Quirks of deep emergent languages

Marco Baroni
DLRL Summer School
August 2020
My aim

• This is an introduction to the nascent area of deep network emergent communication studies
  • What happens when two or more deep networks are encouraged to develop their own discrete communication code in order to solve tasks to together?
• Field is still too young and small to attempt a systematic survey (no standard benchmarks, not much established knowledge...)
• My main aim is to (hopefully) make you curious enough about the topic, so that you might decide to contribute!
Your background

• Not a technical presentation
• Anybody with a vague idea of what deep networks are and how they are trained can follow
• Some topics might be more interesting to students with high-level background in deep network architectures and continuous vs. discrete optimization
Outline

• Language emergence in deep networks: why and how
• Emergent languages and their quirks
• Further topics and future directions
Deep networks do amazing things... but just one thing at a time!

https://cs.stanford.edu/people/karpathy/cnnembed/
How can we harness their powers more flexibly?

• Manual gluing?

$\text{egrep}'^a'$ in.txt | sort | uniq -c > out.txt

https://hackernoon.com/deepmind-relational-networks-demystified-b593e408b643
How can we harness their powers more flexibly?

• Manual gluing?
• Good Old AGI?
How can we harness their powers more flexibly?

• Manual gluing?
• Good Old AGI?
• Language!

Our bet!

Communication games

http://www.publicdomainfiles.com/

Lazaridou et al. ICLR 2017,
Havrylov and Titov NIPS 2017,
Kottur et al. EMNLP 2017,
Evtimova et al. ICLR 2018,
...
Communication games

• Two networks must jointly solve a task
• *Sender* network sees some input (e.g., a target image) and sends a *message* to *Receiver* network
• Receiver gets some input, including Sender message, and performs an action (e.g., point to target image) to complete the task
• The message is a single *discrete* symbol or a sequence of *discrete* symbols from a fixed alphabet
• Networks rewarded for task success only, *no supervision* on the messages generated by Sender
Communication games

• Two networks must jointly solve a task
  • Sender network sees some input (e.g., a target image) and sends a message to Receiver network
  • Receiver sees some input, including Sender message, and performs an action (e.g., point to target image) to complete the task
• The message is a single discrete symbol or a sequence of discrete symbols from a fixed alphabet
• Networks rewarded for task success only, no supervision on the messages generated by Sender

this is the only way in which we constrain the emergent system to be human-language-like
Communication games

- Two networks must jointly solve a task
- *Sender* network sees some input (e.g., a target image) and sends a *message* to *Receiver* network
- Receiver sees some input, including Sender message, and performs an action (e.g., point to target image) to complete the task
- The message is a single *discrete* symbol or a sequence of *discrete* symbols from a fixed alphabet
- Networks rewarded for task success only, *no* supervision on the messages generated by Sender

... because... where could we get the training data from?
Sender and Receiver are deep networks built out of standard input/output processing components, for example...
Input images will be processed by a Convolutional Neural Network module.
Agent architectures

Sender might emit a sequence of messages with a recurrent network conditioned on an image representation.
Agent architectures

Similarly, Receiver might process the message produced by Sender by reading it, symbol-by-symbol, with a recurrent network.
Other components, e.g., fully connected feed-forward layers, might complete the architectures, as required by the dynamics of the game.
Receiver can be straightforwardly trained by backpropagating gradients from standard cost function (e.g., cross-entropy), or with any other appropriate technique (e.g., reinforcement learning)
Since we are sampling discrete symbols from Sender’s output, we can’t backpropagate gradients back to Sender, so we need to use special techniques to estimate these gradients.
Since we are sampling discrete symbols from Sender’s output, we can’t backpropagate gradients back to Sender, so we need to use special techniques to estimate these gradients.

Current work in the area uses the REINFORCE algorithm (Williams, *Connection Science* 1992) or the Gumbel-Softmax approximation (Jang et al, ICLR 2016, Maddison et al, ICLR 2016)
If Sender produces discrete symbols, you have to use special optimization techniques, *independently of whether your final loss function is differentiable or not*.
But why discrete messages?
But why discrete messages?

Because *humans in the loops*
But why discrete messages?

Because humans in the loops

There might be a good reason for human language to be discrete: beneficial properties of a bottleneck (Kharitonov et al ICML 2020)
But why discrete messages?

Because **humans in the loops**

There might be a good reason for human language to be discrete: beneficial properties of a **bottleneck** (Kharitonov et al ICML 2020)

There is however also much research using **continuous communication channel** (e.g., Sukhbaatar et al NIPS 2016)
Play with emergent language without worrying about the technical details!

https://github.com/facebookresearch/EGG

EGG 🐥: Emergence of lanGuage in Games

Introduction

EGG is a toolkit that allows researchers to quickly implement multi-agent games with discrete channel communication. In such games, the agents are trained to communicate with each other and jointly solve a task. Often, the way they communicate is not explicitly determined, allowing agents to come up with their own 'language' in order to solve the task. Such setup opens a plethora of possibilities in studying emergent language and the impact of the nature of task being solved, the agents' models, etc. This subject is a vibrant area of research often considered as a prerequisite for general AI. The purpose of EGG is to offer researchers an easy and fast entry point into this research area.
Outline

• Language emergence in deep networks: why and how
• Emergent languages and their quirks
• Further topics and future directions
The emergence of words

Lazaridou et al. ICLR 2017
The emergence of words
The emergence of words

Pexels, Pixabay
Emergent languages are tricky!

At training time...

dog

Bouchacourt and Baroni EMNLP 2018
Emergent languages are tricky!

At test time!
What would you talk about if all you had to do in life was to discriminate picture pairs?

"dog" (= "category 157 of 500")

9 bits

"larger average intensity in 16 pixels at image center"

1 bit

saves time and effort!
Emergent languages are tricky!

In emergent language studies, understanding the properties of the language is almost always more important than boosting task performance!
Are emergent language words efficient?

Chaabouni et al.
Anti-efficient encoding in emergent communication
NeurIPS 2019
Zipf's law of abbreviation (ZLA)

In human languages, word frequency and length are inversely correlated

**top 10 most frequent words**
- the
- of
- and
- to
- in
- I
- that
- was
- his
- he

**10 random rare(st) words**
- anadromous
- barmaster
- cruddy
- gemstone
- gonzo
- idolization
- pigling
- sanguinity
- unpredictability
- walkman

Source: Gutenberg books frequency list, from Wiktionary
The Skewed-Frequency Referent Game

inputs ("referents") sampled from 1k numbers (encoded as 1-hot vectors), with sampling probability obeying power-law distribution
Most frequent words:
English vs emergent language

<table>
<thead>
<tr>
<th>English</th>
<th>Emergent language</th>
</tr>
</thead>
<tbody>
<tr>
<td>the</td>
<td>Ilmuhmummmmmmmzuuyyzzvqzplan</td>
</tr>
<tr>
<td>of</td>
<td>naauhhhhhhhhhpucczzpbqaoaqpln</td>
</tr>
<tr>
<td>and</td>
<td>shmhhhhhxxxxxxxxxsummyytttlcglgl</td>
</tr>
<tr>
<td>to</td>
<td>jahhhhhhuxxxxxxxxxxxzychoyoawb</td>
</tr>
<tr>
<td>in</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
</tr>
<tr>
<td>that</td>
<td></td>
</tr>
<tr>
<td>was</td>
<td></td>
</tr>
<tr>
<td>his</td>
<td></td>
</tr>
<tr>
<td>he</td>
<td></td>
</tr>
</tbody>
</table>
Anti-efficient word encoding in emergent language

more frequent words significantly longer, unique case compared to human languages and animal communication systems (Ferrer-i-Cancho et al., Cognitive Science 2013)
The origin of anti-efficient encoding

- Artificial Sender does not need to save articulatory effort
- Receiver (even before training) prefers longer words, as they are easier to discriminate
Explicitly penalizing longer words
Are emergent languages compositional?

Generalization
Compositionality and generalization

Train
"banana"
"blue"

Test
"blue banana"

Slide credit: Rahma Chaabouni
What makes an (emergent) language compositional?¹

¹ Where “compositional” is *not just a synonym* for: “able to generalize”
What makes an (emergent) language compositional?

Our intuition:

A compositional language is one where it is easy to read out which parts of a linguistic expression refer to which components of the input.

Just having systematic rules to put together parts in order to express a complex meaning is not enough...
Naïve compositionality

Applies when the only way to combine primitive input elements is to assemble them in a collection:

Examples: a list of attribute-value pairs (equivalently, a vector of values), a set of objects, a list of properties (‘blue’, ‘banana’), ...

90% of current emergent language simulations
Naïve compositionality

Applies when the only way to combine primitive input elements is to assemble them in a collection:

Then, a language is naively compositional if the atomic symbols in its expressions refer to single input elements, independently of either input or linguistic context.
Naïve compositionality

Applies when the only way to combine primitive input elements is to assemble them in a collection:

Then, a language is **naïvely compositional** if the atomic symbols in its expressions refer to single input elements, *independently of either input or linguistic context*

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>L</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Naïvely compositional:
- A -> 29
- L -> 12
- M -> 31

Non-compositional:
- ALM -> [12,29,31]

Non-naïvely compositional:
- A -> 29 if immediately followed by L
- L -> 12 if other input values are odd
- M -> 31 if one of previous symbols is A
Naïve compositionality

Applies when the only way to combine primitive input elements is to assemble them in a collection:

Then, a language is **naïvely compositional** if the atomic symbols in its expressions refer to single input elements, *independently of either input or linguistic context*.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>L</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Naïvely compositional:
A -> 29
L -> 12
M -> 31

Non-compositional:
ALM -> [12, 29, 31]

Non-naïvely compositional:
A -> 29 if immediately followed by L
L -> 12 if other input values are odd
M -> 31 if one of previous symbols is A

we don’t like either!
Compositionality and generalization in emergent languages

Chaabouni, Kharitonov et al. ACL 2020
Compositionality and generalization in a communication game
Compositionality and generalization in a communication game

TRAINING

attribute-value list input

12
29
31

reconstruction task

A L M

12
29
31

TESTING

this is a combination that was not presented at training time (although each single value in it was)

T A F

22
11
56

53
Compositionality and generalization in a communication game

This is a combination that was not presented at training time (although each single value in it was).

Attribute-value list input

TRAINING

TESTING

generalization: are agents successful on unseen combinations?

Compositionality and generalization in a communication game

This is a combination that was not presented at training time (although each single value in it was).

Attribute-value list input

TRAINING

TESTING

generalization: are agents successful on unseen combinations?
Compositionality and generalization in a communication game

**TRAINING**

- Attribute-value list input:
  - 12
  - 29
  - 31

**TESTING**

- 22
- 11
- 56

**Compositionality**

- A L M

**Generalization**

- 12
- 29
- 31

This is a combination that was not presented at training time (although each single value in it was).

**Reconstruction Task**

Is the emergent code (naively) compositional?

**Generalization**

- Are agents successful on unseen combinations?
Quantifying (one type of) naïve compositionality

Positional disentanglement measures strong form of naïve compositionality: to what extent do symbols in a certain position univocally refer to different values of the same attribute.

\[
posdis = \frac{1}{\text{clen}} \sum_{j=1}^{\text{clen}} \frac{\mathcal{I}(s_j; a_1^j) - \mathcal{I}(s_j; a_2^j)}{\mathcal{H}(s_j)}
\]

\[
a_1^j = \arg\max_a \mathcal{I}(s_j; a); a_2^j = \arg\max_a \mathcal{I}(s_j; a)_{a \neq a_1^j}
\]

Similar results in experiments with other compositionality measures!
Do emergent languages support generalization?
Do emergent languages support generalization?

Yes, in function of how varied the training input is!

A general pearl of wisdom: do not test neural network generalization capabilities in small toy worlds!

size = $N_{vals}^{N_{atts}}$
Is compositionality needed for generalization?
No!... no correlation between generalization and compositionality!

top compositionality scores are far from theoretical max (=1)!
What’s going on?
Deep networks have different biases than we do!

Kharitonov and Baroni: Emergent Language Generalization and Acquisition Speed are not Tied to Compositionality

Coordinate and rotated languages

- **Inputs:** coordinates within unit circle
- **Manually-crafted languages** instead of trainable Sender
- **Naïvely compositional “coordinate” language:** two symbols directly corresponding to (discretized) coordinates
  - “Rotated” language: symbols correspond to (discretized) coordinates after rotating the axes by $\pi/4$
    - Identifying either element always requires looking at both symbols, resulting in a very entangled encoding

Image credit: Eugene Kharitonov
Coordinate and rotated languages

- ease of learning
- ease of use
• In the rotated language a linear transformation links (values denoted by) symbols and inputs
• Linear transformations are neural networks’ favorite sport, so for Listener highly entangled rotated language is as easy as perfectly (naïve) compositional coordinate language
• There is nothing universal about more (naïvely) compositional languages being easier to learn and use!
Ad interim conclusion

• Deep networks can make use of discrete signals to successfully communicate, even when no direct supervision on the communication channel is provided.
• Emergent code will “over-fit” task-specific needs and deep network biases.
• Is this a bug or a feature?
Outline

• Language emergence in deep networks: why and how
• Emergent languages and their quirks
• **Further topics** and future directions
Aligning with human language

Lazaridou et al ICLR 2017 (see figure)
Havrylov and Titov NeurIPS 2017
Lee et al EMNLP 2019
Lazaridou et al ACL 2020
...

How to avoid *drifting* while keeping emergent language grounded in the communication game?
Dynamic environments

Mordatch and Abbeel AAAI 2018 (see figure)
Das et al ICML 2019
Kim et al ICLR 2019
...

• Learning to schedule communication and picking addressees
• Do agents listen to each other? (Lowe et al AAMAS 2019)
• Does communication bring added value?
Self-interested agents

Cao et al ICLR 2018 (see figure)
Jacques et al 2019

• Think self-driving cars...
• The dangers of *cheap talk* (Crawford and Sobel *Econometrica* 1982)
Outline

• Language emergence in deep networks: why and how
• Emergent languages and their quirks
• Further topics and future directions
Analyzing emergent language

• Nearly each current study uses different environments, training regimes, measures
  • Often *ad hoc* for very specific setups
  • Often hard to implement

• No standard environments, problem definitions, benchmarks...
Towards a universal deep-network language
Fast generalization to new things in realistic setups

https://www.flickr.com/photos/el_cajon_yacht_club/9718843082
Wikipedia
Towards a universal deep-network language
Fast generalization to new tasks
Towards a universal deep-network language
Fast generalization to new agents

Wikipedia
To learn a bit more about deep-network emergent language...

Angeliki Lazaridou and Marco Baroni:
Emergent Multi-Agent Communication in the Deep Learning Era

thank you

https://github.com/facebookresearch/EGG