Collective Learning of an Emergent Vocabulary: Naming Game with Reinforcement Learning

İsmet Adnan Öztürel Orçun Orkan Özcan

Self-emergent property of shared conventions of socially joint multiple agents in an indeterministic environment can be captured by traditional language game models. Communication of the agents can eventually make their personal vocabulary converge into a shared universal lexicon of that specific environment. In this piece of work, a naming game model is implemented by adopting a reinforcement learning scheme. Specifically, pair selection and word selection strategies of agents are investigated in comparison with the traditional models. The effects of exploration rate, and reinforced reward/punishment rate on convergence trends of the society is investigated for both strategies. It is found that this methodology does not always result in faster convergence. However, it is discussed that utilization of reinforcement learning can introduce a psychologically plausible interpretation for the population based simulations of the emergence of socially shared vocabularies.

Keywords: *language games, emergence, artificial intelligence, machine learning, reinforcement learning*

1 Introduction

Semiotic dynamics is the domain of research which elaborately investigates emergence and evolution of linguistic conventions among a population by using computational multi-agent simulations. Previous studies of the field model the evolution of language over various distinct topologies and origination of the language (Baronchelli et al. 2007, Barrat et al. 2007). Most of these works only examine emergence of shared vocabularies (Baronchelli et al., 2005; Lenaerts et al., 2005; Steels, 1996) whereas recent researches also try to capture the emergence of syntactic structures (Vogt, 2005). Moreover, this domain formalizes several observations on emergent shared word-object maps and grammars within a society of rule-based acting agents. Current methodological approaches ground the discussion on rapidly changing social interactions among individuals. For the members of a society these emergent linguistic conventions will be urgent for interchanging their experiences and knowledge about the environment, which they are continuously acting on. Therefore, it is essential for the agents to converge on a shared semiotic system to survive in a shared environment.

Interpreting how these aforementioned linguistic conventions bootstrap is also a crucial step forward. Languages games, such as naming games (Baronchelli et al., 2006), discrimination games (Steels, 1996) and guessing games (Vogt, 2005) can provide suitable simulation models to make such interpretations. Among these, specifically the naming game literature investigates the emergence of a shared lexicon within a society. The traditional naming game is a special conventional language game, which emanates from late-Wittgensteinian language games (Wittgenstein, 1953). It investigates how vocabulary spreads within a multi-agent community, where each and every agent has a perceptual channel to perceive the surrounding objects. The aim of the agents is to converge on a shared vocabulary by just collaboratively communicating with each other on an iterative

Proceedings of the First Central European Conference in Linguistics for postgraduate Students (2012), 225–237.
ISBN 978-963-308-054-2

basis. In each episode of an interaction a speaker and a hearer is randomly chosen from the population. They both attend to the same object among a set of objects, and try to agree on a shared name for that specific object.

In parallel, this paper presents an exploratory research, in which a psychologically motivated artificial intelligence method, reinforcement learning (RL) (Sutton and Barto, 1998) with an eGreedy learning algorithm, is implemented in a minimal naming game (the term minimal denotes that there is only one object in the environment). We introduce a reinforcement learning algorithm employing a value function which is updated with rewards and punishments after every successful and unsuccessful interaction respectively. Two separate models are implemented, where agents adopt a reinforcement learning strategy either to choose a partner to communicate (agent selection strategy) or a word to transmit to the hearer party (word selection strategy), by using their previous experiences. Generic models in focus, do not use any strategy and work with random selection for partners and words. The study is exploratory in nature, because it aims to see whether the convergence trends are similar to generic models, when the agents are equipped with such a constraining preliminary assumption that they are biased on choosing their communicative partners or the words they exchange.

2 Related Work

Naming game is used as a generic baseline to investigate several properties of the naming game dynamics. Within the previous literature, to attain faster convergence and less memory usage in the game, some distinctive methodologies have been developed for word and pair selection strategies.

2.1 Word Selection

Baronchelli et al. (2005) built word selection strategies for faster convergence and less cognitive effort in naming. Namely, these are play-first, play-last and play-smart. In play-first strategy the agent selects the last word that was successful in a game, while in the play-last the agent utters the last word recorded in its inventory. As a combination of those two approaches, the play-smart strategy is put forth. In play-smart strategy, if the speaker was never successful in a game, it utters the last word recorded. Otherwise, if the speaker had at least one successful game, it utters the word of the last successful iteration.

Play-smart strategy performs much better than the other two strategies. It benefits play-last strategy at the outset since considerable consensus has not been formed between the individuals of the population. Agents utter the last word they record so that new word generation is prevented universally. After successive successful interactions, agents spread the accepted word providing play-first strategy to speed up the convergence trend.

In addition, a reinforcement learning technique is applied for word selection strategy by Lenaerts et al. (2005). This study will be revisited in Section 3.

2.2 Pair Selection

Again based on naming game, Baronchelli et al. (2006) further investigates the topological social structure of the multi-agent environment. A Barabasi-Albert (BA) network is adopted in their study, in which there are fully-connected central nodes and new nodes are added to the central nodes with m = 2 links. In that way, k = 2m average connectedness is assured throughout the growth of the network.

For such a complexity, it becomes important which role is assigned first to the participating parties of a round of communicative interaction. The hearer-first case assigns a low-connected node from majority. Then the speaker happens to be a highly-connected node with high probability. The reverse is true for the speaker-first case and one more case is generated in which roles of being a speaker and a hearer are assigned with equal probability to the edges of a randomly selected link.

The hearer-first case achieves faster convergence since the speaker has a higher probability of being a hub (highly-connected node) as it is selected after the hearer. Hubs provide faster spread of consensus by keeping the number of different words low. On the other hand, if speakers were selected first then more words will be in circulation within the population since they will more likely to be a low-connected node. Therefore, pair selection significantly effects convergence trends in non-trivial population topologies. In accordance, complex networks like BA can be used as plausible real world models.

Nowak et al. (1999) presents a mathematical framework to study the performance of different learning mechanisms in an evolving population. Three distinct models of learning, namely parental learning, role model learning and random learning are employed in their model. For parental learning, they assume that successful communicators in the environment have more offsprings whereas in the role model learning they have more imitators. On the other hand, communicating individuals are simply randomly selected in the random learning. Throughout the generations newly created individuals go through a learning phase with one of these learning models. Consequently, members of the population gradually gain a shared vocabulary through generations. Within a well-defined mathematical framework, it is shown that parental and role-model learning have a significant success over random learning.

Similarly, evolutionary properties of corporate culture over a naming game model are examined by Pan Yang and Jian-Yong (2008). During the interaction between the staff, managers who are doing most of the communication affect the transmission of the corporate culture, namely event-behavior pairs. This outcome overlays the importance of the pair-selection for convergence of the event-behavior inventories.

3 Methodology

3.1 Generic Naming Game

The traditional naming game models a population of n agents $A = \{a_1, a_2, \ldots, a_n\}$, where each and every agent can equally perceive and be knowledgeable about the environment, which contains a set of objects $O = \{o_1, o_2, \ldots, o_m\}$. Agents have their own private lexicon, which defines an inventory of a set of word-object pairs, such as a_j can have an inventory $I_j = \{\{w_i, o_k\}, ...\}$ at a given time throughout the game. Every agent starts the game with an empty inventory $I = \{\}$. Iteratively in each episode of communication a

random speaker a_x and hearer a_y is chosen for $x \neq y$ and $x, y \leq n$. Both agents attend to an object o_k for $k \leq m$, and they try to agree on a name, w_i. The naming game ends after an iteration when all the agents converge on to a shared inventory, where $I_1 = I_2 = \ldots = I_n$. Minimal naming game focuses on an environmental setup where there is only one object. Therefore, a personal inventory of an agent can be reduced to a set of words, such as $I_j = \{ \sigma_1, \sigma_2, \ldots, \sigma_q \}$. Personal inventories only consist of the words, which can be used to name that specific object by an agent. Accordingly, the algorithm for an episode of communication among two randomly selected agents within the minimal naming game can be outlined as follows:

1. Randomly choose one speaker and one hearer from the population.
2. Speaker transmits a name to the hearer.
   (a) If speakers inventory is empty then it invents a new word and transmits it.
   (b) If there is one name in speakers inventory then it transmits that name.
   (c) If there is more than one name in speakers inventory then it randomly transmits one of them.
3. Hearer processes the uttered name.
   (a) If the uttered name is in hearers inventory then communication is a success.
   (b) If the uttered name is not present in hearers inventory then communication is a failure.
4. Both parties make final modifications on their inventory.
   (a) If success then both parties delete all the words from their inventories except the one, which is agreed on (the one which is transmitted by the speaker).
   (b) If failure then only hearer updates its inventory by adding the uttered name to its inventory.

3.2 Pair Selection Algorithm

Only the first step of the generic algorithm for the minimal naming game is modified to implement a pair selection strategy with reinforcement learning. In particular, in the generic algorithm the speaker and the hearer is chosen randomly among the community, whereas in the application of pair selection strategy first the speaker is chosen randomly and then that specific speaker chooses its hearer counterpart. Reinforcement learning technique is employed in a basic level to implement this idea, as the success of previous communications can be stored and used for pair selection within future iterations.

To attain this, each and every agent holds a value function, which is as big as the number of agents in the community except itself. For instance, agent a_x will have a value function $V_x = \{v_1, \ldots, v_{x-1}, v_{x+1}, \ldots, v_n\}$, which will make a_x distinctively remember how successfully it has communicated with the other agents. An empty value function is assigned to each agent, while the game is set to run. After each episode of communication only the speaker collects rewards and updates its value function accordingly. Briefly, when a speaker a_x communicates with hearer a_y, the value of v_y in V_x is updated.

An application of the eGreedy algorithm is used for the speaker to decide the most beneficial hearer (Sutton and Barto, 1998). In other words, agents use their value function to pick out the hearer with the highest value to communicate in the upcoming episode. Moreover, according to the exploration rate an exploratory move is taken by choosing a random hearer throughout the communicative iterations. Speaker makes an

exploratory move according to the given exploratory rate. The exploratory rate determines the probability of choosing the hearer randomly instead of choosing a specific hearer. That is, for a given exploratory rate of 0.2 the speakers will select a random hearer from the population with 0.2 probability. Essentially, the population will be discovered gradually with respect to the exploration rate.

3.3 Word Selection Algorithm

Lenaerts et al. (2005) completed a similar study to investigate the emergence of word-meaning mappings and the algorithm below will be a small scale replica of it. The underlying idea is to examine the emergence of a shared vocabulary using reinforcement learning with a word selection strategy.

Similar to the implementation of pair selection strategy, for the word selection strategy only the second step of the generic algorithm is modified. Speakers and hearers are chosen randomly from the community, similar to the generic algorithm. In fact, for the implementation of word selection strategy a static set of words are used, to make the model a suitable Markov decision process model. Consequently, when a speaker needs to transmit a name, if the inventory is empty a new word is selected from the universal static word set $w = \{w_1, \ldots, w_n\}$. Therefore, speakers cannot invent new words from scratch, as in the case of generic and pair selecting algorithms. However, if they do not have any names in their inventory to name an object, they just select a word from this aforementioned static set of words. If they have only one word for an object in their inventory, they just transmit that word without using any decision algorithms. Within the word selection strategy reinforcement learning is just used when there is more than one word in speakers inventory. In that case, the speaker tries to transmit the most beneficial word (the word that helped to attain more successful communication) by using reinforcement learning.

The value function of each agent holds values which indicate how beneficial a word is regarding that agent's previous experiences. Therefore, value function for an agent a_x can be represented as $V_x = \{v_1, v_2, \ldots, v_n\}$, where n is the size of the static word set. Similar to the pair selection implementation, the eGreedy algorithm is used to explore the word set. At the beginning of each episode of communication an agent picks out the highest valued word to transmit from the value function, if it will not going to conduct an exploratory move. After an interaction with the hearer only the speaker collects rewards. Specifically, when a word w_i is uttered by the speaker a_x, the value of v_i in V_x is updated by using the reinforced reward or punishment depending on the success of the interaction.

3.4 Experiments

Regarding the outlined algorithms three different experiments are conducted to compare the convergence trends of generic, pair selecting and word selecting algorithms. The benchmarks N_w (total number of words generated), N_d (total number of distinct words generated) and S (success rate, the probability of being successful in an iteration), which are used to provide a concise comparison with the previous works of the literature are borrowed from Baronchelli et al. (2005). Conditions, which are detailed below are tested over a simulation framework, which is implemented in Python 2.7.

- Generic, pair and word selection algorithms are compared according to the N_w, N_d and S, for 50 agents with an eGreedy exploration rate of 0.2, where both the reward and the punishment values are fixed to be 0.05.
- The effects of varying the rate of exploration examined on convergence trends of pair and word selection models for 50 agents, where both the reward and the punishment values are fixed to be 0.05.
- The effects of reward/punishment rates on convergence trends of pair and word selection models are examined for 50 agents, where the eGreedy exploration rate is set to be 0.2.

4 Results and Discussion

Behavior of the population for the previously mentioned naming game algorithms can be interrelated in terms of their convergence trends (namely, how fast the population reaches to a final state), total number of words created in the population at a given time during the simulation (which can also be referred as the amount of memory used among agents) and number of distinct words created by the population. When the performance of the generic minimal naming game algorithm is regarded as a baseline, the application of reinforcement learning on pair and word selection strategies does not provide better results in terms of faster convergence. From Figure 1, Figure 2 and Figure 3 it can be interpreted that generic algorithm outperforms the other two modified ones. However, it can also be stated that given the right conditions in terms of simulation variables, populations which adopt both pair and word selection algorithms can also converge on a shared lexicon. Moreover, from above mentioned figures it can also be observed that memory complexity nearly overlaps for the generic and pair selection algorithms, whereas it is comparatively larger for the word selection algorithm. In fact, the memory selection algorithm forces agents to discover the state space with exploratory name selections even when the population starts to form a consensus. Therefore, word selection models have greater memory complexity and slowest performance.

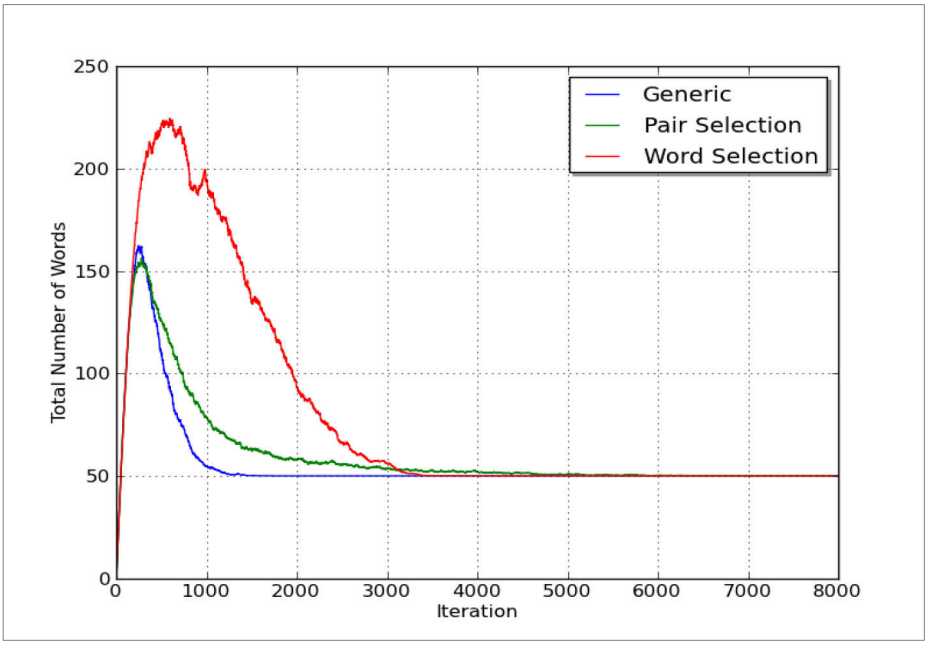

Figure 1. Total Number of Words (Nw) vs. Iterations for generic, pair and word selection algorithms, where number of agents is 50, exploration rate is 0.2, reward and punishment rates are 0.02. Results are averaged for 30 runs.

Despite their latency in convergence, an examination of N_w and N_d values of the modified algorithms in-correlation can be valuable to observe the grouping structures within the population. In depth, in a given time among the populations having the same total number of words, greater number of distinct words may be an indication for the grouping within the society. Precisely, lower rate of N_w/N_d denotes the generation of groups, which agree on distinct words among themselves in the population. The highest grouping rate can be observed through the iterations, where society reaches the peak values of N_w and N_d, in Figure 2 and Figure 3. For the generic algorithm N_w/N_d is 2.81, whereas this ratio is slightly changing around 1.90 for both the pair and word selecting models. The selection strategies directly effect the selection mechanisms for agents of the modified algorithms. Therefore, successfully communicating agents come up with a group in pair selection model. Similarly, words which lead to successful communication will also make the society partition into groups according to distinct words which are favored by distinct groups. The participants of each group form agreements within their groups. This has direct implications on slower convergence for modified algorithms, as the group based conventions needs to be globally spread to attain a universal agreement.

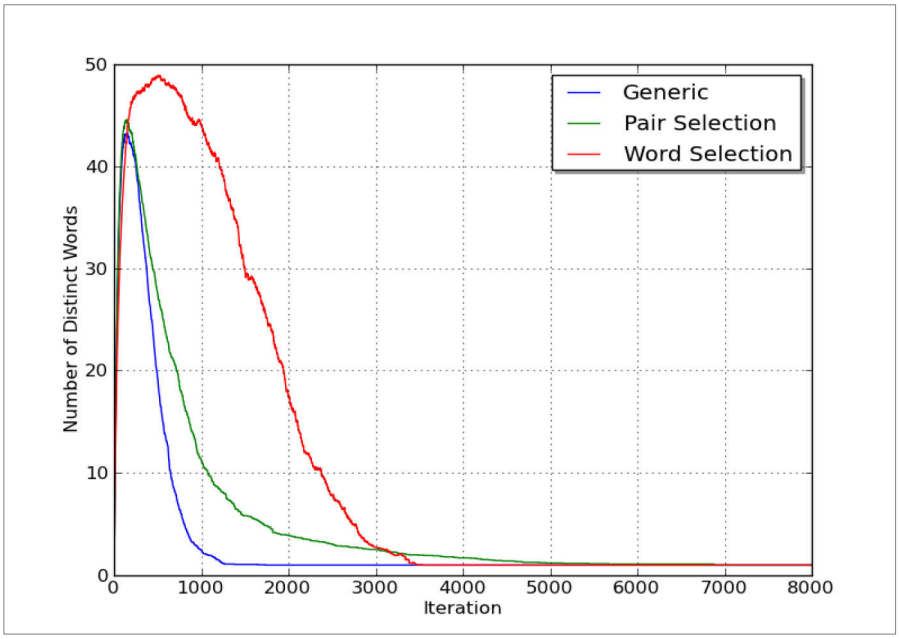

Figure 2. Number of Distinct Words (Nd) vs. Iterations for generic, pair and word selection algorithms, where number of agents is 50, exploration rate is 0.2, reward and punishment rates are 0.02. Results are averaged for 30 runs.

Subsequently, the rate of spread for agreements is also dependent on the exploration rate. For the pair selection algorithm, it is expected that an increase in exploration rate will yield a faster convergence. This is because the agents, who adopt the role of a speaker within an iteration, transmit their vocabulary to a higher proportion of the society for greater rate of exploration. In consequence, when the exploration rate approaches to 1, the pair selection algorithm will behave as a generic algorithm. This so-called direct relation between exploration rate and faster convergence for the pair selection algorithm can be observed on Figure 4. For the word selection algorithm it is expected that an increase in the exploration rate will delay the convergence of the population on a shared vocabulary. This is because, an increase in the exploration rate will force the word selecting agents to transmit varying words from the static word set. In that case, total number of distinct words can increase drastically for higher exploration rates in a word selection algorithm simulation. Significantly, as it can be observed from Figure 5 this assertion does hold for word selection, since time of convergence gradually increases proportionally with the probability of choosing a random word to explore the state space.

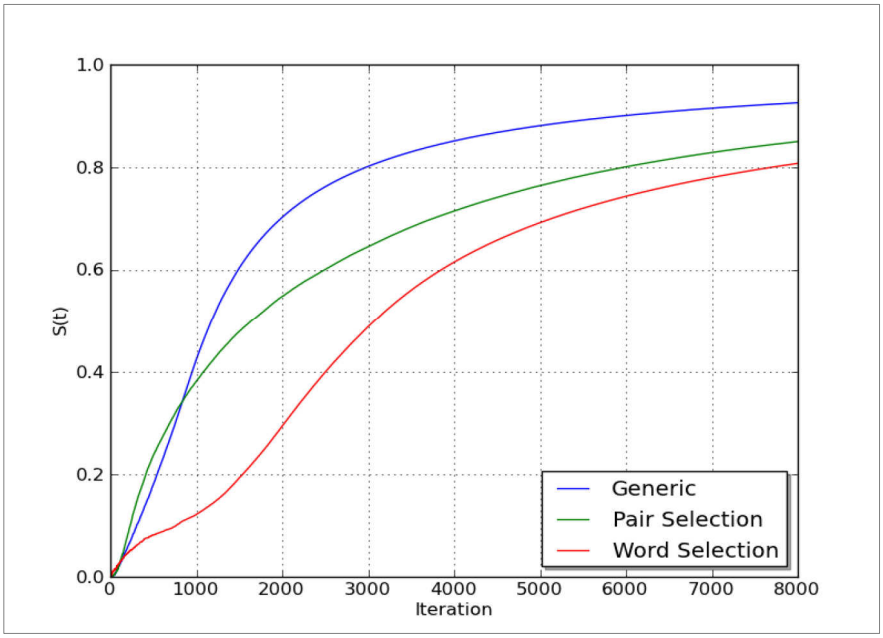

Figure 3. Success Rate (S) vs. Iterations for generic, pair and word selection algorithms, where number of agents is 50, exploration rate is 0.2, reward and punishment rates are 0.02. Results are averaged for 30 runs.

The amount of reinforced reward and punishment rates for the pair selection algorithm does not effect the convergence trend as it can be seen from Figure 6. The amount of reward that the speaker gains does not play a role, because in any case previously granted reward for that hearer will determine the future selections for the speaker. On the other hand, for the word selecting algorithm an increase in reward/punishment ratio will decrease the time needed for convergence. Selection of any highly rewarded word will dominate the value function of an agent. Hence, as it can be seen on Figure 7. convergence comes earlier if a word gets a higher reward after successful interactions, than the punishment it gets after unsuccessful ones. However, equal reward and punishment values (reward/punishment rate = 1) will result in slower convergence as their effects on value functions can cancel each other out at the beginning of consensus formation for the word selection algorithm.

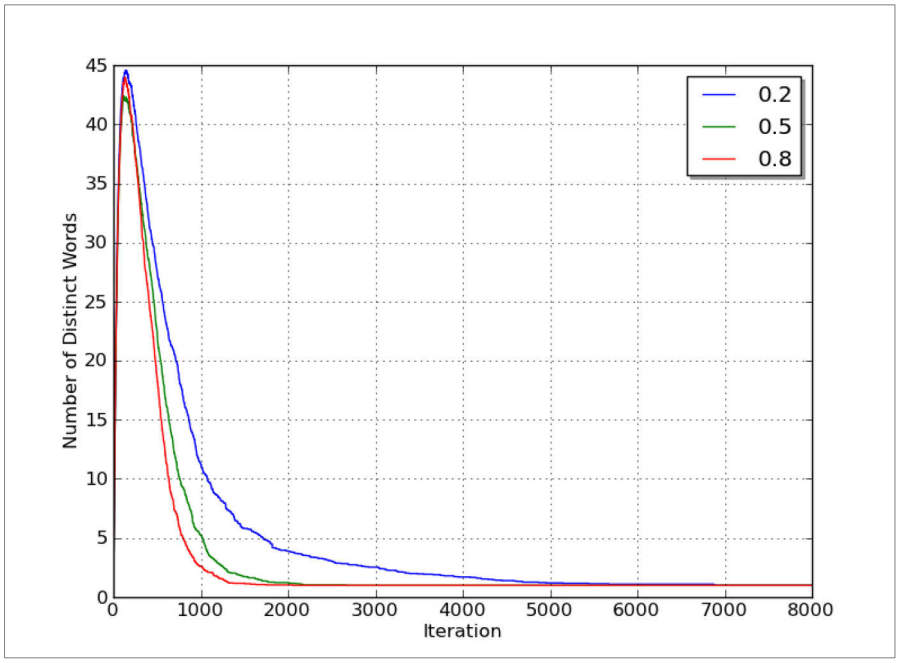

Figure 4. Number of Distinct Words (Nd) vs. Iterations for pair selection algorithm for varying exploration rates 0.2, 0.5 and 0.8, where number of agents is 50, reward and punishment rates are 0.02. Results are averaged for 30 runs.

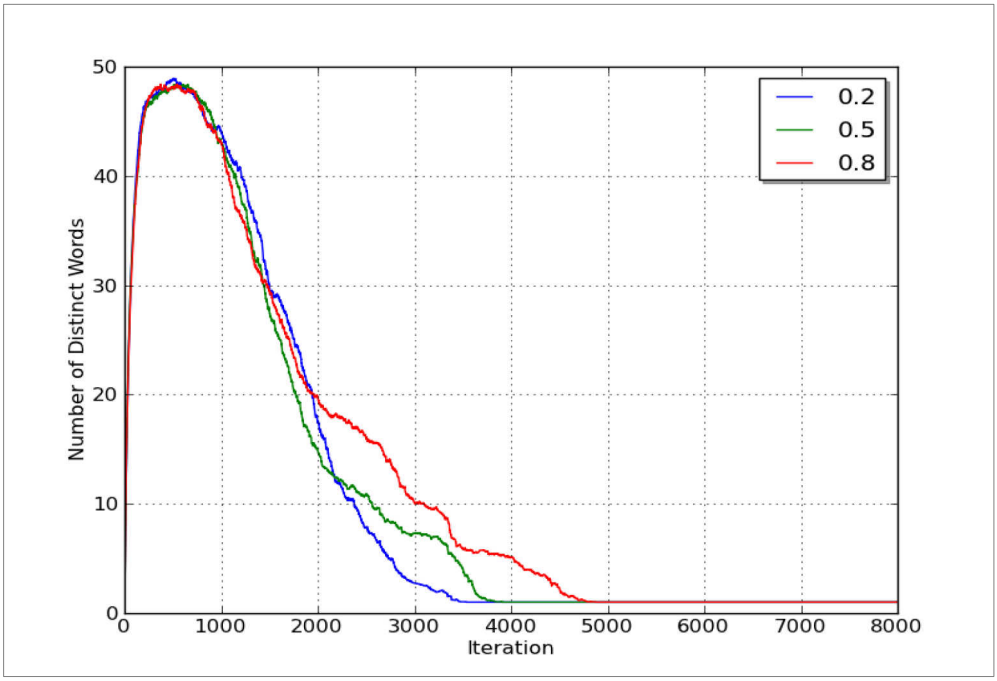

Figure 5. Number of Distinct Words (Nd) vs. Iterations for word selection algorithm for varying exploration rates 0.2, 0.5 and 0.8, where number of agents is 50, reward and punishment rates are 0.02. Results are averaged for 30 runs.

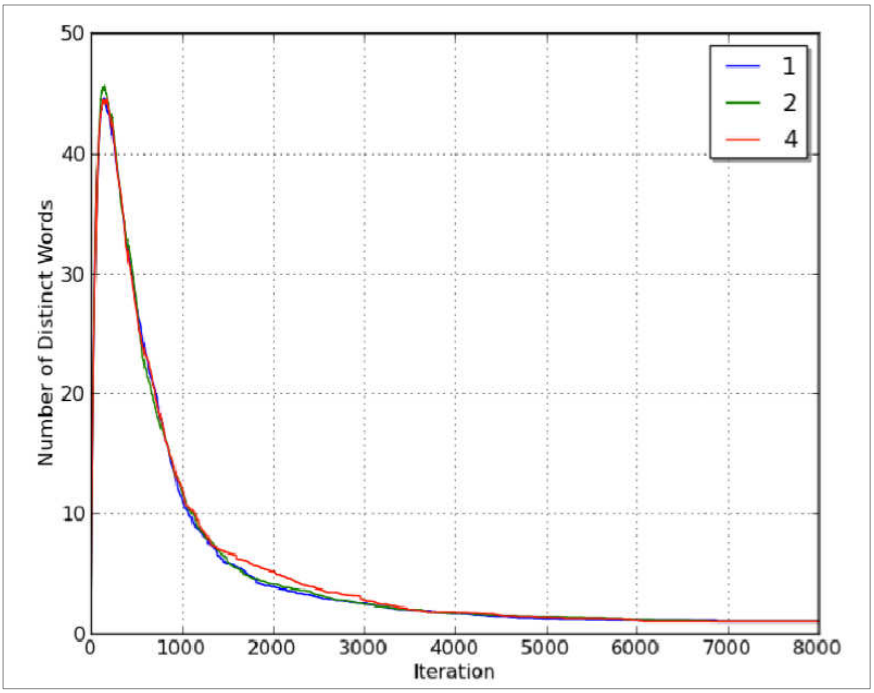

Figure 6. Number of Distinct Words (Nd) vs. Iterations for pair selection algorithm for varying reward/punishment rates 1,2,4 and 8, where number of agents is 50, reward and punishment rates are 0.02. Results are averaged for 30 runs.

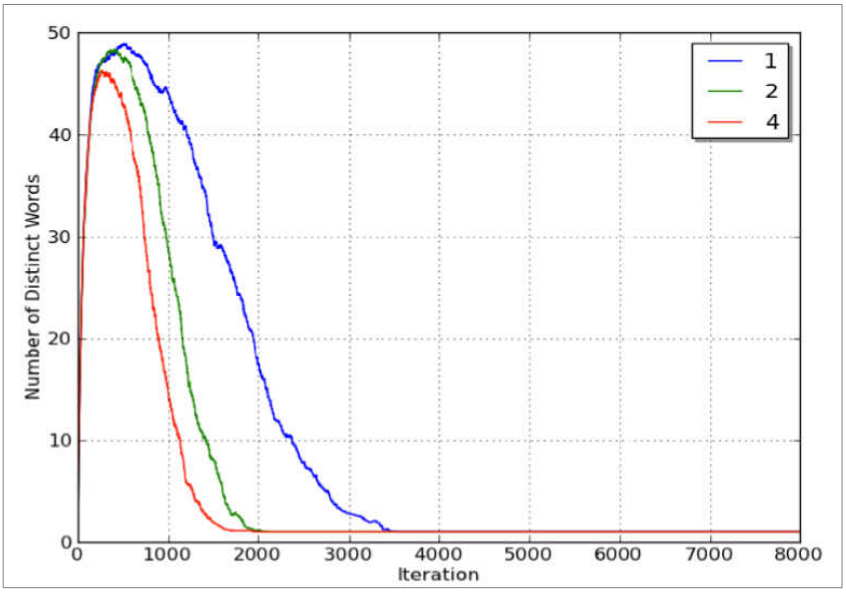

Figure 7. Number of Distinct Words (Nd) vs. Iterations for word selection algorithm for varying reward/punishment rates 1,2,4 and 8, where number of agents is 50, reward and punishment rates are 0.02. Results are averaged for 30 runs.

5 Conclusion

In this piece of work, the tradition naming game model is extended with reinforcement learning to study pair and word selection strategies. Briefly, modified reinforcement learning models of the naming game can also bootstrap shared vocabulary similar to the traditional model, if convenient simulation variables are provided. Convergence trends of the traditional naming game model and modified models are compared and contrasted. Specifically, the effects of different exploration rates, reward and punishment values and memory complexities are comparatively investigated. It is concluded that the convergence trends of the modified models behave similarly to the traditional models however the modified models need more time to converge.

Reinforcement learning algorithm applications for pair selection and word selection strategies are employed to boost communicational convergence of the agents. It is presented that reinforcement learning which is a psychologically motivated artificial intelligence approach could also be devised to study the emergence of linguistic conventions. Within such computational models social structure and language co-evolve. The modified selection strategy algorithms that we have implemented support the co-evolution of vocabulary and structure of the society. For future research, different topological settings can be applied on the network of the agents in order to study their effects on semiotic dynamics.

References

Baronchelli, Andrea, Luca Dall'Asta, Alain Barrat and Vittorio Loreto. 2005. Strategies for fast convergence in semiotic dynamics. Arxiv preprint physics/0511201.

Baronchelli, Andrea, Vittorio Loreto, Luca Dall'Asta and Alain Barrat. 2006. Bootstrapping communication in language games: Strategy, topology and all that. *Proceedings of the 6th International Conference on the Evolution of Language*. 2006: 11-18.

Baronchelli, Andrea, Maddalena Felici, Vittorio Loreto, Emanuele Caglioti and Luc Steels. 2006. Sharp transition towards shared vocabularies in multi-agent systems. *Journal of Statistical Mechanics: Theory and Experiment*. Vol. 2006: P06014.

Baronchelli, Andrea, Luca Dall'Asta, Alain Barrat and Vittorio Loreto. 2007. The role of topology on thedynamics of the Naming Game. *The European Physical 10Journal-Special Topics*. Vol. 143: 233-235.

Barrat, Alain, Andrea Baronchelli, Luca Dall'Asta and Vittorio Loreto. 2007. Agreement dynamics on interaction networks with diverse topologies. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. Vol. 17: 026111.

Gong, Tao, Jinyun Ke, James W. Minett and William S. Wang. 2004. A computational framework to simulate the coevolution of language and social structure. *Artificial Life IX: Proceedings of the 9th International Conference on the Simulation and Synthesis of Living Systems*. 158-164.

Lenaerts, Tom, Bart Jansen, Karl Tuyls and Bart De Vylder. 2005. The evolutionary language game: An orthogonal approach. *Journal of Theoretical Biology*. Vol. 235: 566-582.

Nowak, Martin A., Joshua B. Plotkin and David C. Krakauer. 1999. The evolutionary language game. *Journal of Theoretical Biology*. Vol. 200: 147-162.

Pan, Xiang-dong, Jian-mei Yang and Feng Jian-yong. 2008. Research on the Evolution of Corporate Culture Based on Naming Game. *Computing, Communication, Control, and Management, CCCM'08. ISECS International Colloquium*.

Steels, Luc. 1996. Perceptually grounded meaning creation. *Proceedings of the International Conference on Multi-Agent Systems*. 338-344.

Sutton, Richard S. and Andrew G. Barto. 1998. *Reinforcement Learning: An Introduction*. MIT Press: Cambridge, MA.

Vogt, Paul. 2005. The emergence of compositional structures in perceptually grounded language games. *Artificial Intelligence*. Vol. 167: 206-242.
Wittgenstein, Ludwig. (1953). *Philosophical Investigations*. Oxford: Blackwell.