


# SCIENTIFIC REPORTS



OPEN

## Analysis of the “naming game” with learning errors in communications

Yang Lou & Guanrong Chen

Received: 13 January 2015

Accepted: 15 June 2015

Published: 16 July 2015

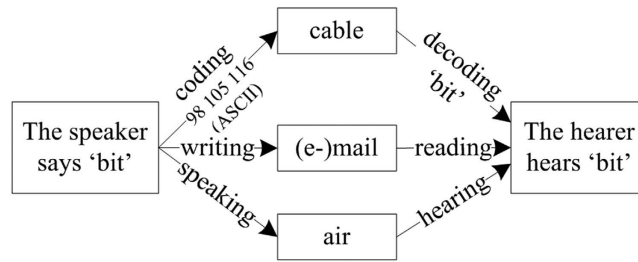
Naming game simulates the process of naming an objective by a population of agents organized in a certain communication network. By pair-wise iterative interactions, the population reaches consensus asymptotically. We study naming game with communication errors during pair-wise conversations, with error rates in a uniform probability distribution. First, a model of naming game with learning errors in communications (NGLE) is proposed. Then, a strategy for agents to prevent learning errors is suggested. To that end, three typical topologies of communication networks, namely *random-graph*, *small-world* and *scale-free* networks, are employed to investigate the effects of various learning errors. Simulation results on these models show that 1) learning errors slightly affect the convergence speed but distinctively increase the requirement for memory of each agent during lexicon propagation; 2) the maximum number of different words held by the population increases linearly as the error rate increases; 3) without applying any strategy to eliminate learning errors, there is a threshold of the learning errors which impairs the convergence. The new findings may help to better understand the role of learning errors in naming game as well as in human language development from a network science perspective.

Naming game (NG)<sup>1</sup> is a simulation-based numerical study that explores the emergence of shared lexicons in a communicating population of agents about a same object which they observed. In the minimal version of naming game, agents reach a consensus state after individual iterative *actions* and pair-wised local *interactions* among them, where *action* means learning words from external lexicons or creating words in case one has nothing in its memory and *interaction* include propagation of words among agents as well as checking the state of consensus. By testing on different models under various conditions, such as networks with realistic topologies and functional parameters as well as broadcasting and learning strategies *etc.*, it is possible to reveal some features and even principles of linguistic conventions via such extensive computer simulations.

In a general NG model<sup>1</sup>, a population of agents with either finite or infinite memory is employed<sup>2</sup>, where the agents may or may not have words in their memories initially<sup>3</sup>. The rule of the game is as follows: At each iteration, a speaker and a hearer are picked from the population at random, and the speaker transmits one word either from its memory to the hearer, or if the speaker has nothing in its memory then it picks up a word from a large enough dictionary (external lexicon), which is equivalent to creating a new word by itself. If coincidentally the hearer has the same word as the one that the speaker named, then they have consent and consequently they both clear their memories except keeping the common word; otherwise, the hearer adds the new word to its own memory as a result of learning this word from the speaker. Through running such an iterative interaction and propagation process until the game converges to a steady state where every agent keeps one and only one common word in its memory.

It is worth mentioning that the speaker and hearer must be neighbors to be able to communicate to each other, which is reflected by the connectivity of the population network in a certain topology.

Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, China. Correspondence and requests for materials should be addressed to G.C. (email: eegchen@cityu.edu.hk)



**Figure 1.** Methods of communication between a speaker and a hearer .

For example, NG on *small-world* networks was studied in<sup>4,5</sup>, and NG on *random-graph* networks and *scale-free* networks was studied<sup>6</sup>.

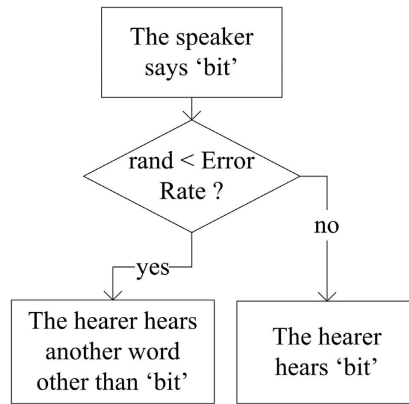
The broadcasting model of naming game, proposed by Baronchelli<sup>7</sup>, studies the following scenario of the NG: At each iteration step, one speaker is randomly picked, together with all its neighbors as hearers, from the population. Then, the game rule is followed throughout. Recently, Li *et al.*<sup>8</sup> proposed the naming game with multiple hearers (NGMH). In this game, at each iteration step, one speaker is randomly picked, but only with a portion of its neighbors as hearers, from the population. Then, the game rule is followed throughout. It turns out that the NGMH has a chance to converge only if all the hearers received and understood the same word that the speaker spoke. In the broadcasting model<sup>7</sup>, however, this will not happen, where the game is played on a one-side consensus model called the hearer-only naming game (HO-NG)<sup>7</sup>. In NGMH here, convergence means that eventually everyone, including the speaker, keeps only the transmitted word but drops all other words from its memory, while in HO-NG the speaker still keeps all words in his memory. This is the main difference between NGMH and HO-NG. Nevertheless, both simulation results<sup>7,8</sup> show that the existence of multiple hearers accelerates the convergence speed as compared to the original NG model. Based on the multiple hearers mode, Gao *et al.*<sup>9</sup> further investigated a generalized version with multiple speakers and multiple hearers (called naming game in groups, or NGG), in which it lets every agent in a selected group from the population to play the role of both a speaker and a hearer simultaneously. It demonstrated that the convergence speed is faster when the number of participating agents increases.

Considering the real-life scenario of human communications, the efficiency of agent learning and information propagation affect the convergence of a naming process. Language acquisition is generally error-prone. But, interestingly, as pointed out by Nowak *et al.*<sup>10</sup>, learning errors can help prevent the linguistic system from being trapped in sub-optimum situations by increasing the diversity, thereby leading to the evolution of a more efficient language where the system is evaluated by the function of payoff. Nowak *et al.*<sup>10</sup> also found some thresholds of the error rate for certain learning models, below which the system gains advantage from learning errors while above which mistakes will impair the system, say, reducing the payoff. Moreover, *noise* may lead to recurrently converging states of a Markov chain model<sup>11</sup>, which is considered beneficial in better detecting social interactions. Therefore, errors or noises may be expected to affect the language system positively, to some extent, in the two NG models introduced above. However, our study in this paper includes neither the formation of a more efficient language as in<sup>10</sup>, nor to obtain a series of recurrently converging states as in<sup>11</sup>. Instead, we study learning errors from the perspective of language communications in a common sense that learning errors likely bring negative effects. Every individual makes mistakes sometimes in real life, but if an individual is more experienced then he will probably make fewer mistakes or even know how to avoid making mistakes. From this consideration, we study a real-life scenario where agents are all error-prone initially but then they gradually learn to avoid making further errors therefore eventually all agents are error-free in the NG process, so that the whole population will reach consensus asymptotically.

Specifically, we will study NG with communication errors during pair-wise conversations, where errors are represented by error rates in a uniform probability distribution. An NG model with learning errors in communications (NGLE) is first proposed, followed by a strategy for agents to prevent learning errors. Three typical topologies of communication networks (*random-graph*, *small-world* and *scale-free* networks) with different parameters are then investigated, to reveal the effects of various learning errors on the convergence performance, with simulation results presented and analyzed. Finally, conclusions are drawn with some discussions.

## Methodology

**We first discuss naming game with learning errors in communications.** Errors in communications could be from different sources such as the speakers, the communication media, and the hearers. There are also many reasons that could lead to errors, for example ambiguous pronunciations, different styles of handwriting, and imperfect coding and decoding techniques of telecommunications, *etc.*, as shown in Fig. 1 where all the boxes and arrows in the figure could lead to errors.



**Figure 2.** The communication model with learning errors .

In this paper, we do not study the reason and source of communication errors, but investigate how learning errors propagate and affect the NG convergence. We assume that the occurrence of any type of error can be represented by a numerical value, *error rate*. We also assume that the results caused by different types of error are equivalent to the situation that *the hearer learns a wrong word from the speaker*, which is represented by *learning error* and measured by *error rate*. For example, a word 'bit' sent by the speaker will be learned by the hearer correctly under a certain probability, while under its complementary probability the hearer will receive a wrong word. Figure 2 provides an illustration, where the conditional judgment rhombus represents all the different communication media shown in Fig. 1.

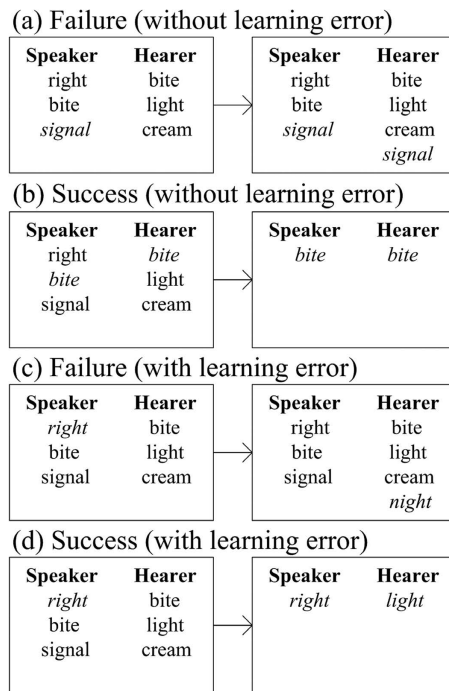
In real life, both people and machines produce errors, but it is not acceptable if it happens perpetually, since people will learn and machines can be modified to work correctly. The NG system includes multiple agents and a great deal of lexicons, where lexicons are objective and would not be wrong themselves, but un-experienced agents are error-prone. In an NG, the agents should be able to learn to prevent or at least reduce learning errors by themselves, or they would be considered unreliable. There are many ways for the agents to be more reliable. Empirically and practically, training or educating could be one effective way to improve the correctness of agents' expressions and understandings in their conversations. Double-checking before sending out a word and after receiving it is another solution. In addition, using a redundant medium or some redundant information, e.g., sending out an email accompanied with an auxiliary voice message, or just attaching a redundant message to verify if the hearer receives the information timely and correctly.

In the proposed NGLE model, we adopt the first method mentioned above. We assume that, when an agent has experience as a speaker before, it will have lower probability to make mistakes. This follows the old saying that *teaching and learning grow hand in hand*, which implies that, when an agent has played the role as a speaker, it benefits from the experience, so it is an experienced agent and would rarely make mistakes. As a result, in the early stage of lexicon propagation, communications among the agents are mainly error-prone, while in the later stage the probability of error appearing would be gradually decreased until all agents have participated as speaker at least once, such that the population will finally converge.

**We next discuss the NGLE model in detail.** In the NGLE model, an agent in the population is represented by a node in the network. If two agents are neighbors, the corresponding two nodes in the network are connected by an edge; therefore, they can communicate with each other. Thus, the terms *node* and *agent* are interchangeable throughout the paper.

The network model of NGLE is summarized as follows.

1. A population of  $n$  agents connected in a certain topology is initialed with empty memories, along with an external vocabulary of very large size.
2. At each iteration, a speaker is randomly picked from the population:
  - If the speaker has empty memory, then it randomly picks a *word* from the external vocabulary;
  - Otherwise, the speaker randomly picks a *word* from its memory.
3. A hearer is randomly picked from the neighborhood of the speaker:
  - If the hearer has never been a speaker before, then within the error rate  $\rho$ , it receives *word'* other than *word*;
  - Otherwise, the hearer receives the *word* correctly.



**Figure 3. Four situations in one iteration during pair-wise communication** ((a) the hearer learns from the speaker; (b) consensus; (c) the hearer learns a wrong word from the speaker; (d) pseudo consensus)

4. The hearer checks if it has the same word in memory as the received one:

If the word was already in its memory, learning is successful, so both the speaker and the hearer clear out their memories except keeping the only word which was just communicated; Otherwise, the hearer adds the new word into its memory.

5. Repeat Step 2 to Step 4 iteratively, until all nodes keep one and only one same word, or until the number of iterations reaches a pre-defined (large enough) value of termination.

In Step 3 all types of possible errors are presented by a single numerical value, the *error rate*. In the following, both Step 3 and Step 4, namely the process of communication, is interpreted. It is quite different from the minimal NG<sup>1</sup>, which has two possible results (*success* and *failure*) after one iteration in communication, in that the NGLE has four possible situations.

At each iteration step, a speaker is randomly selected from the population, and so is a hearer but from the speaker's neighborhood represented by the connected edges in the network. This is a direct strategy<sup>5,6</sup> of agent selection, in which hub nodes have a lower probability to be selected as a speaker, as compared with the so-called reverse strategy<sup>5</sup>, which selects the hearer first and then selects a speaker from its neighbors, both at random.

The communication process starts as soon as both the speaker and the hearer have been selected. As shown in Fig. 3(a), the first situation is that there is no learning error, thus the word *signal* is sent from the speaker to the hearer directly and correctly. Then, the hearer checks the received word with its memory and finds *signal* is not included therein, so the hearer adds the word *signal* into its memory. This situation is named “failure without learning error”; it is analogous to the situation that the hearer directly learns a new word *signal* from the speaker. The second situation shown in Fig. 3(b) is the state of consensus, where the hearer has the word *bite* sent by the speaker and both of them erase their memories but keep the common word *bite* only. These two situations are exactly the same as that in the minimal naming game<sup>1</sup>.

Now, there are two more new situations in the NGLE. The third and fourth situations are related to learning errors. In Fig. 3(c), the speaker says a word *right* to the hearer, under a certain probability (error rate), the hearer receives a wrong word *night*; after checking its memory, the hearer learns a “new word” that was not included in its memory. Note that if the speaker says a word that should lead to consensus (namely, the hearer actually had it in memory), then in this case they both miss an opportunity to be successful.

The fourth situation is an interesting one, where the speaker says *right* but the hearer receives a wrong word *light*, and coincidentally the hearer holds the word *light*. Then, an ambiguous consensus happens:

from the consensus reaction of the hearer, the speaker considers that the hearer agrees with his word *right*, while the hearer actually agrees with *light*. This situation is called pseudo consensus. The result of pseudo consensus is that the speaker deletes all words from its memory other than *right*, while the hearer clears out its memory leaving only *light*. This is analogous to the misunderstanding between the speaker and the hearer in real life, while neither of them realizes the existence of an error.

## Results

The detailed processes of emergence, propagation and consensus of shared lexicons are examined by employing three typical network topologies, *random-graph* (RG)<sup>6,12</sup>, *small-world* (SW)<sup>4,5,13</sup> and *scale-free* (SF) networks<sup>6,14</sup>. As suggested by Baronchelli *et al.*<sup>15</sup>, we perform extensive numerical simulations to study the various naming games with comparisons. As implemented in<sup>8</sup>, when the memory of the picked speaker is empty, it will randomly pick a word from the vocabulary instead of randomly creating a new word; similarly, when learning error occurs, the hearer will randomly pick a word from the vocabulary and this word should be different from the word received.

**Simulation setup.** We simulate nine types of networks, each with 2,000 nodes, among which three types are studied on population sizes of 1,000 and 3,000 (see Table 1), and additionally on population sizes of 200 and 500 (see Table S3 of the Supplementary Information (SI)<sup>16</sup>), aiming to reveal the possible scaling property of the model. To reduce the randomness and increase the confidence level, for each type of network, we perform 20 independent simulations and then take an average. Hence, the data shown in Table 1 and the curves shown from Fig. 4 to Fig. 7 are all statistically averaged results from the 20 independent simulation trials. The size of the external vocabulary is 10,000.

Table 1 shows the detailed network setting. We employ three types of *random-graph* networks by altering the connection probability, including RG/0.03, RG/0.05 and RG/0.1, and three types of *scale-free* networks by altering the number of adding nodes at each step, including SF/25, SF/50 and SF/75. For *small-world* networks, they all have two parameters to adjust, *i.e.*, the number of neighborhoods and the rewiring probability, thus six combinations are obtained by altering each parameter with three values each, including SW/20/{0.1, 0.2, 0.3} and SW/40/{0.1, 0.2, 0.3}. We use networks of 2,000 nodes as the default setting in all simulations, and denote the networks of 1,000 and 3,000 nodes by suffixing 1K and 3K with brackets of the relevant network names, as shown in Table 1. The networks of 200 and 500 nodes are denoted by 0.2K and 0.5K with brackets of the relevant network names, as shown in Table S3 of the SI. For brevity, the parameter setting of SW/40/{0.1, 0.2, 0.3} is shown in Table S1 of the SI, and the simulation results are presented and analyzed in the first section of SI. The study on the population size is presented in the second section of the SI, where we study population sizes of 200, 500, 1,000, 2,000 and 3,000, respectively.

Because the different initial states, say one-word-per-agent or no words at all, they would generate different convergence curves<sup>3</sup>. For the purpose of studying the consensus patterns of the NGLE, assume that there is no initial word in the memory of each agent<sup>8</sup>. A comprehensive study is carried out on the values of the learning error rate,  $\rho$ , which varies from 0.001 to 0.009 with an increment of 0.001, and from 0.01 to 0.09 with an increment of 0.01, and from 0.1 to 0.5 with an increment of 0.1, respectively, along with the reference group having  $\rho = 0$ . Thus, totally 24 groups are studied. We did not consider the situation with error rate greater than 0.5, since it means most of the information propagated is incorrect, which is out of question.

The last parameter to introduce is the maximum number of iterations, which is set to be 10,000,000 in our simulations. This value is empirically large enough for the fifteen networks with 24 error rates. Therefore, in each single run, the population definitely reaches a consensus state before the termination value of ten million iterations.

**Convergence process.** First, we analyze the relationship between the *number of total words* in the population and different values of the error rate. For clarity, we list only 5 representative sets of data out of 24 sets in total. Some common grounds for all types of networks include that the *number of total words* starts from zero, since we assume that agents are memory-free initially, and then it goes through an ascending (learning is dominant, Fig. 3(a,c)) and descending process (consensus is dominant, Fig. 3(b,d)), and finally it converges to 2,000 words, which is exactly the number of agents. In the consensus state, each agent holds one and only one same word.

As can be seen from Fig. 4, two types of networks (Fig. 4(d) SW/20/0.1, and (e) SW/20/0.2) converge after more than 1,000,000 iterations, while the rest seven converge in less numbers of steps. The convergence curves of the error rate less than or equal to 0.01 cannot be visually distinguished from the curves without learning errors. But when the error rate increases to 0.1, the curve difference is recognizable, and when the error rate is 0.5, the curve is significantly different from other curves. This means that when the error rate is less than or equal to 0.01, the influence of the learning error on the number of total words is insignificant, while if the error rate is 0.1, it becomes non-negligible, and for 0.5, it becomes quite significant, which means that it generates more words for agents to store in their memories temporarily (but they will be dropped finally). Therefore, from the viewpoint of memory cost, when the error rate is less than 0.01, it does not require more memory than that without learning

Notation	Network type	Number of nodes	Average degree	Average path length	Average clustering coefficient
RG/0.03	Random-graph network with $P = 0.03$	2,000	59.97	2.1305	0.0300
RG/0.05 (1K)	Random-graph network with $P = 0.05$	1,000	49.97	2.0280	0.0500
RG/0.05	Random-graph network with $P = 0.05$	2,000	99.96	1.9564	0.0500
RG/0.05 (3K)	Random-graph network with $P = 0.05$	3,000	149.98	1.9505	0.0500
RG/0.1	Random-graph network with $P = 0.1$	2,000	199.92	1.9000	0.1000
SW/20/0.1	Small-world network with $K = 20$ and $RP = 0.1$	2,000	40.00	2.8251	0.5360
SW/20/0.2 (1K)	Small-world network with $K = 20$ and $RP = 0.2$	1,000	40.00	2.4657	0.3862
SW/20/0.2	Small-world network with $K = 20$ and $RP = 0.2$	2,000	40.00	2.6963	0.3806
SW/20/0.2 (3K)	Small-world network with $K = 20$ and $RP = 0.2$	3,000	40.00	2.7936	0.3783
SW/20/0.3	Small-world network with $K = 20$ and $RP = 0.3$	2,000	40.00	2.6133	0.2597
SF/25	Scale-free with 26 initial nodes and 25 new edges added at each step	2,000	49.66	2.2312	0.0760
SF/50 (1K)	Scale-free with 51 initial nodes and 50 new edges added at each step	1,000	97.39	1.9044	0.1954
SF/50	Scale-free with 51 initial nodes and 50 new edges added at each step	2,000	98.69	1.9725	0.1217
SF/50 (3K)	Scale-free with 51 initial nodes and 50 new edges added at each step	3,000	99.13	2.0237	0.0918
SF/75	Scale-free with 76 initial nodes and 75 new edges added at each step	2,000	147.10	1.9273	0.1602

**Table 1. Network settings in simulations.**

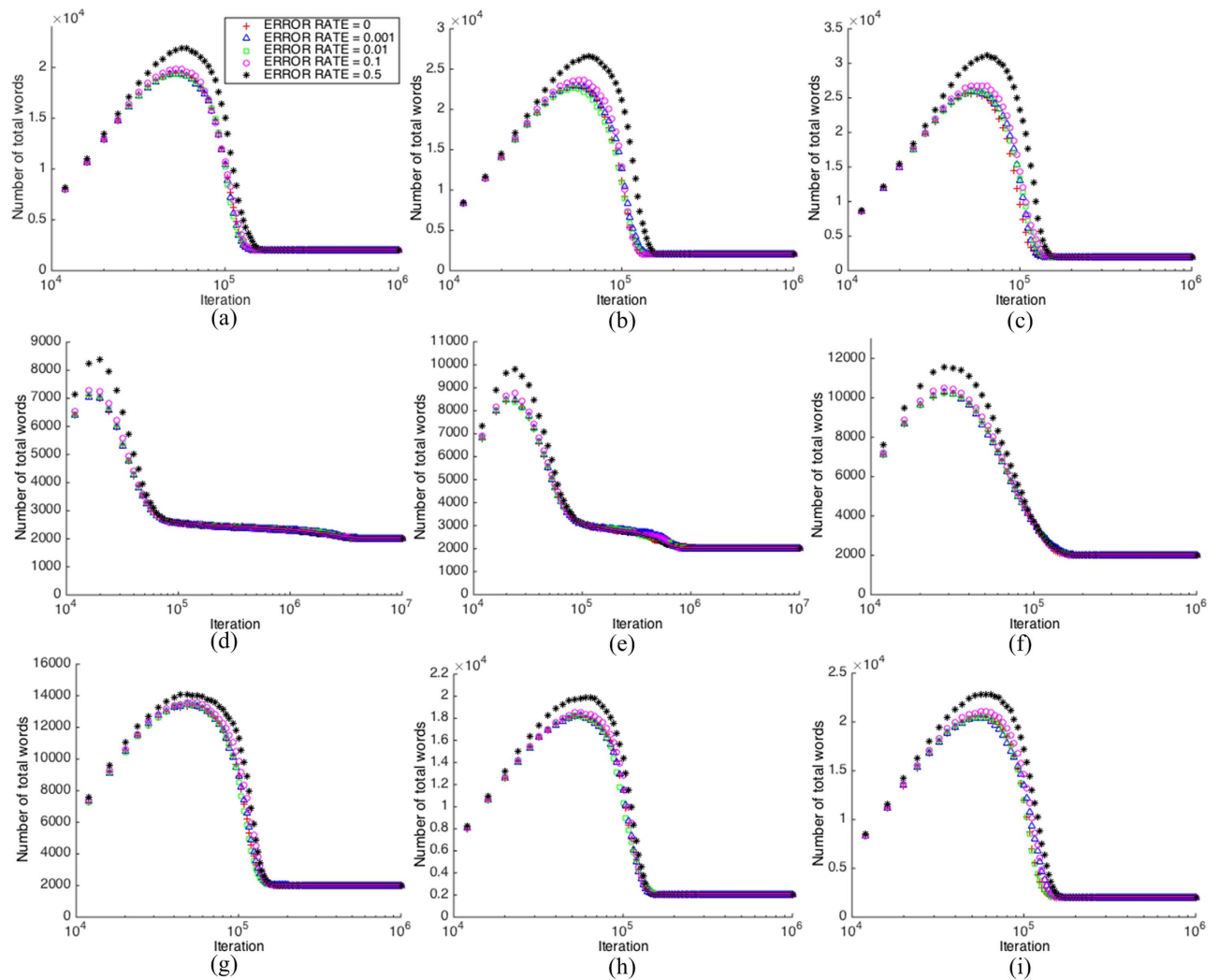
errors; however, when it increases to 0.1, the extra memory cost is recognizable, and when it is 0.5, the cost is quite significant.

Next, we analyze the relationship between the *number of different words* and different values of the error rate during the convergence process. The *number of different words* displaces a similar convergence curve as that of the *number of total words*, which starts with zero and then converges to one in the consensus state. This means that finally the whole population holds one same word only, but it is impossible to predict which word it will converge to. As can be seen from Fig. 5, for all nine networks, when the error rate is less than or equal to 0.01, the difference to the non-error curve is negligible, but when the error rate is 0.1, the curve is almost as twice higher as those with the error rate less than or equal to 0.01 during a long period before settling to consensus. Figure 5 actually tells the same story as Fig. 4 does; that is, an error rate less than or equal to 0.01 brings an insignificant influence to the required size of memory, but error rate 0.1 cannot be ignored, while 0.5 is significant.

One explanation for the phenomena shown in Figs 4 and 5 is that a larger learning error rate gives more chance to randomly pick a word from the vocabulary, so that it is more probable to bring new and different words into the population when learning errors exist, therefore the number of total words and the number of different words both increase when the learning error rate is high.

We found that the convergence processes in terms of both the number of total words and the number of different words are scalable when the size of the network varies. Figs S6 and S7 in the SI show the convergence curves of these two terms, when the population sizes of the agents are 200, 500, 1,000 and 3,000, respectively. As can be seen from Figs S6 and S7, the larger the number of nodes it contains, the slower it reaches the consensus states, while the basic profile of the curves are similar.

**Convergence time and success rate.** In the above, we found that when influenced by learning errors, it may require more memory for storage. Now, we examine the convergence time, which refers to the number of iteration when the agents in the population reach consensus. Table 2 presents the increment of convergence time affected by learning errors, where + sign means the convergence is delayed while – sign means advanced. By using the Mann-Whitney U test (at  $\alpha = 0.05$  significant level), the result shows that, for four networks RG/0.05, RG/0.1, SF/75, and SF/50(3K), when the learning error is 0.5, the



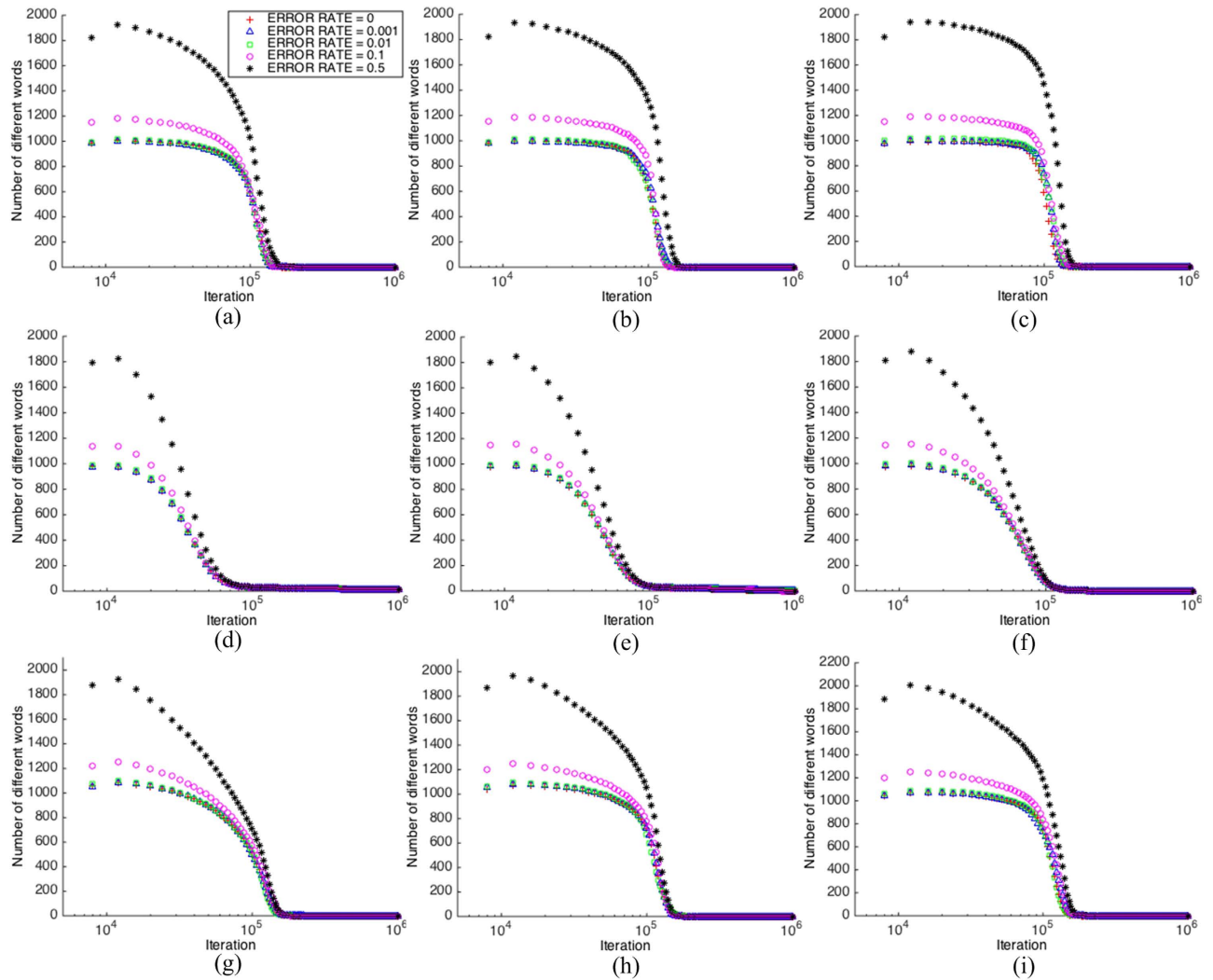
**Figure 4.** The convergence process in terms of the number of total words in a network ((a) RG/0.03; (b) RG/0.05; (c) RG/0.1; (d) SW/20/0.1; (e) SW/20/0.2; (f) SW/20/0.3; (g) SF/25; (h) SF/50; (i) SF/75)

increment of convergence time is statistically significant. However, in all the rest cases, the increment of convergence time caused by learning errors is statistically insignificant. Another detailed increment relationship which contains 24 values of different error rates is presented in Tables S4, S5 and S6 of the SI.

We use the term *success rate* in a similar fashion as defined in<sup>8</sup>, but consider also the pseudo consensus situation as a success. Thereby, the success rate is calculated by the number of iterations of (pseudo) consensus during each of 10 iterations, divided by 10.

As shown in the figures, different types of networks show different characteristics of the success rate. The curves of RG/0.03 (Fig. 6(a,b and c)) are smooth and simple, which is due to the homogeneity of random-graphs, where the population of nodes evolve towards the consensus state gradually and uniformly. The success rate curves of the other networks and small-world networks SW/40/{0.1, 0.2, 0.3} are shown in Fig. S3 of the SI. However, for the small-world network SW/20/0.2, shown in Fig. 6(d,e and f), as the value of the average path length increases, the curves become oscillatory. A longer average path length implies that it is more difficult to reach consensus, thus, the agents diverge during a period of time and it disturbs the success rate significantly. As for the scale-free network SF/50, shown in Fig. 6(g,h and i), especially in the latter two figures, not only the value of the average path length increases, but also the value of the average clustering coefficient decreases. These both increase the difficulty to achieve consensus among the agents. The curves shown in Fig. 6 indicate that learning errors neither increase or decrease the success rate, nor change the shape of the curves of the success rate.

**Maximum number of different words.** As learning errors affect the required memory size more than the convergence delay, we next examine the relationship between the *maximum number of different words* and different error rates.



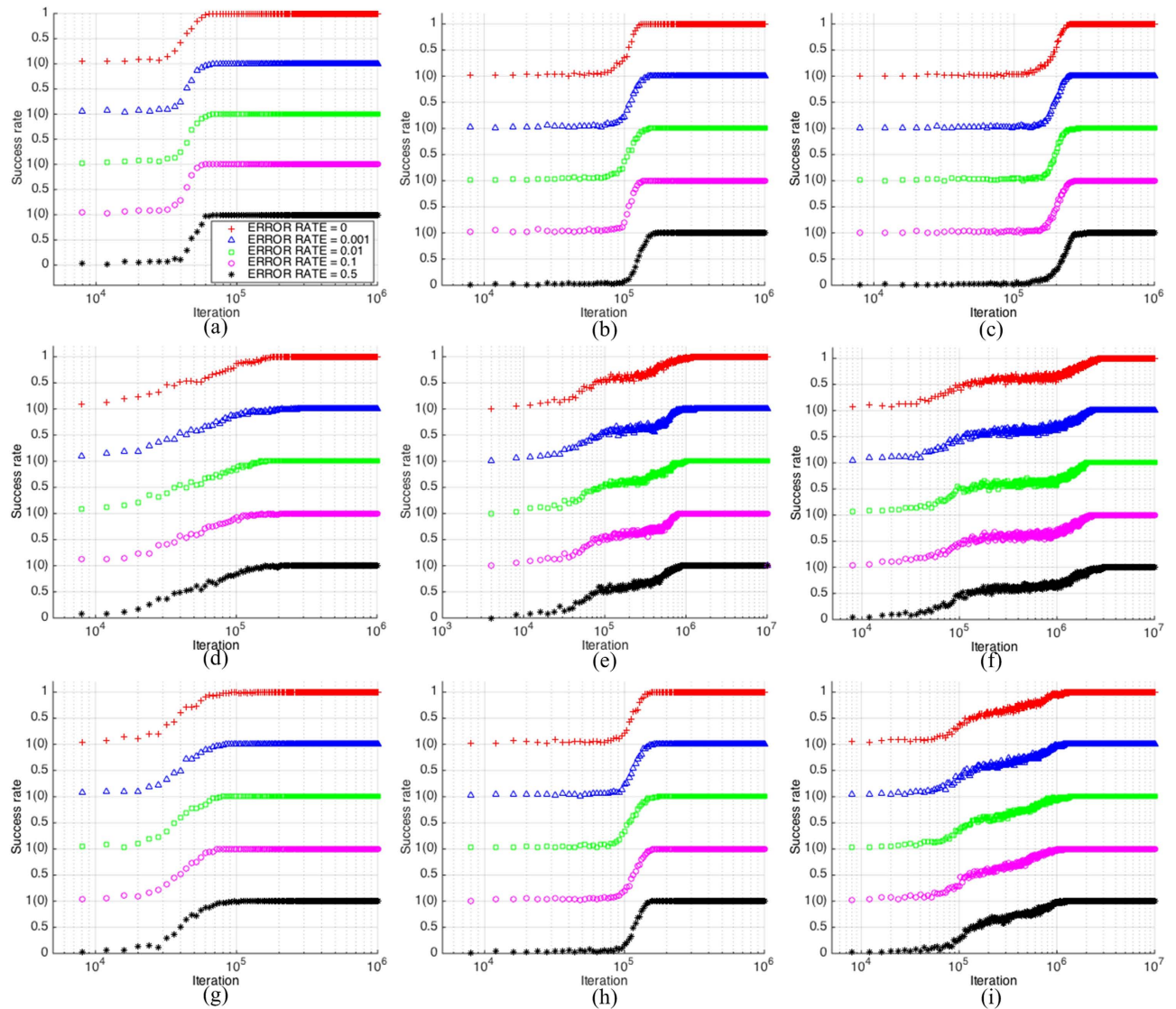
**Figure 5.** The convergence process in terms of the number of different words in a network ((a) RG/0.03; (b) RG/0.05; (c) RG/0.1; (d) SW/20/0.1; (e) SW/20/0.2; (f) SW/20/0.3; (g) SF/25; (h) SF/50; (i) SF/75)

In the NGLE model, we assume that 1) agents are possible to have learning errors, and 2) agents will learn to avoid making further errors by being speakers. Figure 7(a,b and c) show three random-graph networks; three small-world networks with 20 neighbors, and three scale-free networks. The figures in this case show an almost perfectly *linear* relationship between the *maximum number of different words* and the error rate. The straight lines in Fig. 7(a,b and c) are the fitted curves of the numerical data, where colors are in accordance with data for easy reference.

In addition, we repeat the simulations under the same conditions, except for the second assumption above, allowing agents to make errors throughout the process. In this case, the population may not converge at all; therefore, we stop the simulation after 10,000,000 iterations. With 20 independent runs, the averaged results are shown in Fig. 7(d,e and f). In this case, the data cannot be fitted into straight lines, but can be fitted to polynomial curves with degree equal to or greater than two. In those figures, we plot the fitted quadratic curves as reference.

When agents can learn to avoid making further errors by becoming speakers, statistically only a portion of agents introduce learning errors once and once only. This is because the external vocabulary is very large, so any learning error is likely to introduce a new different word into the population, thus the *maximum number of different words* increases linearly as the error rate increases. However, if learning errors are not to be avoided, the speakers would constantly introduce new words, which might lead the population to a non-convergent situation, so that the curve of the maximum number of different words will become a polynomial of higher degree.

Figure 7(g,h and i) show some curves of similar nature for two cases of population sizes, 1,000 and 3,000, respectively. In these figures, for networks of 3,000 nodes, only 5 sampled data are displayed since generating 3,000-node networks and then analyzing 24 sampled data had turned out to be time-unaffordable on a high-speed computer.



**Figure 6.** Success rate curves ((a) RG/0.05(1K); (b) RG/0.05; (c) RG/0.05(3K); (d) SW/20/0.2(1K); (e) SW/20/0.2; (f) SW/20/0.2(3K); (g) SF/50(1K); (h) SF/50; (i) SF/50(3K))

**Convergence thresholds.** To ensure convergence, namely for the population of agents to reach a consensus state before the number of iterations exceeds 10,000,000, the NGLE model employs a rule that when an agent has once been a speaker, it would not make learning error anymore in the future. In the case that all agents continuously make errors during communications, the population may not converge within the pre-defined maximal number of iterations, or even never converge.

As proposed by Nowak *et al.*<sup>10</sup>, there is an optimal error rate that maximizes the performance of the parental learning and role model learning system, but for the random learning model this would not work at all when error rate is greater than a small threshold. Since we do not use the system payoff to evaluate the performance, we study the threshold of the error rate that affects the convergence of the population.

Figure 8 shows the statistic results of the simulations on the threshold. For each type of network, we carry out 20 independent simulations. For each single simulation, we set the initial error rate be 0. If, under the current error rate, the population converges within the maximal number of iterations, then we increase the error rate by an incremental step size of 0.0001. This process repeats until the population does not converge under a certain error rate, and then we record the rate as the threshold of this simulation. It is worth mentioning that, not only suggested by<sup>10</sup>, but also by our trial-and-error simulations, incremental step of error rate less than 0.0001 makes no sense to the results for networks having less than 2,000 nodes.

For each box figure shown in Fig. 8, the blue box represents that the central 50% data lie in this section; the red bar is the median value of all 20 datasets; the upper and lower black bars are the greatest and least values, excluding outliers; and finally the red pluses represent the outliers. As can be seen from the box figure, all the random-graph and small-world networks have similar thresholds for the error rate that locate between 0.0061 and 0.0067. More specifically, in most cases, when the error rate increases to around

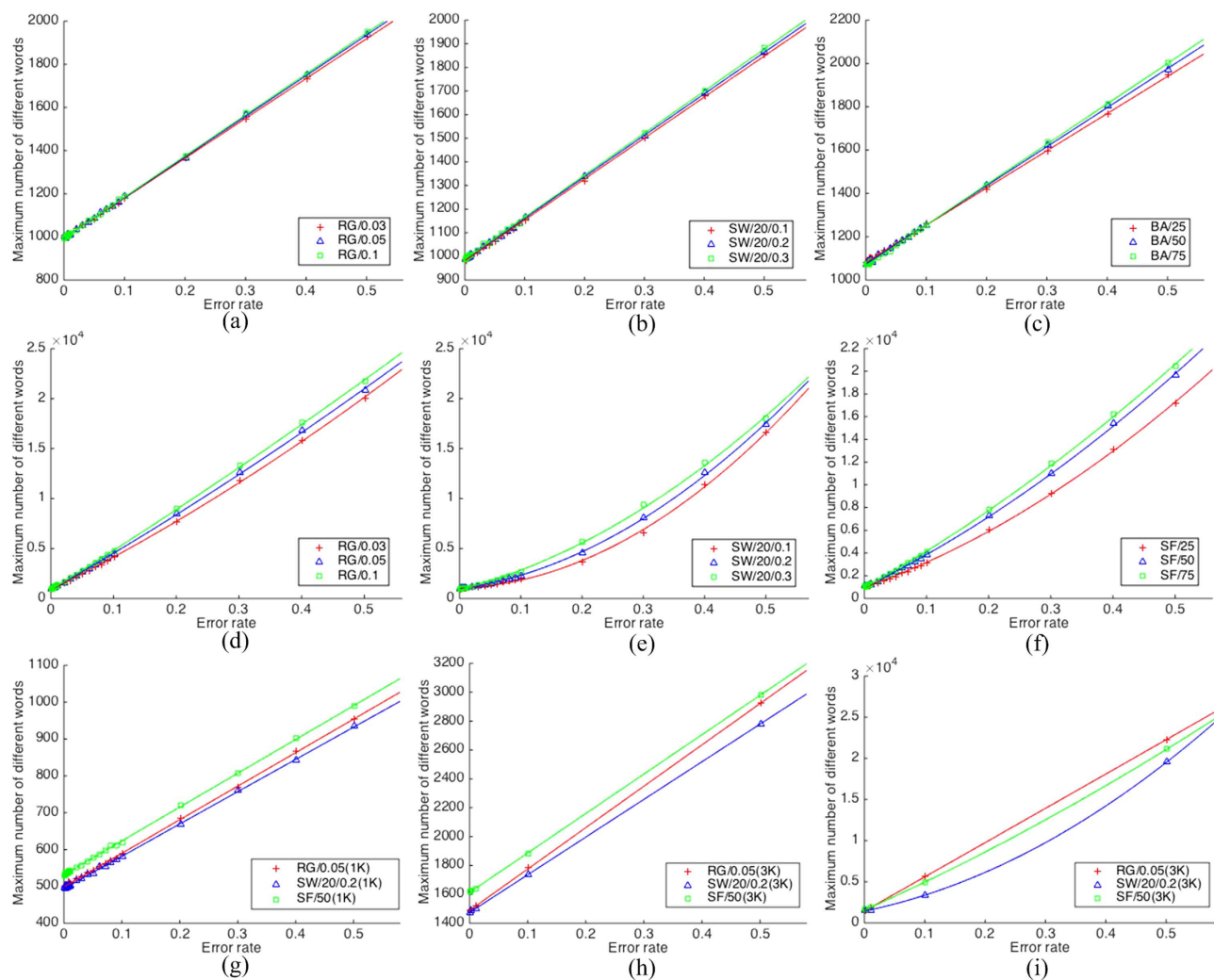
<b>Error Rate</b> <b>Network</b>	<b>0</b>	<b>0.001</b>	<b>0.01</b>	<b>0.1</b>	<b>0.5</b>
RG/0.03	1.32E+05	1.34E+05	1.30E+05	1.39E+05	1.41E+05
(Increment)	NA	+0.0210	−0.0097	+0.0563	+0.0704
RG/0.05	1.30E+05	1.25E+05	1.29E+05	1.28E+05	1.48E+05
(Increment)	NA	−0.0407	−0.0118	−0.0185	+0.1322 (*)
RG/0.1	1.29E+05	1.25E+05	1.31E+05	1.33E+05	1.42E+05
(Increment)	NA	−0.0328	+0.0128	+0.0298	+0.1024(*)
SW/20/0.1	2.89E+06	3.42E+06	2.81E+06	3.49E+06	3.12E+06
(Increment)	NA	+0.1832	−0.0280	+0.2086	+0.0782
SW/20/0.2	5.67E+05	5.01E+05	5.75E+05	6.65E+05	5.48E+05
(Increment)	NA	−0.1162	+0.0133	+0.1719	−0.0332
SW/20/0.3	1.46E+05	1.40E+05	1.49E+05	1.55E+05	1.55E+05
(Increment)	NA	−0.0428	+0.0221	+0.0602	+0.0617
SF/25	1.58E+05	1.59E+05	1.49E+05	1.56E+05	1.61E+05
(Increment)	NA	+0.0017	−0.0600	−0.0166	+0.0197
SF/50	1.49E+05	1.53E+05	1.51E+05	1.44E+05	1.57E+05
(Increment)	NA	+0.0285	+0.0157	−0.0321	+0.0566
SF/75	1.40E+05	1.46E+05	1.45E+05	1.45E+05	1.56E+05
(Increment)	NA	+0.0423	+0.0387	+0.0352	+0.1191(*)
RG/0.05(1K)	5.32E+04	5.34E+04	5.53E+04	5.20E+04	5.72E+04
(Increment)	NA	+0.0040	+0.0401	−0.0225	+0.0764
RG/0.05(3K)	2.31E+05	2.31E+05	2.34E+05	2.38E+05	2.53E+05
(Increment)	NA	−0.0009	+0.0135	+0.0317	+0.0935
SW/20/0.2(1K)	1.18E+05	1.14E+05	1.07E+05	9.98E+04	1.10E+05
(Increment)	NA	−0.0348	−0.0958	−0.1537	−0.0673
SW/20/0.2(3K)	1.82E+06	1.60E+06	1.57E+06	1.63E+06	1.77E+06
(Increment)	NA	−0.1182	−0.1376	−0.1048	−0.0272
SF/50(1K)	5.77E+04	6.20E+04	5.89E+04	5.84E+04	6.29E+04
(Increment)	NA	+0.0736	+0.0215	+0.0115	+0.0897
SF/50(3K)	2.44E+05	2.52E+05	2.53E+05	2.54E+05	2.76E+05
(Increment)	NA	+0.0323	+0.0361	+0.0394	+0.1292(*)

**Table 2.** The number of average convergence time, increment relationship between average convergence time with different values of the error rate (\*indicates the average convergence time is significantly delayed by learning errors).

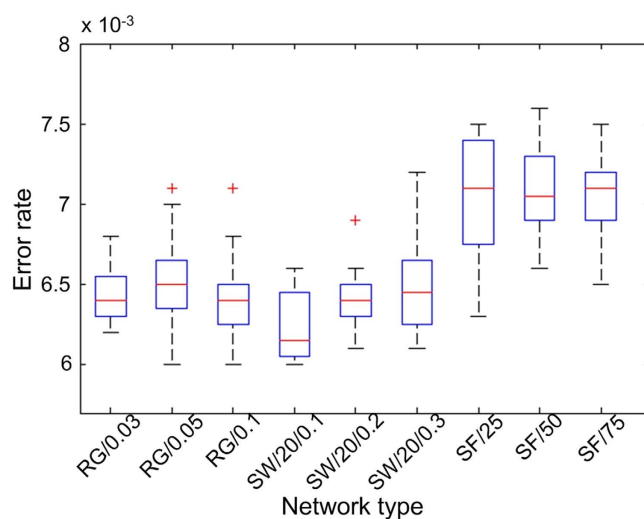
0.0067 or above, the population would probably not converge within the maximal number of iterations. For the three scale-free networks, this threshold is around 0.0068 to 0.0073, which means that the tolerance of learning errors in scale-free networks is higher than in random-graph and small-world networks.

## Conclusions

In this paper, we proposed a novel model of naming game with learning errors in communications (NGLE) and study it by means of extensive and comprehensive computer simulations. We found that if the agents have some learning errors but can learn to avoid making further errors, the convergence will be slightly affected, either accelerated or delayed within a relative small range, which depend also on the different topologies and parameter settings of the underlying communication networks. However, during the period between the initial and the final consensus states, agents in the NGLE learn and discard more words than the agents without learning errors, which means that learning error requires more memory space from the agents. We also realized that the NGLE model has an interestingly linear relationship between the maximum number of different words throughout the convergence process and the error rate, *i.e.*, the higher the error rate is, the larger the maximum number of different words it will have, which are in a linearly proportional relation. In addition, we identified the statistical range of the



**Figure 7.** The relationship between the error rate and the maximum number of different words ((a) RG/{0.03, 0.05, 0.1}; (b) SW/20/{0.1, 0.2, 0.3}; (c) SF/{25, 50, 75}; (d) RG/{0.03, 0.05, 0.1} without convergence; (e) SW/20/{0.1, 0.2, 0.3} without convergence; (f) SF/{25, 50, 75} without convergence; (g) RG/0.05(1K), SW/20/0.2(1K), and SF/50(1K); (h) RG/0.05(3K), SW/20/0.2(3K), and SF/50(3K); (i) RG/0.05(3K), SW/20/0.2(3K), and SF/50(3K), without convergence)



**Figure 8.** The box plot of learning error rate threshold in different types of networks .

error rate threshold, above which the population would not converge in the case without any strategy to prevent learning errors.

It is believed that the new findings reported in this paper are meaningful and helpful to enhance our understanding of the role of learning errors in naming games as well as in evolution of languages.

## References

1. Baronchelli, A., Felici, M., Loreto, V., Caglioti, E. & Steels, L. Sharp transition towards shared vocabularies in multi-agent systems. *J. Stat. Mech: Theory Exp.* **6**, P06014 (2006).
2. Wang, W. X., Lin, B. Y., Tang, C. L. & Chen, G. R. Agreement dynamics of finite-memory language games on networks. *Eur. Phys. J. B* **60**, 529–536 (2007).
3. Lu, Q., Korniss, G. & Szymanski, B. K. The naming game in social networks: community formation and consensus engineering. *J. Econ. Interact. Coord.* **4**, 221–235 (2009).
4. Dall'Asta, L., Baronchelli, A., Barrat, A. & Loreto, V. Agreement dynamics on small-world networks. *Europhys. Lett.* **73**, 969 (2006).
5. Liu, R. R., Jia, C. X., Yang, H. X. & Wang, B. H. Naming game on small-world networks with geographical effects. *Physica A* **388**, 3615–3620 (2009).
6. Dall'Asta, L., Baronchelli, A., Barrat, A. & Loreto, V. Nonequilibrium dynamics of language games on complex networks. *Phys. Rev. E* **74**, 036105 (2006).
7. Baronchelli, A. Role of feedback and broadcasting in the naming game. *Phys. Rev. E* **83**, 046103 (2011).
8. Li, B., Chen, G. R. & Chow, T. W. Naming game with multiple hearers. *Commun. Nonlinear Sci. Numer. Simul.* **18**, 1214–1228 (2013).
9. Gao, Y., Chen, G. R. & Chan, R. H. M. Naming game on networks: let everyone be both speaker and hearer. *Sci. Rep.* **4**, 6149, (2014).
10. Nowak, M. A., Plotkin, J. B. & Krakauer, D. C. The evolutionary language game. *J. Theor. Biol.* **200**, 147–162 (1999).
11. Lim, C. C. & Zhang, W. Noisy naming games, partial synchronization and coarse-graining in social networks. *Proc. of Network Science Workshop (NSW), IEEE.* 25–29 doi: 10.1109/NSW.2011.6004654 (2011).
12. Erdős, P. & Rényi, A. On the strength of connectedness of a random graph. *Acta Math. Hungar.* **12**, 261–267 (1961).
13. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442 (1998).
14. Baronchelli, A., Dall'Asta, L., Barrat, A. & Loreto, V. The role of topology on the dynamics of the naming game. *Eur. Phys. J. Special Topics* **143**, 233–235 (2007).
15. Baronchelli, A., Loreto, V. & Steels, L. In-depth analysis of the naming game dynamics: the homogeneous mixing case. *Int. J. Mod. Phys. C* **19**, 785–812 (2008).
16. Lou, Y. & Chen, G. R. Supplementary information for the paper “Analysis of the “naming game” with learning errors in communications”. <http://www.ee.cityu.edu.hk/~gchen/pdf/NGLE-SI.pdf> (Date of access: 17/12/2014).

## Acknowledgement

This research was supported by the Huawei Technologies Co. Ltd.

## Author Contributions

Y.L. designed and performed the numerical analyses, wrote the main text and supplementary information, and generated the tables and figures. G.C. supervised the project, suggested some additional topics, reviewed the game-theoretic background and put the entire work into context. Both Y.L. and G.C. reviewed, revised and finalized the manuscript.

## Additional Information

**Supplementary information** accompanies this paper at <http://www.nature.com/srep>

**Competing financial interests:** The authors declare no competing financial interests.

**How to cite this article:** Lou, Y. and Chen, G. Analysis of the “naming game” with learning errors in communications. *Sci. Rep.* **5**, 12191; doi: 10.1038/srep12191 (2015).



This work is licensed under a Creative Commons Attribution 4.0 International License. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in the credit line; if the material is not included under the Creative Commons license, users will need to obtain permission from the license holder to reproduce the material. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>