

# School of Informatics



## Informatics Project Proposal Emergence of Numerals in Multi-agent Autonomous Communication System

**B131025**  
**April 2019**

### **Abstract**

This project aims to develop a new computing simulation method for the emergence of numeral system based on multi-agent autonomous communication system following deep reinforcement learning methodology. With this new method, we do not need to pre-assign any linguistic knowledge to computer models any more. Instead, we can facilitate computational agents to invent their numeral system all from scratch. Thus, this project can help us to evaluate not only the assumptions about the emergence of numeral systems in natural language but the importance of certain factors in this procedure. From the perspective of artificial intelligence, this project may also could facilitate a new technique to address the long-standing natural language understanding problem.

Date: Thursday 11<sup>th</sup> April, 2019

**Tutor:** Prof. Stuart Anderson

**Supervisor:** Dr. Ivan Titov

# Contents

## 1 Motivation

How do natural language emerge and evolve? This is a critical question to the field of evolutionary linguistics [?]. As linguistic behaviors are typically not preserved during pre-history [?], it is necessary to develop some quantitative methods to overcome this kind of time limit [?]. Among all those quantitative methods, computer simulation <sup>1</sup> has now become increasingly important since it was first introduced by [?]. During the last two decades, computer simulation methods attracted an increasing number of attention (e.g. [?, ?, ?, ?, ?, ?]). However, in all these previous works, the basic linguistic elements (meanings to communicate about, a signalling channel to employ) are provided from the outset and thus the learning product of models are thus insufficient to test hypotheses about the emergence of these basic linguistic elements, such as the emergence of numerals, pronouns and etc.

With the recent rapid development of deep reinforcement learning (DRL), e.g. [?, ?], computers are now available to master a variety of complicated cognitive activities. Thus, several papers in grounded language learning (GLL), e.g. [?, ?], apply DRL to language game [?] and signalling game [?] to facilitate agents to understand natural language or invent a communication protocol that may share some identical characteristics of natural language. Under the assumption that agents should also learn and understand language by interacting with environments and others (as humans do), GLL aims to address the natural language understanding problem with agents that can ground symbols/utterances to their experiences [?].

With these promising progresses in GLL, this project aims to propose a new simulation method for the emergence of specific linguistic phenomenon in multi-agent autonomous communication system trained by DRL techniques to solve some specifically designed language games that can imply linguistics hypotheses or theories. However, as language is a too huge a topic, we limit our project in numeral systems for the following reasons: i) numeral systems are relatively simple and self-contained[?]; ii) concepts related to numeral systems are more straightforward to model with numeric representations; iii) functions of emergent numeral systems can be formalised and verified more reliably in simulation. Moreover, with our new simulation method, we can not only further evaluate these hypotheses/theories about the emergence of numeral systems but shape the numeral systems <sup>2</sup> invented by computational agents to be similar as those in natural language and thus facilitate agents to understand natural language better (with assumptions from contact linguistics).

In the following sections, we would first introduce background knowledge from evolutionary linguistics and GLL respectively. Then, in Section ?? we would introduce the high-level architecture and also the proposed language game. Finally, we would analyse the potential risks in this project and give a preliminary timetable of this project.

## 2 Background

In this section, we would introduce some related works from evolutionary linguistics community and GLL community respectively, as they all share a similar argument and assumption to our

---

<sup>1</sup>Unless indicated otherwise, computer simulation in this paper refers to the computer simulation methods employed in evolutionary linguistics.

<sup>2</sup>Formally, it should be a kind of communication protocol.

work.

## 2.1 Computer Simulation Methods in Evolutionary Linguistics

The fact that linguistics behaviours can hardly be recorded during history [?] makes computer simulation an “operational” empirical method to verify the hypotheses or theories about the emergence and evolution of linguistics [?]. In evolutionary linguistics community, it usually consists of 3 steps to transform linguistics theories into computing models [?]: i) set up environments where agents can action and communicate; ii) pre-define basic linguistic knowledge to models; iii) analyze the pre-define quantities. We would further discuss the first and second step as they share similar characteristics with our project.

With different simulation objectives, the environments can be generally classified into the following 3 types [?]:

- *Iterated learning*: This was first introduced by [?] in which all the individual agents form a chain, learn the underlying linguistics mechanism by observing the precursor and generate behaviours for the next. Basically, this game simulates the cultural transition from generation to generation.
- *Language game*: This was first introduced by [?] in which all the individual agents play specific games and develop some shared communication protocols by completing specific tasks. There exist lots of derivations, such as naming game [?], category game [?] and etc.
- *Genetic evolution*: This was first introduced by [?] in which all individual agents have their own genomes encoding idiolects and specific linguistic elements. Based on the assumption of natural selection [?], agents can get higher and higher values with random mutation in reproduction.

In our project, we also follow the framework of **language game** like other works in GLL [?]. However, different from the existing works, we do not need to pre-define any linguistic elements which are introduced as follow.

Most of the computing models from the evolutionary linguistics community are to learn bi-directional meaning–utterance mappings (M–U mappings) [?]. Thus, different forms of pre-defined linguistics knowledge may result in various kinds of model. Generally speaking, these models can be classified into two categories: i) lexical models; ii) syntactic and grammatical models. In lexical models, such as [?, ?, ?], the main focus is whether a common lexicon can form during the autonomous communication between agents. On the other hand, in syntactic and grammatical models, agents aim to map semantic items to either structured or unstructured utterances, like [?, ?]. No matter how this mapping function is learnt, e.g. by neural network models [?] or by mathematical equations [?, ?], the most basic elements of linguistics, such as symbols and rules of generating phrases, are all pre-defined between agents operating on these basic elements.

## 2.2 Grounded Language Learning

Natural language understanding is a long-standing and challenging problem in natural language processing and artificial intelligence. With the development of deep learning, most of the current state-of-the-art methods are based on deep learning models trained on the massive static textual

corpus. We humans, however, learn and understand languages by interacting with others and the real world, which is a different paradigm from statistical learning. Therefore, GLL argues that agents also need to learn the language by doing so. Basically, in grounded language learning, agents learn to communicate with symbols from a fixed vocabulary by completing tasks in a specific environment, the procedures of which are called games.

The games mentioned above derive from the language games first introduced by [?] and also the signaling games proposed by [?]. Based on the number of participated agents, we briefly classify such games into 2 categories : i) single-agent game such as [?, ?, ?], in which researches mainly explore how could an agent learn natural language from humans; ii) multi-agents games like [?, ?], in which researches mainly focus on how language emerge and develop during the autonomous communication between agent population.

As for the multi-agent games themselves, they are mainly designed to explore the conditions for the emergence and development of natural-language-like communication protocols, in which agents are quite like the primitive human who create natural languages during production, or to say, survival games. To be specific, [?] observes that demonstrative pronouns emerge when there exist more than two agents in their game. Meanwhile, it is shown in [?] that collaboration between different agents is a crucial motivation for the development of cooperative communication, especially when each agent holds partial information about the environment.

However, instead of aiming to make DRL models generalise better, this project targets at proposing a systematic methodology of facilitating the emergence of numeral systems in a multi-agent autonomous communication system.

### 3 Architecture and Proposed Language Games

Although all the previous work on computer simulation in evolutionary linguistics gained promising results, one congenital deficiency of these method is that they lack the capability to simulate the emergence of basic elements of numeral systems, such as cardinal numerals and ordinal numerals. Take the work of [?] for example, the semantic primitives is set up at the beginning of simulation as well as the basic cognitive operations. With the success of DRL in GLL, we however propose a new framework that can simulate these basic elements in numeral systems in order to further relax the restrictions on the learning procedure of computing models, which makes it more consistent with the learning procedure of humans.

In this section, we would first give an overall illustration of the architecture and components of our simulation system. Then, we would give a specific language game that can simulate the emergence of cardinal numerals. Further, we would propose two more games to simulate the emergence of ordinal numerals and numeral grammars and analyse the main challenges in such simulations.

#### 3.1 Architecture of Proposed Simulation Method

We keep to the traditional framework of transforming linguistic theories into simulation mechanism expect that we do not need to set up any artificial language or prior linguistic knowledge to computational agents. Thus, the main steps in our framework are: i) design a language game that implies linguistic hypotheses or theories that need to be verified; ii) implement this language game in a virtual environment; iii) let agents interact with and adapt to the environment<sup>3</sup>,

---

<sup>3</sup>This step is achieved with DRL technique.

during which the communication is necessary and autonomous; iv) analyse the experiment results and get the verification results. With respect to different steps, we propose that there are 3 main components of system designed by our method:

1. **Environment:** Virtual environments in our systems are all multi-agent systems, in which agents need to cooperate/compete with each other to complete tasks that are actually some specifically designed language games. By defining observations<sup>4</sup>, action spaces and reward functions for every agent, we are able to build up specific language games and further imply linguistic hypotheses or theories by carefully designing the mechanisms of these language games. As our goal is to study linguistic phenomena, for all environments, we should always establish communication tunnels for agents and make communication necessary for agents to complete their tasks.
2. **Agents:** As we do not want expose any prior linguistic knowledge to agent, our multi-agent system are all trained by multi-agent reinforcement learning algorithms like MADDPG[?]. Also, as communication is just one kind of actions in virtual environments, the communication behaviours of agents are all spontaneous and autonomous. Thus, based on this assumption, we can verify the hypotheses/theories about the emergence of basic linguistic elements by analysing the produced symbol system among the multi-agent system.
3. **Vocabulary:** As we use a discrete vocabulary, i.e. words in natural language are all separate symbols, we force agents to communicate with a fixed-sized discrete vocabulary in order to simulate our natural language. Considering that the range of real numbers are unlimited, we argue that agents are able to communicate as much information as they want with a single-digit real number and thus it is necessary to limit their communication behaviour to use a fixed-sized discrete vocabulary. Moreover, inspired by symbol grounding proposed by [?], we further argue that agents could ground symbols to their observations in virtual environment after learning.

### 3.2 Game for Cardinal Numeral: Food-Gathering Bandit Game

As cardinal numerals are the very basic ingredients in numeral systems, we thus first propose the so called “Food-Gathering Bandit Game” (FGBG) to simulate the emergence of cardinal numerals. Following the architecture demonstrated in subsection ??, we design this game by specifying that: i) components of this game; ii) objective of agents; iii) game procedure and restrictions. A brief diagram of our proposed game is given in Figure ?? as follow and more details are described in the following paragraphs.

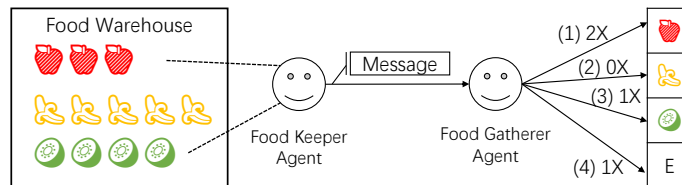


Figure 1: Diagram of proposed Food-Gathering Bandit Game.

<sup>4</sup>Terminology from reinforcement learning community. More information can be found in [?].

In order to better describe our game configuration, we first introduce the following components of FGBG:

- **Food Warehouse:** There is always 3 specific kinds of food in food warehouse whose quantity is initialised randomly from episode to episode. Besides, the quantity is represented as a single integer  $n \in \{1, 2, \dots, 5\}$ .
- **Speaking Agent:** This is the one that can check status of warehouse and send a message to the other agent. However, its action space is limited such that it can not push the bandit.
- **Listening Agent:** This is the one that first receives message from speaking agent and then can push the bandit according to the message. However, its observation is limited such that it can not observe the status of food warehouse.
- **4-armed Bandit:** There are only 44 actions that are supported by the 4-armed bandit, i.e. get an apple, get a banana, get a kiwi or end the episode immediately.
- **Message Channel:** Messages consist of some discrete symbols that are initially meaningless. However, we assume that different reward configuration would lead to different preference of agents and we thus would further illustrate this in the following paragraph.

The **Objective** of FGBG is to fulfill the food warehouse with exactly 5 apples, 5 bananas and 5 kiwis. In the beginning of every episode, speaking agent would first check the status of food warehouse and then send a message to the listening agent. After receiving the message, the listening agent would push the bandits. And, once it pushes the bottom bandit, the game ends. If the listening agent pushes the bottom bandit exactly as the numbers needed, i.e. 5 minus numbers of each fruit, both agents would get reward +100, otherwise, they would get -100. Take the situation in Figure ?? for example. As there are 3 apples, 5 bananas and 4 kiwis in the food warehouse after initialisation, the desired behaviour of listening agent is to *push the 1st bandit twice, push the 3rd bandit once and then push the end bandit*, after receiving the message from speaking agent. If so, both agents would get reward +100 otherwise -100.

With the assumption that all the information need to communicated between two agents is the number of each fruit in food warehouse, we argue that all the symbols combinations appear as messages should all be combinations of numerals. From a functional perspective, numerals emerged in this game are to describe how many times an action should be executed. Take number 3 for example, its function is to compress a string consists of 3 same symbols into 1 tokens, e.g.

$$3(apple) = apple \ apple \ apple$$

.

Ideally, agents could assign every symbol to a specific number. However, it is also possible that meanings of symbols in their communication are not atomic, i.e. they may use successive several symbols to represent a single number like the 8-digits integers in digital computers.

## Future Works

Based on FGBG, we could also alter the components to make the simulation be able to express/reflect more linguistic phenomena related to cardinal numerals. For instance, with a penalisation mechanism on the length of messages, we could further verify the hypotheses about the balance between efficiency and ambiguity of language.

Unlike cardinal numerals and ordinal numerals, the emergence of numeral grammars are not self-supported but rely on other basic elements in numeral systems. Thus, we propose the following hypotheses that can facilitate the emergence of numeral grammars and, more importantly, can be implied in the proposed simulation framework.

1. Dynamic changing objective and dialogue: concepts related to arithmetic operations in numeral grammars become necessary when agents need dialogues to capture the dynamic changing objective during completing tasks. For example, in FGBG, once the speaker agent realised that the status of food warehouse changed, it can directly tell the listener how many more apples are needed, which is a more efficient communication protocol.
2. Update the existing cardinal numerals: once agents realise the boundness of the existing cardinal numeral systems, they could take n-based positional numeral systems as an update the existing communication protocol (i.e. numeral systems). Assuming that vocabulary could not be enlarged during the simulation, this is the only way for agents to enhance the representation capability of the existing numeral systems.

## 4 Evaluation

Key problems here:

- How to evaluate the effectiveness of proposed method? (Rewards gained during the game.)
- How to evaluate the emergent numeral system? How to decipher the communication protocol?

## 5 Expected Outcomes

As an interdisciplinary project, the expected outcomes of our project are various.

First and foremost, the main contribution of this project is a new computer simulation framework for emergence and evolution of numeral systems as well as other topics in evolutionary linguistics. Our work also fills the gaps between existing works in evolutionary linguistics and hypotheses/theories about basic elements of language. Moreover, we provide a more powerful tool to verify the hypotheses and theories in evolutionary linguistics.

Meanwhile, as we follow the reinforcement learning framework, we would also contribute techniques that estimate the gradients of discrete symbols in vocabulary during the end to end multi-agent training procedure.

## 6 Research Plan, Milestones and Deliverables

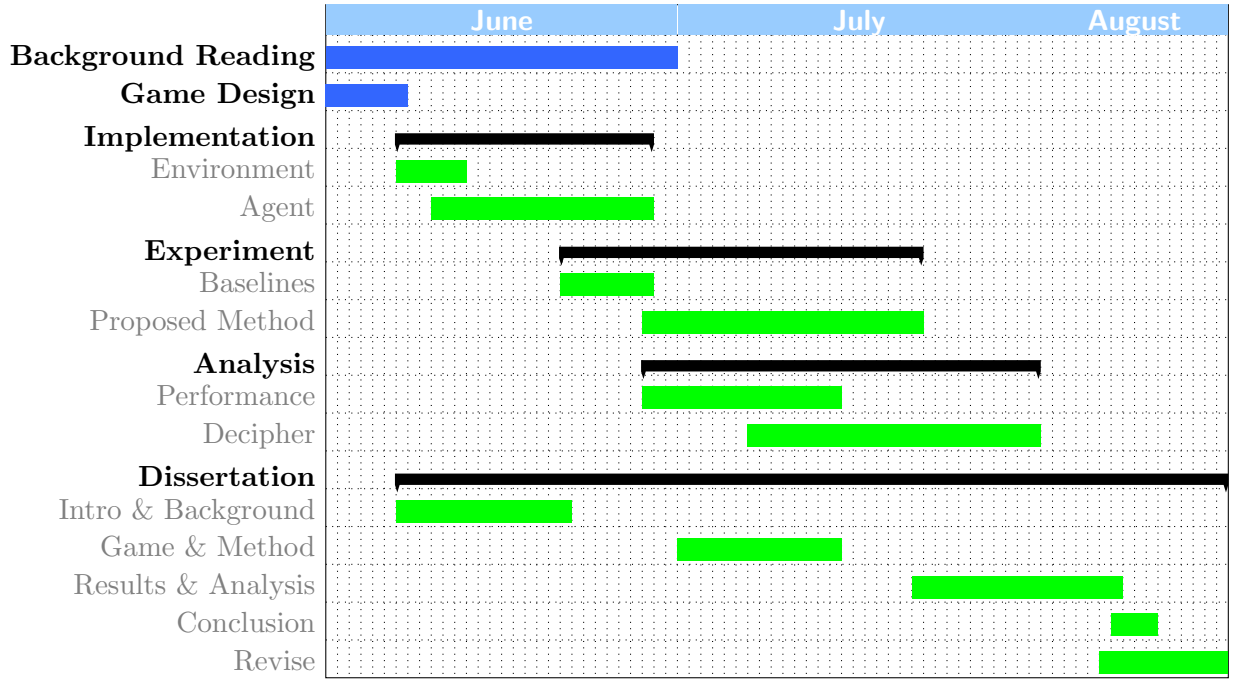


Figure 2: Gantt Chart of the activities defined for this project.

Milestone	Week	Description
$M_1$	2	Feasibility study completed
$M_2$	5	First prototype implementation completed
$M_3$	7	Evaluation completed
$M_4$	10	Submission of dissertation

Table 1: Milestones defined in this project.

Deliverable	Week	Description
$D_1$	6	Software tool for ...
$D_2$	8	Evaluation report on ...
$D_3$	10	Dissertation

Table 2: List of deliverables defined in this project.