Simulating Auto–organisation of shared lexicon: Luc Steels' Embodied guessing game model

Term Project
CS784 (Language Acquisition)
Instructors: Prof A Raina, Prof H Karnik

Submitted by:
Anand Vivek Srivastava (Y1043)

Table of Contents

Introduction

The focus of linguistics has usually been on description of sentences of a language that is the grammar of the language. There have been attempts to understand the commonality in various natural languages, the existence of a universal grammar. All these studies have assumed an existence of a shared lexicon among agents. There is however no doubt that the lexicon is not innate. This is an attempt to simulate one of the proposed mechanisms for evolution of language.

Issues in Lexical acquisition

When a few agents interact, the problems faced in understanding what the speaker meant caused arise because of the gavagai problem and the grounding problem.

- Gavagai Problem: This refers to the fact that when an agent utters a form in a particular situation, what meaning the speaker wanted to convey is usually not clear unless and until the agents in question have a common lexicon.
- Grounding Problem: This refers to the fact that it is difficult to convey meanings without relying in spatial or temporal experience. Meanings of words cannot be conveyed unless a set of concepts is shared among agents.

About the model

The model that Steels has proposed consists of a set of autonomous embodied and situated agents. The system is a closed one, so agents do not enter or leave the system. The environment also consists of a set of objects that could enter of leave the system. Each object is characterized by a set of properties, some of which might change during the course of the experiment. Agents can look at agents at make utterances one at a time, while another agent hears to the utterance and attempts to understand the meaning speaker intended to convey. Agents do not have an insight into the internal state of another agent. The agents are co-operative.

For an interaction to occur involving the speaker and the hearer, the following steps need to be performed:

- Making contact
- Topic identification
- Perception
- Encoding
- Decoding
- Feedback

The system always attempts to raise the probability of successful communication occurring.

About the Simulation Software

High level Design

The software consists of the following classes:

- **Environment**: This class corresponds to the actual environment. It has the set of **Agents**, **Objects** and **Forms** as its member attributes.
- **Agent:** This class represents a general embodied, situated agent. Member fields of an agent include :
    - Role: an agent could be a speaker, a hearer or none. This role can change every game.
    - Agent ID: This is an unique identifier for an agent
    - Scorecard: This is a two dimensional matrix used to store the form-meaning associations. A higher score means a stronger association. A score of -1 indicated no association
- **Object**: This class represents a general object that about which agents can talk. Its member attributes include
    - Properties: This is a set of characteristics of the object.
    - Object ID: This is an unique identifies for an object
- **Property**: This class represents characteristics of objects.
- **Simulation**: This class is responsible to carry out the simulation. It acts as an arbiter when required. This class also logs the games if required .

The system also uses a set of data files used to initialize the system.

Steps in the game

Simulation proceeds in the following manner:

1. Environment is initialized
2. Two agents are picked randomly, first one is made the speaker, the second one takes on the role of the hearer
3. The speaker chooses topic (the meaning) that it wants to convey. This is done in a random fashion like by tossing a coin
4. The speaker finds a set of objects that satisfy the chosen characteristic.
5. Speaker chooses a form that according to it conveys the topic. In case no former association exists, the agent generates a new form.
6. The speaker utters the form and the other agent hears it.
7. Hearer guesses topic that according to it is referred to by the form. The hearer points out a set of objects to the speaker, which according to it satisfy the guessed criteria. In case the hearer is not sure, it requests the speaker to point out the object set.
8. Speaker sends a success event in case the objects picked by hearer indeed satisfy the chosen topic. Otherwise, a failure event is generated.
9. Scorecards of both agents are updated for the form used to insure better communication success in future.

Steps from 2 to 10 constitute a game. A large number of games are carried out.

Updating the scorecard

1. **For the speaker**:
    - **Failure scenario**: The association between the form used and the topic chosen is made weaker by decrementing the corresponding value. This ensured that this form meaning mapping it is less likely to be used in the future, hence increasing success rate.
    - **Success scenario**: The association between the form used and the topic chosen is strengthened by incrementing the corresponding value.
2. **For the hearer**:
    - **Failure scenario**: A set of topics are deduced from the set of objects pointed out by the speaker. The associations between the form used and deduced the topics is strengthened by incrementing the corresponding value. This ensured that this form meaning mapping when used in future is more likely to be understood by this agent.
    - **Success scenario**: The association between the form used and the topic chosen is strengthened by incrementing the corresponding value.

Representation of form-topic mapping

The following figure shows the form-topic association matrix. The x-axis represents the various forms, which the topics form the y-axis. A higher score represents a stronger association.

```
Agent# 1
#1=0    -1      -1      10      443     -1      -1      -1      1       3
#1=1    39      -1      -1      -1      469     -1      -1      -1      -1
#2=0    -1      -1      -1      -1      -1      -1      -1      424     3
#2=1    1       -1      -1      -1      -1      -1      384     -1      -1
#3=0    -1      441     -1      -1      -1      -1      -1      -1      -1
#3=1    -1      -1      -1      -1      -1      432     -1      -1      -1

Agent# 2
#1=0    -1      -1      21      389     -1      -1      -1      -1      1
#1=1    54      -1      -1      -1      447     -1      -1      -1      -1
#2=0    -1      -1      -1      -1      -1      -1      -1      426     1
#2=1    -1      -1      -1      -1      1       -1      418     -1      -1
#3=0    -1      412     -1      -1      -1      -1      -1      -1      -1
#3=1    -1      -1      -1      -1      -1      488     -1      -1      -1

Agent# 3
#1=0    -1      -1      34      413     -1      -1      -1      5       -1
#1=1    3       -1      -1      -1      445     -1      -1      -1      -1
#2=0    -1      -1      -1      -1      -1      -1      -1      441     2
#2=1    93      -1      -1      -1      1       -1      355     -1      -1
#3=0    -1      452     -1      -1      -1      -1      -1      -1      -1
#3=1    -1      -1      -1      -1      -1      425     -1      -1      -1
```

Fig.: Association matrices for three agents. The system consisted of 3 agents, 15 objects each having 3 properties. The simulation was run for 2000 games.

Results

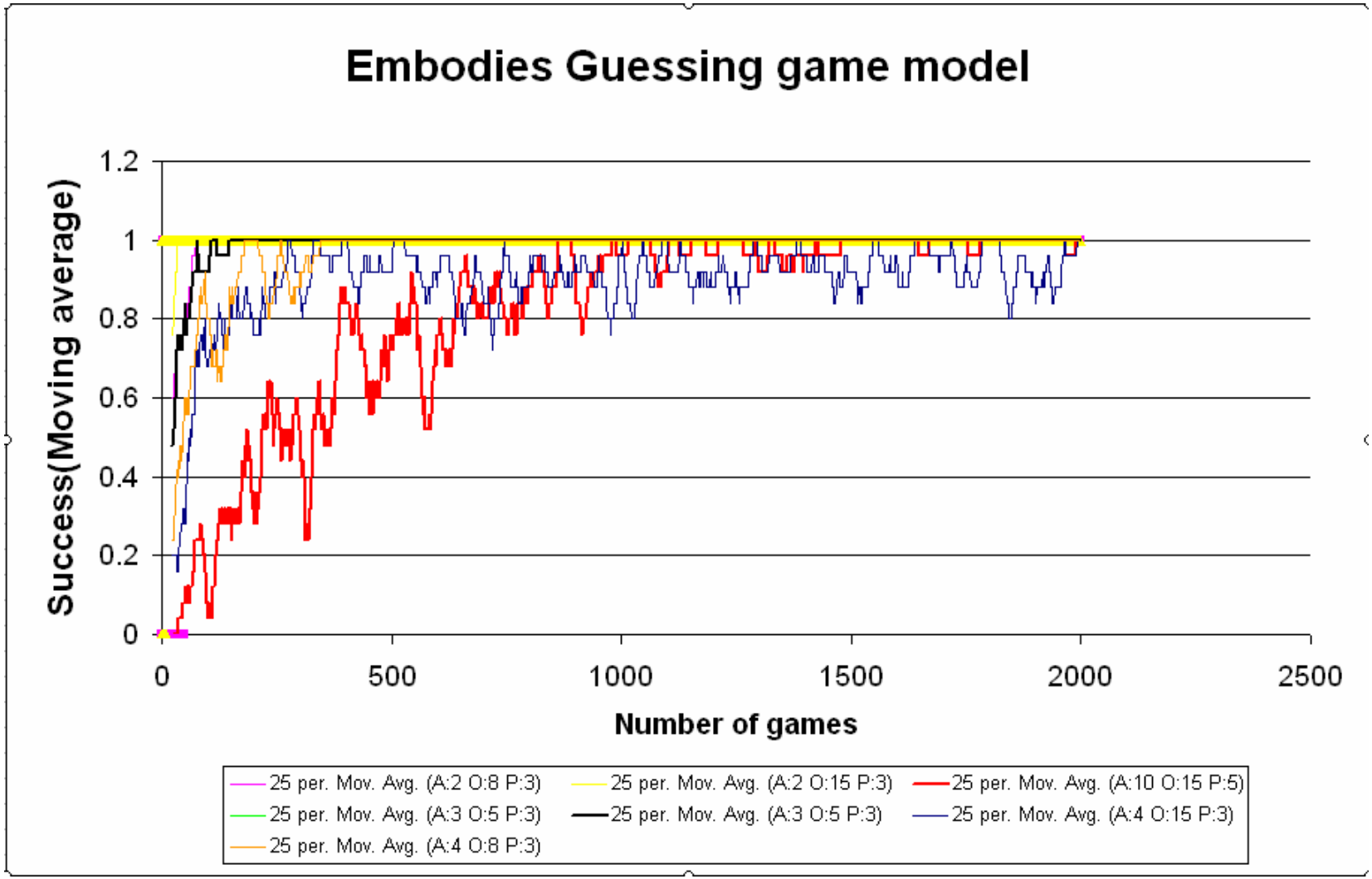
After running the simulation for various numbers of agents, objects and topics, the system appears to converge for almost all systems.
Some features of the solution:

- **High communication success:** The association matrices converge to similar values distinct peaks. This indicated strong association.
- **Unused forms**: Some forms do not have high values for any topic, indicating that the form is no longer in use by any agent for any topic. This generally occurs when an agent unknowingly generates new forms for topics for which other agents already have associations.
- **Low ambiguity**: The matrices generally show single peaks in a column, so most forms are uniquely mapped to a topic
- **Low synonymy**: The matrices generally show single peaks in a row, so most topics are uniquely mapped to a form. However, sometimes we do see more that one form being used, though some more frequently than the others are.
- **Rate of convergence depends on number of agents:** Converges slowly as the number of agents increases. As we can see from the graph on the next page, with increased number of agents it takes more games for the system to organize itself.

Cases when the simulation does not converge, or converges with poor success ratio:

- **Too simple environments**: When we use two or fewer objects, the agents fail to arrive at a shared lexicon. This occurs because the system is not complex enough to be able to convey all meanings
- **Uncaptured meaning**: When the objects are such that some one or more properties do not take on one or more values, the objects fail to convey the meaning. This problem clearly points out the grounding problem.

The graph on the following page shows the communication success as a function of the number of games that have been played. Each graph represents a moving average of last 25 games for a total of 2000 games. The number of agents, objects and topics was varied and some of the simulation results have been plotted.
It can be seen that in the simulation with ten agents, fifteen objects and five properties takes the longest to converge and have the poorest success rate.


Embodies Guessing game model
Success(Moving average)
1.2
1
0.8
0.6
0.4
0.2
0
0
500
1000
1500
2000
2500
Number of games
25 per. Mov. Avg. (A:2 O:8 P:3)
25 per. Mov. Avg. (A:2 O:15 P:3)
25 per. Mov. Avg. (A:10 O:15 P:5)
25 per. Mov. Avg. (A:3 O:5 P:3)
25 per. Mov. Avg. (A:3 O:5 P:3)
25 per. Mov. Avg. (A:4 O:15 P:3)
25 per. Mov. Avg. (A:4 O:8 P:3)

References

- "Grounding adaptive language games in robotic agents" - Luc Steels and Paul Vogt; 1997
- "Collective learning and semiotic dynamics" - Luc Steels, F Kaplan; 1999
- "Bootstrapping Grounded Word Semantics" - Luc Steels, F Kaplan
- "Self-organising vocabularies" – Luc Steels,1996

Email to avivek[@]cse.iitk.ac.in in case the source code is required