega_a}{2\omega^*_a}\right)} \tag{9}$$

Therefore, the updated σ^*_a can be obtained by:

$$\sigma^*_{a,n} = \max\left(\frac{-|x_{i,n} - \mu^*_{a,n}|}{F^{-1}\left(\frac{1}{2\omega^*_a}\right)}, \frac{-|\mu_{a,n} - \mu^*_{a,n}|}{F^{-1}\left(\frac{\omega_a}{2\omega^*_a}\right)}, \sigma_{a,n}\right), \tag{10}$$

The conditions (8) and (9) stipulate that the area $S_1 \geq \frac{\omega_a}{2}$ and $S_2 \geq \frac{1}{2}$, as shown in Fig. 2. For example, we let $\omega_a = 7$ and $\sigma_a = 1$ for c_a, and $\omega_i = 1$ for the object. Eq. (3) thus gives $\omega_a^* = \omega_a + \omega_i = 8$. For the nth dimension, the shaded areas of c_a and x_i are $\omega_a/2 = 3.5$ and $\omega_i/2 = 0.5$, respectively. Consider $S_1 = \omega_a^* \times Pr(x_{j,n} \leq \mu_{a,n})$ and $S_2 = \omega_a^* \times Pr(x_{j,n} \geq \mu_{i,n})$ in the updated category. Now, if $\mu_{a,n} = 0$ and $\mu_{i,n} = 0.8$, then Eq. (4) gives $\mu_{a,n}^* = 0.1$ and $\sigma_{a,n}^*$ is maintained as $\sigma_{a,n}$ $[\sigma_{a,n}^* = \max(0.456, 0.636, 1) = 1]$. Correspondingly, one has $S_1 = 3.68 > \omega_a/2$ and $S_2 = 1.94 > 1/2$. For another example, if $\mu_{a,n} = 0$ and $\mu_{i,n} = 2$, then one has $\mu_{a,n}^* = 0.25$ and $\sigma_{a,n}^* = \max(1.141, 1.589, 1) = 1.589$. As a consequence, $S_1 = 3.5 = \omega_a/2$ and $S_2 = 1.083 > 1/2$.

Note that object x_i (middle) should be a Dirac delta function ($\sigma_{i,n} \to 0$) at $\mu_{i,n}$, and it is plotted as a bell-shape only for visual clarity.

2.1.3. Category creation activity (CCA)

If an object o_i with name$_i$ concluded in a game cannot be related to any existing category associated with the agent, a new category c_a is created. The parameters σ_a and ω_a are set at some default values ($\sigma_{default}$ and $\omega_{default}$, respectively) while $\mu_a = x_i$ and name$_a$ = name$_i$.

2.1.4. Category deletion activity (CDA)

In reality, if knowledge is not recalled, it will be forgotten gradually. Similarly, in LCGM, category will be removed from one's memory if there is no updating, i.e., it has not been learned for a long time. The removal of a category c_k is based on its weight ω_k. For any time t, ω_k is updated as

$$\omega_{k,t} = \omega_{k,t-1} \times e^{-\phi} \tag{11}$$

where ϕ is the "forgetting" factor that describes the "forgetting speed" of a category in the agent's memory, $\omega_{k,t}$ and $\omega_{k,t-1}$ are the value of ω_k at time t and $t-1$, respectively.

As Eq. (1) indicates, a reduction in ω_k will also affect the chance of category c_k being assigned in OIA. If $\omega_{k,t}$ further reduces and becomes smaller than a pre-defined threshold (denoted as ω_{Th}), the category c_k will be removed from the memory of the agent.

To eliminate the influence of the removal on learning process, ω_{Th} should be very small. Then, only categories with nearly no probability to be recalled by CUA will be removed. Theoretically, removing (or not removing) such an insignificant category will not affect the categorization results, given proper choices of ω_{Th} and LST (such as our settings). A category with $\omega_{k,t} < \omega_{Th}$ will never be recalled again when $\tanh(\alpha\omega_{Th}) < LST$ holds. However, keeping many of these insignificant categories throughout the learning process would lead to a waste of computing resources, and hence category deletion in CDA is practically essential. On the other hand, once an object falls into category c_k, ω_k becomes larger than one [Eq. (3)] via CUA.

It is remarked that such deletion process of a category c_k is solely governed by the condition $\omega_{k,t} < \omega_{Th}$, while other categories would not be affected (i.e. their weights and corresponding N's distributions remain unchanged).

2.2. Rules of a game

Game participants are all agents in a population on a network. Initially, every agent has empty memory: $\mathcal{C}_j = \emptyset,\ \forall j$. In each iteration, a pair of connected agents, Agent$_A$ and Agent$_B$, are randomly selected in a game that proceeds as follows.

(1) An arbitrary object o_i is presented to both agents. Via OIA, Agent$_A$ returns name$_a$ and $LS(o_i, c_a)$, while Agent$_B$ returns name$_b$ and $LS(o_i, c_b)$.

(2) Agent$_A$ and Agent$_B$ conclude the name of o_i, referred to as name$_i$, based on the following rules: if $LS(o_i, c_a) > LS(o_i, c_b)$, name$_i$ = name$_a$ and Agent$_A$ wins the game; if $LS(o_i, c_a) < LS(o_i, c_b)$, name$_i$ = name$_b$ and Agent$_B$ wins the game; if $LS(o_i, c_a) = LS(o_i, c_b) = 0$, name$_i$ is randomly generated and the game is a draw; if $LS(o_i, c_a) = LS(o_i, c_b) \neq 0$, a winner is randomly selected. For example, if Agent$_A$ is the winner, one has name$_i$ = name$_a$. In addition, if name$_a$ = name$_b$ $\neq$ null, the game is successful and name$_i$ = name$_a$. Otherwise, it fails.

(3) Agent$_A$ and Agent$_B$ update/create their categories according to a set of rules. The rules for Agent$_A$ can be described as follows (similar for Agent$_B$). If name$_i$ equals the name of a category, say $c_k \in \mathcal{C}_A$, CUA is operated on category c_k. However, if name$_i$ is not contained in any category in $\mathcal{C}_A$, CCA is carried out to create a new category with name$_i$ and $\mu_{i,n} = x_{i,n}$ for $n = 1, 2, \ldots, N$.

(4) All agents in the population delete expired categories via CDA.

Steps (1) to (4) are repeated until a predefined number of iterations is reached.

To better illustrate the game, two examples are given in Fig. 3. In Game I, via OIA, Agent$_A$ considers that the object is "Blue" with 50% certainty (according to the LS score = 0.5), while Agent$_B$ regards it as "Green" with 70% certainty. Since Agent$_B$ is more confident, the object is concluded as "Green". The two agents then update their memories based on the new knowledge that the object is named "Green". In addition, both agents forget categories that have not been recalled for a long period of time.

In Game II, neither agent can identify the object and both agents return name = null and $LS = 0$ via OIA. As a result, a new name, say "Green", is assigned. The remaining steps, Steps (3) and (4), are the same as those in Game I.

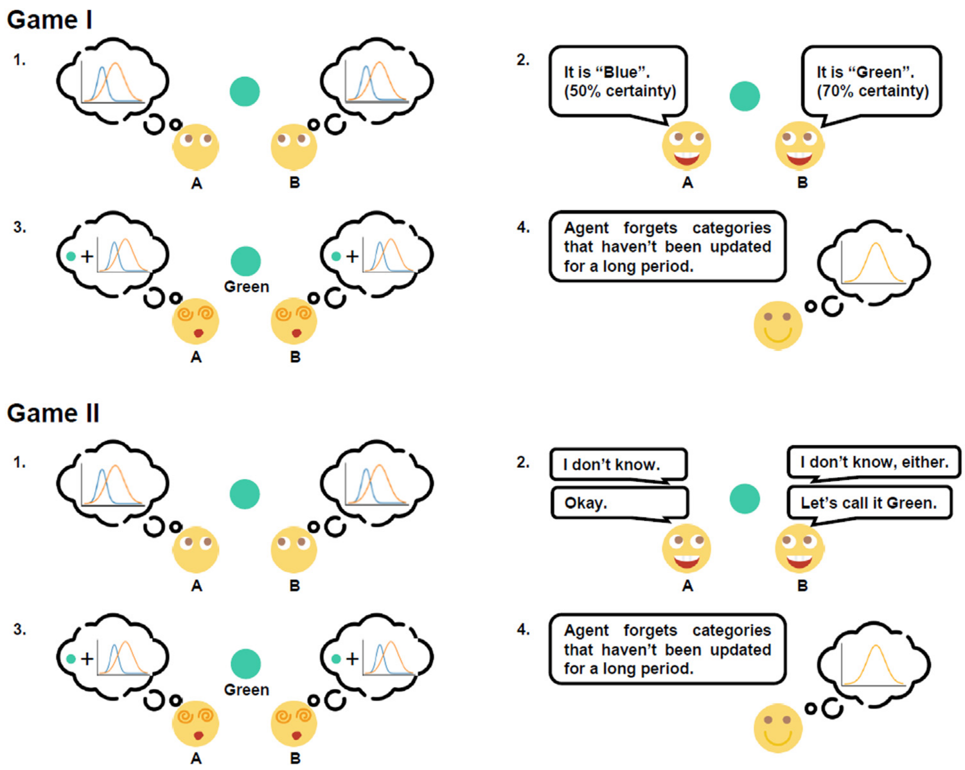

Fig. 3. Two illustrative examples of game process in terms of color objects.

3. Simulation results

To demonstrate and analyze LCGM, we consider the color categorization as an example, and conduct LCGM onto a set of agents. Color objects are randomly generated in *RGB* form with sample size $256 \times 256 \times 256$, and are then mapped into *CIELab* color space for agents. Specifically, the color objects and categories are represented in the *CIELab* color space (L*, a*, b*), where L* stands for the lightness, a* and b* are the two abstract opponent dimensions [29]. The *CIELab* color space is assumed to be homogeneous, i.e., the difference between two color objects, ΔE, is defined as the Euclidean distance between them, as specified by the year 1976 version of *CIELab* color space. (Remark: To better fit with human vision, ΔE has been re-defined in the year 2000 version [30] where the color space is no longer homogeneous. However, this does not affect our main results.)

3.1. Simulation settings

LCGM is applied to a population of agents connected through a pre-defined network, either from the real world or from a specific model. Games are conducted as described in Section 2.2, and repeated until a predefined number of iterations is reached, while for all the simulations in this paper, it is set as 10^7.

The main model's parameters are given as follows.

1. Scaling factor in Eq. (2): $\alpha = 2.5$
2. Default value of σ in CCA: Empirically, two colors with the value of ΔE between 1 and 10 appear similar in human vision, so we set $\sigma_{default} = 5.0$.
3. Default value of ω in CCA: $\omega_{default} = 1.0$.
4. The threshold parameter ω_{Th}: A category will be deleted in CDA if its weight ω is smaller than ω_{Th}. We set $\omega_{Th} = 0.01$.
5. Likelihood score threshold (LST): If the LS computed from an object-category pair is smaller than LST, the object should not belong to that category. We choose different values for LST to investigate its impact on the learning process.
6. Forgetting factor ϕ: It characterizes the speed at which a category is forgotten. We choose different values of ϕ to investigate its impact on the learning process.

3.2. Performance metrics

To better present the results, we consider the following performance metrics.

Accuracy (ACC) indicates the success rate of the games. ACC is computed for every K games and defined as:

$$ACC = \frac{k_{success}}{K} \tag{12}$$

where $k_{success}$ is the number of successful games in every K iterations and $K = 10{,}000$ in our simulations.

Number of categories (NC) specifies the number of categories maintained by each agent, which reflects the resolving power of the agent after learning. The average number of categories (NC_{Avg}) over the whole population is primarily concerned in this paper.

Total number of distinct names (TDN) records how many distinct names remained in the whole population.

Consensus score (CS) characterizes the consensus of the whole population. A set of sampled objects ($\mathcal{O}$) is firstly selected. In this work, $\mathcal{O}$ consists of 512 objects uniformly sampled from the domain D. These objects are presented to every pair of agents in turns (even they are not neighbors) and games are performed. For Agent$_i$ and Agent$_j$ with $i \neq j$, $CS_{i,j}$ is defined as

$$CS_{i,j} = \frac{agm_{i,j}}{|\mathcal{O}|} \tag{13}$$

where $agm_{i,j}$ is the total number of successful games that Agent$_i$ and Agent$_j$ made for all $o_i \in \mathcal{O}$, and $|\cdot|$ is the cardinality of a set. The CS of the entire population $\mathcal{V}$ is the average over all possible agent pairs, which is given by

$$CS = \frac{\sum_i \sum_{j \neq i} CS_{i,j}}{|\mathcal{V}| \times (|\mathcal{V}| - 1)} \tag{14}$$

3.3. Consensus in social networks

We apply the LCGM algorithm to two social networks, a subgraph of the Facebook social network [31] and a subgraph of an E-mail network [32], collected by the SNAP laboratory of Stanford university. The Facebook subgraph has 1,034 nodes and 26,749 edges. Its clustering coefficient, average path length and degree-based Gini coefficient are 0.5, 2.9 and 0.48, respectively. The E-mail network has 986 nodes and 25,552 edges. Its clustering coefficient, average path length and degree-based Gini coefficient are 0.31, 2.6 and 0.56, respectively. The relatively large clustering coefficient and short average path length are typical of small-world networks, while the relatively high Gini coefficients suggest that the networks are inhomogeneous.

Typical simulations with $LST = 0.1$ and $\phi = 0.00002$ have been conducted, and results are averaged by 10 runs (Note: Same for the other results). On Facebook social network, ACC reaches 88.0%, the average NC and TDN are about 4.7 and 6.8, respectively. And on E-mail network, ACC reaches 88.6%, the average NC and TDN are 4.6 and 5.8, respectively.

However, the high accuracy does not guarantee full consistencies of agents under the framework of category game. Since accuracy only considers gaming actions occur between two neighbors, consensus between agents who are not neighbors in the population is ignored. The consensus score (CS) is then used to characterize the consistency of the whole population. The average CS for Facebook social network and E-mail network are about 0.71 and 0.85, respectively.

The performance of CS is further investigated by plotting the histograms of consensus score as given in Fig. 4 (a) and (b). The majority of agent pairs have CS higher than 0.8 for both networks. However, some agent pairs have relatively low CS, particular for the Facebook social network. As noticed in Fig. 4 (c) and (d), CS and distance of agent pairs are negative correlated. It is because agents involved in the transmission path would incorporate their understandings to the information. As a result, a longer distance between two agents generally imposes a higher variation in the knowledge between them.

The adoption of LCGM on the two social networks reveals that agents basically reach the consensus through mutual learning, with consensus score 0.71 (the Facebook network) and 0.85 (the E-mail network) respectively. While the CS of the E-mail network is much higher than that of the Facebook network even though they have similar ACC. Extended analyses on CS and distance reveal that the shorter average path length of the E-mail networks may facilitate the consensus, supported by the above-mentioned relationship between CS and distance. Other topology properties, such as clustering and heterogeneity, will be discussed later in this section.

Based on LCGM, an interesting question is, among the agents, who more frequently takes the lead in the game by providing the answer? As specified by the game rules, the leader wins the game. In order to identify the leader, we define the following game score for each agent:

$$GS(A) = \sum_i GS_i(A) \tag{15}$$

where $GS_i(A)$ is the game score obtained by Agent$_A$ in ith game:

$$GS_i(A) = \begin{cases} 1, & \text{Agent}_A \text{ joined } i\text{th game and won} \\ 0.5, & \text{Agent}_A \text{ joined } i\text{th game and the result was a draw} \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

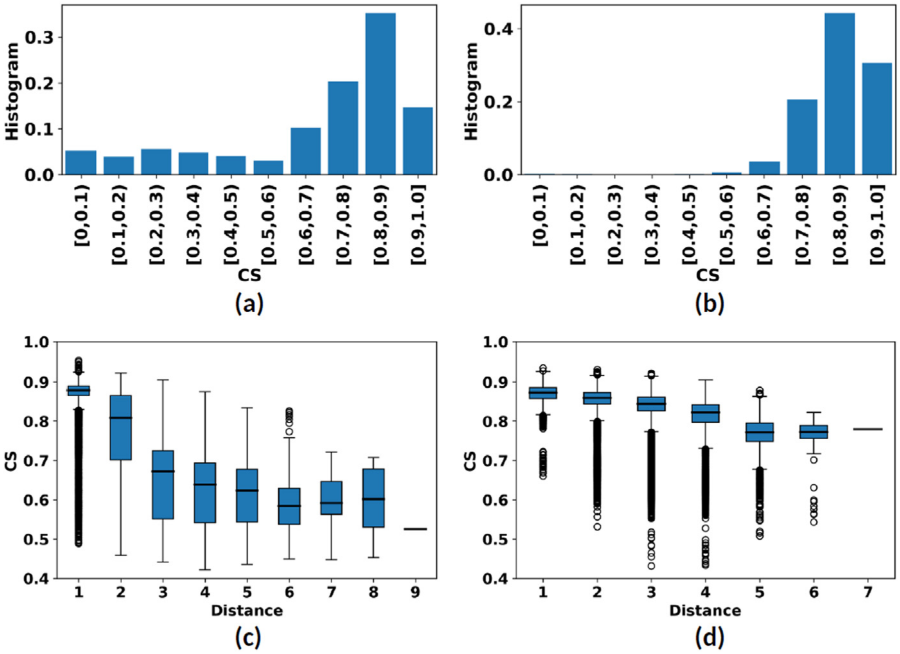

Fig. 4. Consensus score (CS) of agent pairs. Shown are (a) the histogram of the consensus score of Facebook social network, (b) the histogram of the consensus score of E-mail network, (c) boxplot of the correlation between distance and CS of Facebook social network and (d) boxplot of the correlation between distance and CS of E-mail network.

Eq. (16) indicates that our proposed learning game is effectively a positive-sum game. In each game, the increment of the total score is one. When an agent wins a game, it dominates the game solely and takes the whole score. The loser receives no penalty for the reason that learning should not be discouraged. When the game is a draw, both players score 0.5 as they contribute to the knowledge formation equally. To eliminate the effect of opportunity earning (i.e. agents involved in more games likely obtain higher scores), we further define the following game scoring rate:

$$GSR(A) = \frac{GS(A)}{\#\text{Games}(A)} \tag{17}$$

where $\#Games(A)$ is the total number of games involving Agent_A.

Since results are similar for both studied social networks, we only discuss the results of the Facebook social network here. Fig. 5 (a) shows the GSR distribution, giving a bell-shape in which the majority (93.5%) of agents have GSR values between 0.4 and 0.6. To check whether the distribution is normal, we use the Quantile–Quantile plot (QQ plot). Specifically, a dot (x, y) in the QQ plot means $F_X(x) = F_Y(y)$, where $F_X(x)$ is the cumulative density function of random variable X. If two distributions are identical, all dots would be located along the diagonal. As shown in Fig. 5 (b), the GSR possesses the characteristic of a normal distribution within $x \in (-2, 2)$. Such a bell-shape distribution reflects most agents have moderate intelligence after sufficient learning and communications. Furthermore, the emergence of a light right tail and a heavy left tail observed in Fig. 5 indicates that “geniuses are minority” and “dummies are more than you expected”.

The correlation between GS/GSR and degree is shown in Fig. 6. As defined in Eqs. (15) and (16), GS reflects the frequency of transmitting self-knowledge to others. Fig. 6 (a) shows that agents with large degrees are likely to have high GS, implying that agents with larger degree contributed more to the consensus process, which is also observed in other agent interacting model, such as for social synchrony [33]. Different from GS, GSR measures the capability of scoring in a game [Eq. (17)]. Fig. 6 (b) reveals a positive correlation between GSR and degree, indicating that agents with larger degrees are more “intelligent” with respect to eventual knowledge formation through more active learning and communication.

3.4. Effects of LST and FF

Then, the effects of likelihood score threshold (LST) and forgetting factors (FF) of categories are investigated. Again, the studies are based on the Facebook social network.

Likelihood score threshold of agents. As shown in Fig. 1, LST imposes the minimum requirement for classifying an object into a category. Accordingly, a high LST value encourages the generation of a new category. To investigate the effect of LST, we simulate the learning process for different LST with fixed forgetting factor $\phi = 0.00002$ in Eq. (11).

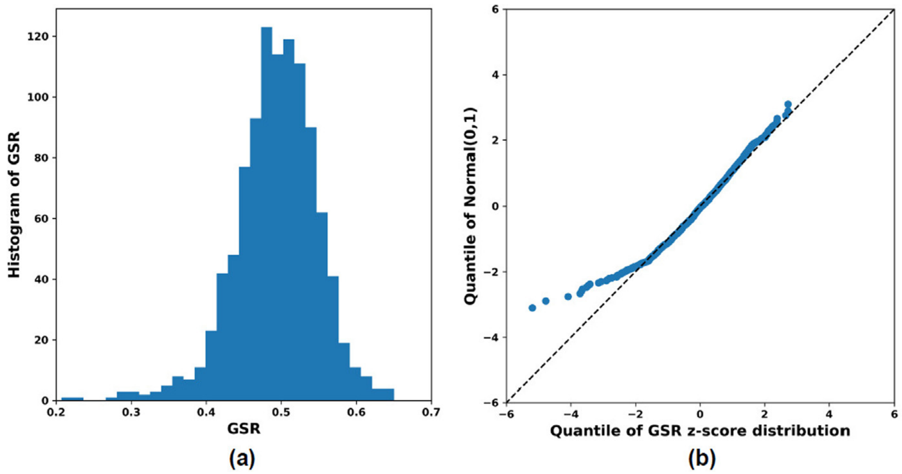

Fig. 5. Statistical features of game scoring rate for the Facebook social network. Shown are (a) the histogram of the game scoring rate (GSR) and (b) the corresponding Quantile–Quantile plot (QQ plot) for $\mu_{GSR} = 0.49$ and $\sigma_{GSR} = 0.06$.

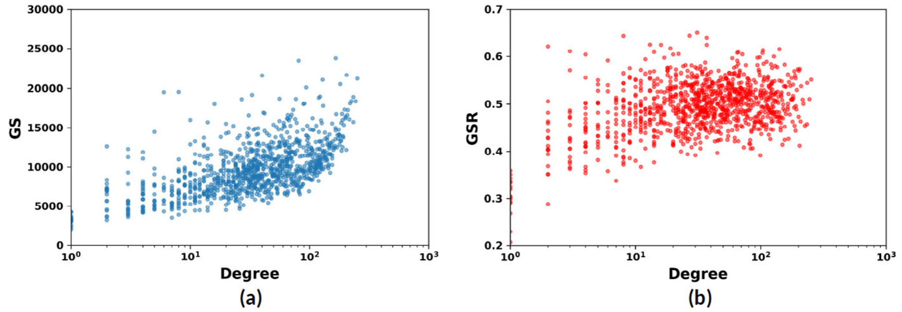

Fig. 6. Correlation between GS/GSR and degree centrality for the Facebook social network. The Pearson correlation coefficients are $\rho_{log(d),GS} = 0.60$ and $\rho_{log(d),GSR} = 0.40$.

Table 1
Summary of LGCM results (after 10^7 iterations) for different values of LST/FF.

LST ($\phi = 0.00002$)				FF ($LST = 0.2$)			
LST	ACC	Avg.NC	TDN	ϕ	ACC	Avg.NC	TDN
0.1	88.0%	4.7	6.8	0.00001	81.3%	10.2	15.4
0.2	85.5%	5.5	7.7	0.00002	85.5%	5.5	6.7
0.3	84.1%	7.5	10.0	0.00004	89.3%	3.6	4.3
0.4	82.3%	9.3	42.3	0.00008	93.8%	2.2	2.2
0.5	79.4%	13.6	412	0.00016	98.0%	1.3	1.3
0.6	71.9%	27.6	6472	0.00032	99.9%	1.0	1.0
0.7	38.6%	154.6	67670	0.00064	79.8%	3.55	1399
0.8	4.2%	394.3	174366	0.00128	0%	6.92	3549

As shown in Table 1 (left), agents with high LST (the "rigorous" agents) spontaneously possess more categories. The more complex knowledge also makes it harder to reach consensus, resulting in a lower accuracy. This is further confirmed by the plot of consensus scores with different LST as shown in Fig. 7 (a), clearly showing that the increment of LST would depreciate CS significantly.

Forgetting factor of categories. We next study the mechanism responsible for categories to be gradually forgotten associated with category deletion activities. In general, the threshold for removal of a category, ω_{Th}, should be small because, if ω_{Th} is large, even useful categories would be deleted, possibly resulting in a dramatic effect on the learning process. However, if the value of ω_{Th} is too small, the category with small ω results small likelihood score, leading to a small probability of any update (learning). These empirical considerations lead to our choice of $\omega_{Th} = 0.01$.

To be concrete, we fix LST = 0.2 and focus on the impact of varying the forgetting factor ϕ. The results are shown in Table 1 (right) and Fig. 7 (b). When ϕ increases, it becomes hard for agents to remember information, eventually leading to a reduction in the number of categories. Apparently, when there are fewer categories, it is easier to reach consensus,

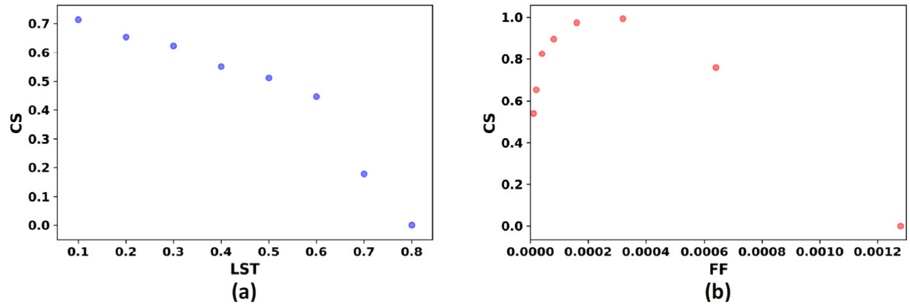

Fig. 7. CS of simulations with different LST/FF.

Table 2
Summary of network characteristics and LCGM results for BA scale-free networks of 1000 agents with different average degree (k_{avg}). The notation GC strands for Gini coefficient to degree, CC stands for clustering coefficient, and APL stands for average path length.

	Network Characteristics			LCGM results		
k_{avg}	GC	CC	APL	ACC	Avg. NC	CS
6	0.38	0.02	3.44	86.6%	5.85	0.83
8	0.37	0.03	3.13	87.6%	5.30	0.85
10	0.36	0.04	2.94	85.4%	6.89	0.84
12	0.36	0.04	2.81	85.8%	6.86	0.83
14	0.35	0.05	2.72	86.5%	6.37	0.85
16	0.35	0.06	2.64	86.2%	5.99	0.85

Table 3
Summary of network characteristics and LCGM results for BA scale-free networks with different number of agents (N) and fixed average degree of 10. The time scale is normalized by tuning λ and the number of iterations.

	Network Characteristics			LCGM results		
N	GC	CC	APL	ACC	Avg. NC	CS
400	0.36	0.07	2.68	90.9%	4.03	0.90
500	0.36	0.06	2.73	89.6%	6.25	0.88
600	0.36	0.06	2.78	88.4%	5.48	0.87
700	0.36	0.05	2.83	89.2%	5.79	0.87
800	0.36	0.05	2.87	86.9%	6.81	0.86
900	0.36	0.04	2.91	86.3%	6.69	0.84
1000	0.36	0.04	2.94	85.4%	6.89	0.84

which is verified through the ACC and CS results. However, the correlation between the FF and CS is non-monotonous. For $\phi > 0.00032$, the CS decreases dramatically and even equals to 0 for $\phi = 0.00128$ through the process. That is, quite naturally, consensus can never be reached if agents forget things too fast.

It is remarked that the average result of CS is presented. For the case with $\phi = 0.00064$, some experiments reached consensus with only one category (similar to the case with $\phi = 0.00032$) while others had $CS = 0$ (similar to the case with $\phi = 0.00128$).

3.5. Effect of network topology

As revealed by many studies [4–8,34], the network topology and social activities are strongly correlated. Therefore, it is natural to ask whether network topology also has a significant effect on LCGM. To address this question, we consider four topological characteristics, namely the average degree, network size, heterogeneity, and clustering.

Based on original BA scale-free networks [35], the impacts of average degree (with fixed network size) and network size (with fixed average degree) onto the performance of LCGM are investigated. For each setting, ten network realizations are used to calculate the various statistical averages. The simulation results show no observable influence on LCGM by varying the average degree (see Table 2). While the network size affects the consensus of LCGM slightly (see Table 3), network with smaller size can reach a better consensus, which meets most people's intuition that consensus is easier for fewer agents.

To investigate the effects of heterogeneity and clustering, two types of heterogeneous networks are adopted: scale-free network with aging [36] and scale-free network with clustering attachment [37]. The first type of networks incorporates

Table 4
Summary of network characteristics and LCGM results for scale-free networks incorporating aging.

	Network Characteristics			LCGM results		
λ	GC	CC	APL	ACC	Avg. NC	CS
0	0.36	0.04	2.94	86.7%	6.70	0.84
0.005	0.25	0.04	3.28	85.6%	6.54	0.84
0.01	0.21	0.05	3.46	85.7%	6.08	0.80
0.015	0.20	0.05	3.71	87.4%	5.61	0.75
0.02	0.19	0.05	4.05	88.8%	6.18	0.73
0.025	0.18	0.06	4.41	85.2%	6.34	0.64
0.03	0.18	0.07	4.79	85.5%	5.92	0.58

Table 5
Average consensus score (CS) of agent pairs with different distance on scale-free networks incorporating aging.

	distance								
λ	1	2	3	4	5	6	7	8	9
0	0.84	0.84	0.84	0.84	0.84	\	\	\	\
0.005	0.85	0.84	0.84	0.83	0.83	\	\	\	\
0.01	0.85	0.83	0.82	0.82	0.78	0.74	0.72	\	\
0.015	0.87	0.85	0.84	0.74	0.72	0.69	0.69	0.65	\
0.02	0.87	0.85	0.82	0.76	0.69	0.63	0.60	0.58	\
0.025	0.85	0.83	0.79	0.68	0.56	0.47	0.44	0.43	\
0.03	0.85	0.82	0.77	0.65	0.54	0.46	0.37	0.28	0.25

Table 6
Summary of network characteristics and LCGM for scale-free networks with different clustering coefficients ($LST = 0.2$ **and** $\phi = 0.00002$).

	Network Characteristics			LCGM results		
p	GC	CC	APL	ACC	Avg. NC	CS
0	0.36	0.04	2.94	86.7%	6.70	0.84
0.25	0.36	0.06	2.98	85.9%	7.02	0.83
0.5	0.36	0.09	3.02	86.5%	6.53	0.84
0.75	0.35	0.11	3.07	87.7%	5.70	0.85
1	0.35	0.15	3.19	86.9%	6.40	0.83

the aging effect of nodal attraction, and the connection probability is redefined as

$$\mathrm{Prob}(i) = \frac{\mathrm{degree}_i \times e^{-\lambda\times \mathrm{age}_i}}{\sum_j \mathrm{degree}_j \times e^{-\lambda\times \mathrm{age}_j}} \tag{18}$$

where λ is a tunable parameter reflecting the aging speed of nodal attraction and age_i denotes the age of node i. When node i is newly added, $\mathrm{age}_i = 0$. After every cycle of node addition, age_i is incremented by one. Thus, networks with distinct degrees of homogeneities can be generated by adjusting the value of λ.

We perform LCGM on the aging scale-free networks of size 1000 with average degree of 10 for different values of λ. As λ is increased, the Gini coefficient is significantly reduced and the average path length increases, but the clustering coefficient is hardly affected. We find that an increase in the value of λ has little effect on the accuracy (ACC) and the average number of categories (NC_{avg}) in LCGM. However, the consensus score (CS) decreases significantly with λ even though the accuracy remains high (see Table 4). As discussed before, there is a negative correlation between the agent-pair distance and CS. Therefore, We further investigate the average consensus scores of agent pairs with respect to distance and λ (see Table 5). It is observed that λ has no observable influence on CS between agents with short distance (≤ 2), while for agent pairs with longer distance, CS decreases generally as the network becomes more homogeneous (λ increases).

Then, we investigate the effect of clustering on LCGM. Scale-free networks with different clustering coefficients are obtained through the process of clustering attachment [37] where an edge is added to connect a new node and one of its two-hop neighbors, forming a triangle. The probabilities of clustering attachment and preferential attachment are p and $1 - p$, respectively. If no two-hop neighbor is available for a new node, preferential attachment is applied. LCGM is performed on networks of size 1000 (with average degree of 10) for different values of p (See results in Table 6). It can be easily observed that the clustering coefficient increases with the value of p. An increase in the value of p, however, has only limited effect on the Gini Coefficient and the average path length. From these results, it can be concluded that the clustering coefficient does not hurt the consensus. Since categories have weights in LCGM, several strong categories will survive through evolution. Different from traditional naming game models, LCGM allows these categories to coexist, and consequently, they will dominate the whole population together.

4. Discussions and conclusion

The basic idea underlying our proposed likelihood category game model (LCGM) is that knowledge acquiring is essential to achieving consensus. In LCGM, self-organized agents rely on acquired knowledge to define category and employ statistical likelihood estimation to distinguish and "name" objects that belong to the same category. Particularly, the agents equipped with knowledge acquiring are capable of exploiting distinct likelihoods as determined by their knowledge to classify objects. Importantly, knowledge is not static but dynamic: agents update knowledge through learning. The agents in our LCGM are thus "smart", more closely mimic those in the real social, economical, and political world.

The distinct features of our proposed LCGM are the followings. Firstly, it is a truly autonomous category game model, eliminating the need for ground truth knowledge. Secondly, introducing the concept of likelihood in this context makes the game model more realistic because, in each game, agents not only return the names but also the likelihood scores that represent their confidence corresponding to the answer. The information communicated among agents is "smart" in the sense that it is no longer restricted to some form of absolute answer but is more content- or feature-based. Thirdly, in a pairwise game, the more knowledgeable agent (with a higher likelihood score) becomes the "teacher", a feature that fits with the interaction in the real world. Through the incorporation of learning, our model generalizes the existing naming game models and is closer to describing reality. (Our model reduces to a variant of the much studied minimum naming game in the special case where the learning domain is singular in the sense that its extent is effectively zero, i.e., no learning.)

Our model also provides novel insights into consensus dynamics. For example, there is a trade-off between the amount of knowledge and consensus. LCGM is able to provide a quantitative explanation for the phenomenon that consensus is hard to be reached among serious agents with a high LST (who know more categories). Another observation is that hubs contributed more to knowledge formation, which accordingly have larger probability to become "smart" and take lead in a game. By studying the effects of network structural characteristics on the consensus dynamics, we identify the impacts of distance and heterogeneity on consensus. While the findings are preliminary, they are useful for understanding the dynamical evolution of consensus and may even serve as the base for formulating control strategies to harness consensus dynamics, warranting further efforts in investigating learning and likelihood based games.

Acknowledgments

YCL would like to acknowledge support from the Vannevar Bush Faculty Fellowship program sponsored by the Basic Research Office of the Assistant Secretary of Defense for Research and Engineering and funded by the Office of Naval Research through Grant No. N00014-16-1-2828. ZYF and WKST would like to acknowledge support from City University of Hong Kong through Research Grant No. 7004835.

References

[1] R.V. Solé, B. Corominas-Murtra, J. Fortuny, Diversity, competition, extinction: the ecophysics of language change, J. R. Soc. Interface (2010) rsif20100110.
[2] A. Baronchelli, The emergence of consensus: a primer, R. Soc. Open Sci. 5 (2) (2018) 172189.
[3] L. Steels, Experiments in Cultural Language Evolution, John Benjamins Publishing, 2012, pp. 1–37.
[4] L. Dall'Asta, A. Baronchelli, A. Barrat, V. Loreto, Nonequilibrium dynamics of language games on complex networks, Phys. Rev. E 74 (2006) 036105.
[5] L. Dall'Asta, A. Baronchelli, A. Barrat, V. Loreto, Agreement dynamics on small-world networks, Europhys. Lett. 73 (2006) 969–975.
[6] A. Barrat, A. Baronchelli, L. Dall'Asta, V. Loreto, Agreement dynamics on interaction networks with diverse topologies, Chaos 17 (2007) 026111.
[7] Q. Lu, G. Korniss, B.K. Szymanski, The naming game in social networks: community formation and consensus engineering, J. Econ. Interact. Coord. 4 (2009) 221–235.
[8] Y. Lou, G. Chen, Z. Fan, L. Xiang, Local communities obstruct global consensus: Naming game on multi-local-world networks, Physica A 492 (2018) 1741–1752.
[9] C.W. Wu, Can stubbornness or gullibility lead to faster consensus? A study of various strategies for reaching consensus in a model of the naming game, in: IEEE International Symposium Circuits and Systems, ISCAS, 2011.
[10] R.-R. Liu, W.-X. Wang, Y.-C. Lai, G.-R. Chen, B.-H. Wang, Optimal convergence in naming game with geography-based negotiation on small-world networks, Phys. Lett. A 375 (2011) 363–367.
[11] B. Li, G. Chen, T.W.S. Chow, Naming game with multiple hearers, Commun. Nonlinear Sci. Numer. Simul. 18 (2013) 1214–1228.
[12] Y. Treitman, C. Lim, Z. W., T. A., Naming game with greater stubbornness and unilateral zealots, in: IEEE 2^{Nd} Network Science Workshop, 2013.
[13] Y. Gao, G. Chen, R.H.M. Chan, Naming game on networks: Let everyone be both speaker and hearer, Sci. Rep. 4 (2014) 6149.
[14] D. Mistry, Q. Zhang, N. Perra, A. Baronchelli, Committed activists and the reshaping of status-quo social consensus, Phys. Rev. E 92 (2015) 042805.
[15] Y. Lou, G. Chen, Analysis of the "naming game" with learning errors in communications, Sci. Rep. 5 (2015) 12191.
[16] X. Niu, C. Doyle, G. Korniss, B.K. Szymanski, The impact of variable commitment in the naming game on consensus formation, Sci. Rep. 7 (2017) 41750.
[17] G. Fu, Y. Cai, W. Zhang, Analysis of naming game over networks in the presence of memory loss, Physica A 479 (2017) 350–361.
[18] Y. Lou, G. Chen, J. Hu, Communicating with sentences: A multi-word naming game model, Physica A 490 (2018) 857–868.
[19] J. Zhou, Y. Lou, G. Chen, W.K.S. Tang, Multi-language naming game, Physica A (2018).
[20] E. Kronmüller, I. Noveck, N. Rivera, F. Jaume-Guazzini, D. Barr, The positive side of a negative reference: the delay between linguistic processing and common ground, R. Soc. Open Sci. 4 (2) (2017) 160827.
[21] D. Li, Z. Fan, W.K.S. Tang, Domain learning naming game for color categorization, PLoS One 12 (2017) e0188164.

[22] L. Steels, T. Belpaeme, Coordinating perceptually grounded categories through language: A case study for colour, Behav. Brain Sci. 28 (2005) 469–488.
[23] A. Puglisi, A. Baronchelli, V. Loreto, Cultural route to the emergence of linguistic categories, Proc. Natl. Acad. Sci. 105 (2008) 7936–7940.
[24] A. Mukherjee, F. Tria, A. Baronchelli, A. Puglisi, V. Loreto, Aging in language dynamics, PLoS One 6 (2011) 16677.
[25] P. Cui, M. Tang, Z.X. Wu, Message spreading in networks with stickiness and persistence: Large clustering does not always facilitate large-scale diffusion, Sci. Rep. 4 (2014) 6303.
[26] A. Baronchelli, V. Loreto, A. Puglisi, Individual biases, cultural evolution, and the statistical nature of language universals: The case of colour naming systems, PLoS One 10 (2015) 0125019.
[27] H.-X. Hu, G. Wen, W. Yu, Q. Xuan, G. Chen, Swarming behavior of multiple Euler–Lagrange systems with cooperation–competition interactions: An auxiliary system approach, IEEE Trans. Neural Netw. Learn. Syst. 29 (11) (2018) 5726–5737.
[28] V.M. Eguiluz, M.G. Zimmermann, Transmission of information and herd behavior: An application to financial markets, Phys. Rev. Lett. 85 (2000) 5659.
[29] R.S. Hunter, Photoelectric color difference meter, J. Opt. Soc. Amer. 48 (1958) 985–995.
[30] M.R. Luo, G. Cui, B. Rigg, The development of the CIE 2000 colour-difference formula: CIEDE2000, Color Res. Appl. 26 (2001) 340–350.
[31] SNAP: Network dataset: Facebook, http://snap.stanford.edu/data/egonets-Facebook.html, (Access August 2017).
[32] SNAP: Network dataset: email-Eu-core network, https://snap.stanford.edu/data/email-Eu-core.html, (Access June 2018).
[33] Q. Xuan, Z.-Y. Zhang, C. Fu, H.-X. Hu, V. Filkov, Social synchrony on complex networks, IEEE Trans. Cybern. 48 (5) (2018) 1420–1431.
[34] Y. Zhang, G. Wen, G. Chen, J. Wang, M. Xiong, J. Guan, S. Zhou, Gaming temporal networks, IEEE Trans. Circuits Syst. II 66 (4) (2018) 672–676.
[35] A.-L. Barabási, R. Albert, Emergence of scaling in random networks, Science 286 (5439) (1999) 509–512.
[36] H. Zhu, X. Wang, J.Y. Zhu, Effect of aging on network structure, Phys. Rev. E 68 (2003) 056121.
[37] P. Holme, B.J. Kim, Growing scale-free networks with tunable clustering, Phys. Rev. E 65 (2002) 026107.