Principled Evaluation of Deep Learning-Based Emergent Communication

Brendon J. Boldt

February 12, 2026

Language Technologies Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania

Thesis Committee

David Mortensen, *Chair*	Carnegie Mellon University
Yonatan Bisk	Carnegie Mellon University
Daniel Fried	Carnegie Mellon University
Kenny Smith	University of Edinburgh

Thesis Statement

Advancing science and engineering with deep learning-based emergent language requires principled, environment-agnostic analytical algorithms.

Abstract

Emergent communication is the field of research which studies how human language-like communication systems evolve from scratch in agent-based simulations. The most recent incarnation of this topic, starting in 2016, has focused on leveraging recent advancements in deep neural networks, reinforcement learning, and natural language processing. Emergent communication, as a paradigm, has significant potential applications from powering unprecedentedly detailed simulations of how humans invent, acquire, and use language to providing rich, embodied language data for training and evaluating models free of contamination or surveillance. Despite this potential, the field has yet to make progress towards the more revolutionary applications because it lacks the methodological resources to enable cumulative research. The design space of emergent language is dizzyingly large, and the outputs of said environments are similarly difficult to interpret. As a result, analyses of emergent languages, and consequently their findings findings, are often "one-off", unable build on prior work or produce generalizable contributions.

This thesis, then, advances the field of emergent communication by developing the resources necessary for a cumulative research paradigm, something that is critical to the advancement of any area of science or engineering. Specifically, it introduces analytical methods for deep learning-based emergent communication that enable measurable progress in the field with a particular mind towards solving practical applications and improving scientific understanding. I first review the particular ways in which emergent communication can solve practical applications and improve scientific understanding. This is followed by introducing emergent language data resources which enable empirical evaluation across a variety of emergent languages. These resources are then used to develop (1) a deep transfer learning-based evaluation metric for emergent communication to measure the practical applicability of emergent language and (2) algorithms for discovering the morphology of emergent languages as a foundation for further linguistic analysis.

Contents

1 Introduction **5**
- 1.1 Background . . . 6
- 1.2 Motivation . . . 8
- 1.3 Approach . . . 10

2 Review of the Applications of Emergent Language **12**
- 2.1 Introduction . . . 12
- 2.2 Internal Goals . . . 17
- 2.3 Task-Driven Applications . . . 26
- 2.4 Knowledge-Driven Applications . . . 34
- 2.5 Discussion . . . 45
- 2.6 Conclusion . . . 48

3 Building a Library of Emergent Languages **49**
- 3.1 Introduction . . . 49
- 3.2 Related Work . . . 50
- 3.3 Design . . . 51
- 3.4 Content . . . 53
- 3.5 Analysis . . . 57
- 3.6 Discussion . . . 60
- 3.7 Conclusion . . . 62

4 Evaluation with Deep Transfer Learning **63**
- 4.1 Introduction . . . 63
- 4.2 Related Work . . . 65
- 4.3 XferBench . . . 66
- 4.4 Experiments . . . 69
- 4.5 Results . . . 73
- 4.6 Discussion . . . 75
- 4.7 Conclusion . . . 77
- 4.8 Limitations . . . 77

5 Optimizing for Statistical Similarity Human Language **78**
- 5.1 Introduction . . . 78
- 5.2 Related Work . . . 80

5.3 Methods . . . 81
5.4 Experiments . . . 83
5.5 Analysis . . . 86
5.6 Discussion . . . 88
5.7 Conclusion . . . 89

6 Discovering Morphemes in Rich Corpora **91**
6.1 Introduction . . . 91
6.2 Problem Definition . . . 92
6.3 Algorithm . . . 94
6.4 Empirical Validation . . . 96
6.5 Analysis of Emergent Languages . . . 101
6.6 Discussion . . . 104
6.7 Conclusion . . . 104
6.8 Limitations . . . 104

7 Communicating with Induced Morphological Phrasebooks **106**
7.1 Introduction . . . 106
7.2 Related Work . . . 108
7.3 Methods . . . 108
7.4 Experiments . . . 111
7.5 Discussion . . . 115
7.6 Conclusion . . . 118
7.7 Limitations . . . 118
7.8 Ethical Considerations . . . 119

8 Conclusion **120**

Bibliography **124**

A Review of the Applications of Emergent Language **149**

B Building a Library of Emergent Languages **156**

C Evaluation with Deep Transfer Learning **163**

D Optimizing for Statistical Similarity to Human Language **171**

E Discovering Morphemes in Rich Corpora **180**

F Communicating with Induced Morphological Phrasebooks **191**

Prologue

The year is 2018. In the last couple years, AlphaGo defeated Lee Sedol, reigning Go world champion (BBC, 2016). OpenAI has been making for itself as a reinforcement learning lab with OpenAI Gym (Brockman et al., 2016) and OpenAI Five (OpenAI et al., 2019). New deep algorithms are being introduced left and right: Deep Q-Network, Advantage Actor Critic, Deep Deterministic Policy Gradient, Trust Region Policy Optimization, Proximal Policy Optimization (Yuxi Li, 2018). And now, AlphaZero (Silver et al., 2017) has made its predecessor, AlphaGo, obsolete by discarding human-generated training data and doubling down on the power of deep reinforcement learning. It is in this milieu that I begin to develop the ideas that would grow into this thesis. Richard Sutton's "Bitter Lesson" is only a year way from this, all but declaring application-specific human expertise dead and done for at the hands of scaling compute, parameters, and data.

Expert-designed systems reigned in games like chess, starting with Deep Blue's 1997 victory over Garry Kasparov (IBM, n.d.), but AlphaZero, replacing domain knowledge with deep learning-driven search, dealt a mighty blow to the paradigm's dominance. It is here we can draw an analogy with the subject matter of this thesis: The origins of human language and its evolution had for decades been studied through expert-designed simulations based on theoretically-motivated models (i.e., *emergent language*) (Kirby, 2002), but perhaps deep reinforcement learning could revolutionize the study of language evolution like it had many a time before. Just as AlphaZero had re-derived many chess strategies based only the rules of the game (as well as found some new ones), maybe similar techniques could rederive the communication strategies which characterize human language. Toward this hope, papers such as Foerster et al. (2016), Lazaridou et al. (2016), and Havrylov et al. (2017) took the first steps towards harnessing deep RL for emergent language.

One way to look at scientific exploration through simulation is searching for the "interesting" interface between order and chaos much like the boundary region of the Mandelbrot set. Too much order? Uninteresting. The simulation is so tightly defined that results are obvious from the starting conditions. Too much chaos? Uninteresting. The simulation so unconstrained so as to exhibit no discernible patterns or just collapse to a trivial outcome. While the classical approaches to emergent language start with order and introduce chaos to find the interesting, deep learning-based emergent language begins in the chaos: design the environment and let multi-agent reinforcement learning work its magic, hopefully the order is there.

But if the order is not there, what are we to do?

Read this thesis.

Chapter 1

Introduction

Large language models are good at mimicking human language. Some might even say they are good at *using* human language, but this is either imprecise or inaccurate: LLMs' production of text is based the statistical likelihood of meaningless (so far as they are concerned) tokens, fine-tuned to humans' preferences. This is in contrast to humans' use—and even more so their acquisition—of language which is laden with meaning derived from the rich internal, physical, and social context which permeates language use. The end result is that LLMs' approximation of human language fails at a number of tasks but, more significantly, is limited in providing insights into the nature of human language itself.

Emergent language (also known as *emergent communication*[1]) is an alternative paradigm to developing language-capable models that does not train on human language data but rather invents a communication system *de novo*. In its most basic form, emergent language comprises a simulation using neural network-based agents which are trained to cooperatively complete some task in a virtual environment. These agents are equipped with a communication channel of discrete tokens with no *a priori* meaning—the meaning of the messages is established through the optimization process encouraging communication which is advantageous to completing the task, more analogous to humans' use of communication.

Emergent language differs from the more "traditional" approach to modeling language that LLMs use in that it does not try to mimic patterns in human language but instead tries to rederive such patterns from functional pressures analogous to those hypothesized to have guided human language's own evolution. Since this process language emerging is far more analogous to how human language develops and is learned, it has a much greater potential to contribute to the scientific understanding of human language. Furthermore, certain practical tasks might lie beyond the reach of the mimicry approach LLMs employ due to the often superficial "understanding" they have of language; these problems, too, can be addressed by emergent language which models not only the surface features of language but also its semantics and social context. Finally, generating language data through emergent communication avoids many of the ethical issues that crop up with LLMs' dependence on human language data from amplifying toxicity from the Web to freely (ab)using copyrighted and personal content (Weidinger et al., 2021; Carlini et al., 2021).

Here is the problem: despite its potential, the field of emergent communication has not

[1] I *will* use these interchangeably.

made measurable progress toward its more revolutionary applications to scientific understanding or practical problems. This thesis, then, introduces resources and algorithms for emergent language that enable measurable progress in the field, enabling practitioners to tackle its most significant applications through cumulative contributions. It accomplishes this first by reviewing the particular ways in which emergent communication can solve practical applications and improve scientific understanding. This is followed by introducing emergent language data resources which permit empirical evaluation across a variety of emergent languages. These resources are then used to develop environment-agnostic analytical algorithms centering on the key applications of emergent communication. Namely, this thesis introduces (1) a deep transfer learning-based evaluation metric for emergent communication to measure the practical applicability of emergent language and (2) algorithms for discovering morphology of emergent languages as a foundation for further linguistic analysis. Such tools help to discern order among the often chaotic output of these simulations.

1.1 Background

Agent-based models of language evolution The origin of human language has been a perennial question of human inquiry from ancient mythology to contemporary linguistics. While much of the history of this question is relevant to this work, I will limit this discussion to the most salient approach: namely that of computer simulations investigating the origin and evolution of human language. This intersection between linguistics and computer science gained momentum in the 1990's with the advances in computational power enabling the use of *agent-based modeling* for simulations of communication systems (Steels, 1997). Agent based models are simulations which model systems at the level of *individuals* and their interactions in an environment; such models allow us to see how "global properties emerge by local interactions" in order to better understand complex systems like language (Steels, 1997).

Employing agent-based models to study the origin and evolution of human language is what is termed *emergent language* (or *emergent communication*) insofar as the phenomenon of language (or at least some communication system) emerges from the features of the agents and their environment. Such techniques have been applied to a number of closely related topics including (Kirby, 2002): how genetically-defined communication systems (like animal communication) might spontaneously emerge (MacLennan et al., 1993), how learned systems (like human language) are transmitted (Smith, 2002), how semantic and syntax could emerge in the context of language (Steels, 1999; Batali, 1998), and how language and language *acquisition* could co-evolve (Kirby et al., 1997).

Deep learning vs. classical approaches The work mentioned above I term *classical* emergent language (CEL) insofar as it uses more historically typical modeling techniques: namely, the behavior of the agents and the environment is largely hardcoded into the implementation as part of the design of the simulation. The learning behaviors employed by the agents are relatively simple and constrained, drawing from classical AI and machine learning techniques (e.g., symbolic methods, Bayesian learning). In short, CEL builds computer models to test formal models.

Deep learning-based emergent language (DEL)—the body of literature in which this thesis is most directly situated—it is the rebellious child of CEL and the deep reinforcement learning boom of the late 2010's (cf. the Prologue). It applies the techniques of deep neural networks and multi-agent reinforcement learning to the same questions presented by CEL.[2] I say rebellious here because, rather than being a harmonious union of the two, (1) DEL diverges from deep reinforcement learning by being more concerned with the byproducts of the learning process (viz. the communication protocols) than with effectively solving the particular tasks (communicative success), and (2) DEL diverges from CEL by a relative absence of grounding in the broader linguistics literature, often starting nearly from scratch in the formulation of hypotheses and experimental approaches. Furthermore, implementing any such formal models from CEL in DEL is made difficult by the complexity of deep learning approaches.

Nevertheless, the intention of applying deep learning to emergent language is founded in improving the realism of the agent-based models which are key to the classical approach. In fact, just as Simon Kirby argues for agent-based modeling[3] approaches for studying the evolution of language beyond "the classical alternative—analytic mathematical modeling—[which] may require the kinds of idealization that will necessitate the removal of the very network of interactions that give rise to the target of explanation" (Kirby, 2002), DEL takes the next step on the path of removing such idealizations still present in CEL to further extend the range of observable phenomena. Specifically, deep learning exhibits two features towards this end: First, it has a greater representational capacity insofar as the complexity and nuance of phenomena it can represent exceeds what is possible with classical approaches (cf. the ubiquity of deep learning in contemporary computer vision and natural language processing). Second, it also offers the prospect of introducing fewer inductive biases into the model of language evolution (via its generally-applicable learning processes), leading to fewer assumptions being built into the simulation. Finally, the shift to deep learning methods also opens up further possible applications to engineering challenges such as creating realistic synthetic language data at scale or developing robust multi-agent communication protocols.

Deep learning-based emergent language Turning our attention now fully to deep learning-based emergent language, this subfield has its genesis in 2016 with seminal papers such as "Learning to communicate with deep multi-agent reinforcement learning" (Foerster et al., 2016) and "Multi-Agent Cooperation and the Emergence of (Natural) Language" (Lazaridou et al., 2016). These were some of the first papers to applying deep multi-agent reinforcement learning to developing discrete token-based communication systems *de novo* in a way analogous to human language. With deep learning, agents could observe pixel-based images with convolutional neural networks and produce the messages with recurrent neural networks—two techniques which had made significant advances towards matching human performance in image and language processing.

The prototypical emergent communication experiment consists two or more deep neural network-based agents situated in some kind of environment or game where they must cooperate

[2] In theory, DEL is concerned with the same questions, but in practice, the bulk of investigation is limited to the emergence of semantics and syntax, as mentioned above.

[3] Referred to "artificial life" in the cited work.

in order to succeed. The agents are equipped with a communication channel consisting of discrete tokens with no *a priori* meaning; it is only through the reinforcement learning-based optimization that messages passed between agents begin to take on meaning. The resulting behavior, most especially the communication protocol, is typically the object of analysis, addressing question such as: Did an effective communication protocol emerge at all? What structural features characterize it? Do these features align at all to human language? What can we infer about language formation more generally from the above?

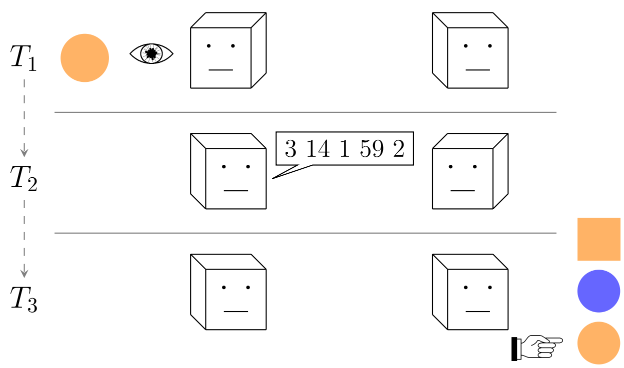
In practice, much of the literature has focused on the signaling game and the emergence of compositionality in communication (Havrylov et al., 2017; Mordatch et al., 2018; Chaabouni et al., 2022). The signaling game itself was introduced in the context of game theory by David Lewis (Lewis, 1969); it is one of the simplest possible environments for emergent communication contributing, in large part, to its popularity. The game consists of only two agents: a sender and a receiver. The sender makes on observation (e.g., an orange circle) and sends a message to the receiver who must, based on the message alone, determine the nature of the observation (e.g., it was an orange circle, not a blue circle). A visualization of the signaling game is provided in Fig. 1.1. The question of compositionality arises when we look at how the communication protocol encodes compound meanings like a red square: A compositional protocol would encode "red" and "circle" with their own words which could be reused to express meanings like a red circle or a blue square. On the other hand, holistic communication sometimes emerges where a unique word refers to red square, bearing no relation to the word(s) for red circle. Compositional communication, generally, is seen as more desirable both for practical reasons (more efficient encoding of information) as well as for its resemblance to how humans tend to encode meaning in language.

Other environments do appear in the literature such as navigation tasks or dialogue-based games (Unger et al., 2020; Brandizzi et al., 2022a). In addition to compositionality, other phenomena have been the subject of investigation such as pragmatics, transfer learning, and the information theoretic properties of emergent language (Kang et al., 2020; Yao et al., 2022; Tucker et al., 2021). For a general review of the emergent communication literature, I recommend Lazaridou et al. (2020).

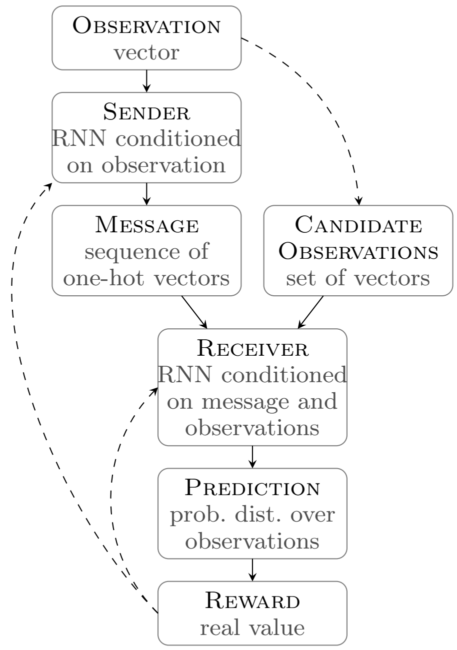
1.2 Motivation

Early on in the history of deep learning-based emergent language, papers such as "Natural Language Does Not Emerge 'Naturally' in Multi-Agent Dialog" (Kottur et al., 2017) and Bouchacourt et al. (2018) already argued the difficulty of finding informative patterns in emergent languages, and such papers would prove to be harbinger of the decade to come: there would be little measurable progress toward the farther-reaching applications of the field. Many research papers end up employing *ad hoc* analyses tailored to the particular phenomenon of interest without a clear way to unify the findings into a broader understanding. While normal science (as Kuhn (1962) terms it) often proceeds by small, additive research contributions, emergent communication has not developed research paradigm where the small contributions can truly add together.

One can read much of the literature on emergent communication, learn of many different trends that appear in particular environments, and still largely have little idea why emergent



(a) T_1: The sender (left) observes an object. T_2: The sender passes a message to the receiver (right). T_3: The receiver chooses from a handful of candidate objects.



(b) Illustration of the technical architecture of the signaling game.

Figure 1.1: An illustration of the discrimination variant of the signaling game, one of the simplest and most common environments in emergent communication research.

languages look the way they do nor what they might look like in a new environment. Furthermore, despite the potential for groundbreaking applications in natural language processing and linguistics, emergent communication has not produced any broadly-applicable contributions in either (Chapter 2). In contrast, classical emergent language research has proved fruitful, yielding results such as the iterated learning model which are more easily reincorporated into the broader field of linguistics (Kirby et al., 2014). Thus, the problems mentioned above are most pertinent to the deep learning approach to emergent language.

I intend for this thesis, then, to make foundational contributions addressing the unique structural issues in deep learning-based emergent language research that has resulted in this. In a large part, I argue that this requires making the patterns and structures present in emergent languages (if they are there) more easily observable. This is analogous to the broader project of interpretability in deep learning specified for emergent language—although it proves to play a more central role in emergent language insofar as interpretation is fundamental to the field. The methods of classical emergent language, for the most part, assume that the basic structures in the emergent languages (e.g., words) are already evident, meaning that the project of finding structure in deep emergent languages is, in a way, trying to recover some of what was lost in the classical-to-deep transition and better connect the two sides of the broader field.

1.3 Approach

Thus, this thesis introduces principled (Section 1.3.1), environment-agnostic (Section 1.3.2) analytical algorithms to support measurable progress towards both the engineering-centric (Section 1.3.3) and scientific (Section 1.3.4) goals of emergent language.

1.3.1 Finding a lodestar

Chapter 2 provides a comprehensive review on the goals and applications of emergent language research. This chapter is not a general review but review of the *goals* of this field of research because it is imperative for any field of research to understand where it is going and why. While many of the goals of deep learning-based emergent language overlap with those of classical emergent language, others extend to the machine learning world. Beyond the general clarity that articulating directions provides, this chapter performs two specific functions for this thesis. First, by articulating the goals and, in many cases, the minimal progress towards achieving them, it motivates the paradigmatic changes which this thesis proposes in later chapters. Second, the articulated goals form the very basis of the analytical methods which are introduced in later chapters. Using emergent language to pretrain language models? Evaluate them against XferBench (Chapter 4). Trying to make claims about the morphology of emergent languages? Induce a morpheme inventory with CSAR (Chapter 6). Thus this chapter provides both the motivation and scaffold for the *principled* analytical algorithms introduced the later chapters.

1.3.2 Towards unity in a diverse landscape

Chapter 3 introduces the Emergent Language Corpus Collection (ELCC), a diverse collection of emergent language corpora derived from free and open source implementations of emergent language environments. As stated above, emergent language research is often organized into little islands working with specific environments to test specific phenomena with considerably less work spanning even a small subset of the wide variety of emergent language environments. Introducing a collection of diverse emergent language corpora in a unified format (along with ready-to-run versions of the corresponding codebases) makes a significant step towards more comparative research yielding more *environment-agnostic* contributions. Furthermore, making the corpora available in a simple JSON-based format (and obviating running the environments in the first place) lowers the barrier of entry to studying emergent languages in general, especially for those without a background in computing. The promise of analysis across diverse languages is borne out within this very thesis as ELCC is employed both in XferBench (Chapter 4) and CSAR (Chapter 6).

1.3.3 Seeing language like a neural network

Chapter 4 introduces XferBench, an evaluation metric for emergent language corpora based on deep transfer learning. The intuition behind XferBench is that when a model is pretrained on emergent language data, its downstream performance on human language-based natural language processing tasks (e.g., language modeling, machine translation) correlates with its similarity to human language, as far as a deep neural network is concerned. XferBench is *environment-agnostic* insofar as it only requires an emergent language corpus as input, meaning that it can be applied to most any environment, and compare differing environments on single quantitative scale. XferBench is *principled* insofar as it is directly measuring how well emergent languages function in place of human languages in deep learning settings, a major component of the *engineering* goals of emergent language. Furthermore, the type of realism which XferBench is measuring is complementary to the *scientific* goals of emergent language which stand to benefit from more realistic simulations of linguistic phenomena. Chapter 5 demonstrates XferBench in action by using it as an objective function for hyperparameter search over the hyperparameters of an emergent language environment, effectively finding measurably more human-like emergent languages with minimal manual input.

1.3.4 Discovering order in the chaos

While the previously mentioned chapters employ black-box techniques to measure engineering-centric progress in emergent language, Chapters 6 and 7 introduce good, old-fashioned algorithms (no neural networks) for uncovering morphosyntactic structures in emergent languages. This is motivated by the fact that the *scientific* goals of emergent language require understanding the patterns and regularities present in emergent language despite the largely cryptic output produced by the simulations in question. Specifically, Chapter 6 introduces CSAR, an algorithm for inducing morphemes from emergent language corpora annotated with the individual utterances meanings. This approach is motivated by the fact that simply identifying the minimal units of meaning (i.e., morphemes) in emergent languages is non-trivial and critical to understanding most higher-level linguistic structures (e.g., morphology, syntax, pragmatics). Furthermore, CSAR makes very few prior assumptions about what kind of structure is present in the emergent languages analyzed, making it suitable to handle the wide variety of patterns which could possibly emerge. Chapter 7 puts the morphemes CSAR induces to the test by measuring their effectiveness for synthesizing and analyzing emergent language utterances produced by the neural network-based agents. The patterns observed in synthesizing and analyzing emergent language at the morpheme-level present one of the most *principled* investigations of the syntax of deep learning-based emergent languages to date.

Chapter 2

Review of the Applications of Emergent Language

Abstract

Emergent communication, or emergent language, is the field of research which studies how human language-like communication systems emerge *de novo* in deep multi-agent reinforcement learning environments. The possibilities of replicating the emergence of a complex behavior like language have strong intuitive appeal, yet it is necessary to complement this with clear notions of how such research can be applicable to other fields of science, technology, and engineering. This paper comprehensively reviews the applications of emergent communication research across machine learning, natural language processing, linguistics, and cognitive science. Each application is illustrated with a description of its scope, an explication of emergent communication's unique role in addressing it, a summary of the extant literature working towards the application, and brief recommendations for near-term research directions.[1]

2.1 Introduction

Deep learning-based methods in natural language processing and multi-agent reinforcement learning provide a powerful way simulate how human language-like communication systems emerge *de novo*. This area of research is called *emergent communication* or *emergent language*. Multi-agent reinforcement learning-based systems like AlphaZero (Silver et al., 2017) and OpenAI's hide-and-seek agents (B. Baker et al., 2020) have leveraged self-play to exhibit convincing examples of complex behavior emerging from basic environment dynamics. Such deep reinforcement learning techniques were applied to discrete communication systems starting in 2016 and 2017 with papers like Foerster et al. (2016), Lazaridou et al. (2016), Havrylov et al. (2017), and Mordatch et al. (2018). Although replicating as complex a behavior as human language is intuitively important, it is necessary to complement such notions with clear directives as to how it could apply to other areas of science, technology, and engineering.

[1]Based on "A Review of the Applications of Deep Learning-Based Emergent Communication" appearing in the *Transactions on Machine Learning Research* (Boldt et al., 2024a).

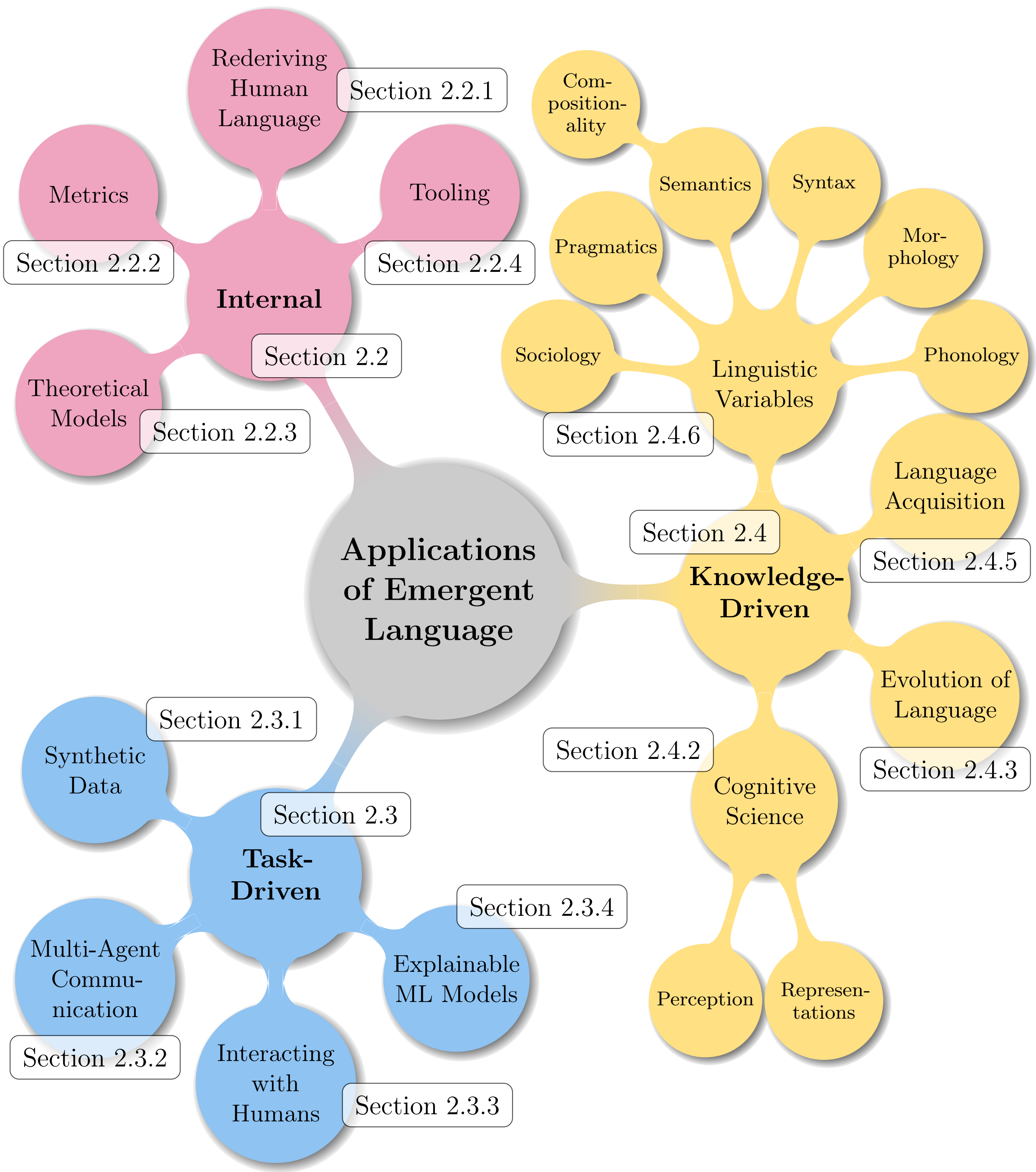
Thus, this work is a review of the most salient goals and applications of deep learning-based emergent communication research. We illustrate each of the applications by providing a description of its scope, an explication of emergent communication's unique role in addressing it, a summary of the extant literature working towards the application, and brief recommendations for near-term research directions. This work has three primary goals. (1) This work is meant to inspire future emergent communication research by compiling the most salient areas of research into a single document with relevant work cited. (2) It illustrates to practitioners outside of emergent communication the potential ways that emergent communication can be used in an easily-referenced format. (3) It define the ultimate aims of emergent communication, which is critical to guiding the field of research through practices like establishing evaluation metrics and benchmarks. Evaluation metrics require explicitly defining what a *good* or *desirable* emergent language is, and understanding what emergent communication can be used for is a foundational step in their development.

2.1.1 Scope

In order to effectively select papers for the review, we need to define particular scope of "emergent communication" that we are dealing with. We are not claiming that work excluded by these criteria is unimportant or unrelated to the included work, nor are we arguing that these criteria should be viewed as normative. Rather, these criteria are merely intended to be sufficient for conducting a complete, coherent review of a field of research. The scope of this review specifically comprises the following criteria:

- *Necessarily*, the topic is an agent-based computer model, that is, the simulation of individual computer agents in an environment.
  - *Typically*, it uses reinforcement learning.
  - *Necessarily*, it is not simply the result of training a model on human language data (e.g., emergent properties of pre-trained language models do *not* qualify).
  - *Typically*, the system contains multiple agents (e.g., an agent talking to itself could still qualify).
- *Necessarily*, the agents have a communication channel.
  - *Necessarily*, the communication is analogous to human language in some way.
  - *Typically*, the communication channel is discrete symbols (i.e., analogous to words or subword units).
  - *Sometimes*, the communication channel may be continuous (e.g., analogous to speech sounds), but the structure of the channel or the resulting protocol must be of interest (i.e., an unconstrained, unstudied continuous channel does *not* count.)
- *Necessarily*, the exact nature of communication (e.g., the structure or content of the protocol) is not determined ahead of time; it "emerges" from simpler characteristics of the environments and agents.
- *Necessarily*, the approach uses deep learning methods.

Figure 2.1: Structure of the applications discussed in this review.



– *Typically*, methods use neural networks optimized by gradient descent.
– *Typically*, work is associated with the communities of ICLR[2], NeurIPS[3], and ICML[4] conferences and the EmeCom workshop[5].

2.1.2 Related work

This section briefly discusses some closely related areas of research that fall outside of the scope of this paper. Although the goals and applications of these research areas are relevant to those discussed in this paper, we do not incorporate these into this paper in the interests of length. While their applications are very similar to those of deep learning-based emergent communication, the particular issues, methods, and possibilities which deep learning techniques present are quite different from these related areas.

Emergent communication Lazaridou et al. (2020) offer a general review of emergent communication research. It covers the same body of literature as this paper but with a general scope. Readers unfamiliar with the field of emergent communication would benefit greatly by reading this review first as it covers the essential elements of the field (background, methods, results, related work, etc.). This paper, on the other hand, focuses specifically on the goals and applications of emergent communication research.

A few other position papers have been published on emergent communication and share this paper's goal of guiding future work through a direct analysis and discussion of the literature. LaCroix (2019), Moulin-Frier et al. (2020), and Galke et al. (2022) synthesize linguistic research on the evolution of language with contemporary methods in emergent communication, highlighting what aspects do not line up and how emergent communication research might change its approach. Finally, Zubek et al. (2023) provide a more robust critique of current methods in emergent communication from various linguistic perspectives.

NLP and multi-agent RL Some areas of natural language processing focus on learning human language by leveraging deep multi-agent reinforcement learning in a way similar to emergent communication. This includes approaches like Lee et al. (2019) and Cogswell et al. (2020) which use multi-agent dialog grounded with visual referents or Lu et al. (2020) which uses iterated learning framework for tuning dialog agents. Although the methods are similar, these approaches typically do not care about communication systems that are emerging from scratch and instead focus directly on improving performance with human languages.

Emergent language without deep learning Computer simulations of the emergence of language are also possible without recourse to deep learning methods. Simulations along these lines might use other forms of machine learning or simply mathematical models of agents and environments. For example, Werner et al. (1991) simulate the emergence of a communication system in a population of mating animals. Female animals guide the male animals towards

[2]https://iclr.cc/
[3]https://neurips.cc/
[4]https://icml.cc/
[5]https://sites.google.com/view/emecom2022

them by emitting discrete messages. Each agent is implemented as a connectionist artificial neural network which is optimized with a genetic algorithm. Another instance is Kirby (2000), which verifies the possibility of compositional communication emerging without biological evolution. Specifically, it presents a mathematical model of an agent population where new members must learn to communicate from older members (who eventually die off). This is implemented as a computer program which can empirically verify the hypothesis.

Although this research area has significant overlap in terms of goals, the methods have significantly different challenges. In particular, methods not based on deep learning tend to have strong inductive biases which constrain the range of languages that can emerge. In contrast, one of the main challenges of deep learning-based emergent communication is trying to find the environmental pressures and functional advantages which shape language in place alongside the weaker inductive biases of deep neural networks.

Emergent communication with humans Research on the emergence of human language also takes place outside the context of computer simulation altogether. Experimentally, small-scale studies can be done with humans in the laboratory. For example, Kirby et al. (2008) test the emergence of structure in language from language transmission dynamics by having humans serve as the agents in a laboratory experiment. Observationally, there are recorded instances of a full human language emerging as in the case of Nicaraguan Sign Language, where deaf children with minimal prior linguistic knowledge developed language when placed together in a school environment (Kegl et al., 1999). Despite the relevance of this research to deep learning-based emergent communication, the challenges of human-base studies diverge significantly from those based on machine learning.

Symbol emergence in robotics Taniguchi et al. (2015) survey a research area called "symbol emergence in robotics" (SER). SER is concerned with developing autonomous robots with the ability to discover meaning and communication skills from sensory-motor experiences with humans and other robots. In this way, both SER and emergent communication study "bottom-up" methods of autonomous agents acquiring the ability to use language in a deep, embodied way. SER is more concerned with the development of robotic agents which can dynamically learn to interact like humans through pragmatic and social facets of language. In contrast, emergent communication is more concerned with observing the entire process of language creation in virtual environments through agents interacting with other agents.

2.1.3 Structure of review

We divide the applications of emergent communication into three broad categories:

- *Internal goals* (Section 2.2) aim towards improving the technique in its own right. In a sense, these goals are the "basic research" of emergent communication.
- *Task-driven applications* (Section 2.3) aim at solving a well-defined problem. These goals generally correlate with goals in the domain of engineering such as those found in NLP.

- *Knowledge-driven applications* (Section 2.4) aim at increasing the knowledge of some phenomenon. These goals generally correlate with the goals of sciences such as linguistics.

We illustrate each application with four sections which each answer a question:

- "Description": What exactly is the problem being solved?
- "Applicability": How do the techniques of emergent communication (in practice or in theory) uniquely address this problem?
- "Current state": What progress has been made in the literature toward addressing this problem?
- "Next steps": What does the next important research paper towards this application look like?

We give a brief of analysis of the trends we found in reviewing the literature in Section 2.5 before concluding in Section 2.6. Details of the review process are presented in Section A.1, and a complete list of works surveyed is given in Section A.2.

2.2 Internal Goals

Internal goals are not what we would typically consider applications at all since they are focused on issues internal to the field of emergent communication. Nevertheless, these applications are important because they are (1) prerequisites for applying emergent communication to other areas and (2) the primary contributions of many papers that reference these goals. While, in a sense, any contribution could be considered an internal goal, we choose to address the internal goals which represent the clearest and most salient waypoints within emergent communication research.

2.2.1 Rederiving human language

Description Aiming to create emergent communication that resembles human language is a central characteristic of emergent communication and drives much of the research on the topic. This resemblance can include everything from low-level traits like compositional semantics and tree-like syntax to high-level traits like implicature (pragmatics) and sociolects (sociolinguistics), although how exactly to define this resemblance is an open question within linguistics. Aiming for resemblance does not necessitate exact replication of human language (or even having an exact definition of it): within human language we see a large amount of variation upon fundamental commonalities (e.g., distinguishing between nouns and verbs, having distinct units of meaning which appear in many contexts). *Rederiving human language*, then, is the process of developing the conditions (e.g., environment, agent architecture, games) which produce emergent communication which resembles human language.

Rederiving human language distinct from, although related to, *Origin of language* (Section 2.4.3) and *Language acquisition* (Section 2.4.5). Research into the origin and acquisition of language has a primary interest in the specific historical, environmental, and cognitive contexts of humans and their use of language. In contrast, rederivation is only concerned with these contexts for their instrumental value in developing emergent communication which is similar to human language.

Applicability The resemblance of emergent communication to human language is the nexus of most other goals in the field: other goals will either work toward resemblance in some respect or derive their effectiveness from resemblance to human language (or both). This resemblance to human language need not be perfect—even a partial rederivation of human language could still support many important downstream applications.

Internal Goals (Section 2.2) primarily work toward the rederivation of human language, while the task- and knowledge-based applications primarily derive their effectiveness from the rederivation. In particular, *Task-Driven Applications* (Section 2.3) rely on emergent communication having: structural similarities to human language (*Synthetic language data* (Section 2.3.1)), generalizability to new situations (*Multi-agent communication* (Section 2.3.2)), discourse structure (*Interacting with humans* (Section 2.3.3)), and the capacity to externally represent internal states (*Explainable machine learning models* (Section 2.3.4)). *Knowledge-Driven Applications* (Section 2.4), for example, rely on emergent communication resembling human language in terms of: cognitive processes influencing linguistic behavior (*Language, cognition, and perception* (Section 2.4.2)), macro-scale social processes (*Origin of language* (Section 2.4.3) and *Language change* (Section 2.4.4)), mechanism of learning and acquisition (*Language acquisition* (Section 2.4.5)), and general structure at every level (*Linguistic variables* (Section 2.4.6)).

Current state No work in the current body of literature has explicitly pursued the rederivation of human language. There are a large number of papers that study aspects of making emergent communication more human language-like in isolation (almost any paper in *Linguistic variables* (Section 2.4.6) does this in some capacity), but no papers have made steps towards rederivation holistically. While studying just one aspect of emergent communication at a time yields more tractable research questions, the risk is that isolating individual aspects misses the ways in which emergent communication is truly an *emergent* phenomenon within a complex system (Bar-Yam, 2002, Sec. 1.3). Complex systems are characterized by non-obvious interactions among the many moving parts, and taking away single elements of the system might change the behavior in significant, unpredictable ways. To the extent to which this is true, studying isolated phenomena in simple environments has limited potential.

For example, Ren et al. (2020) show, in line with established experiments with mathematical models (Kirby et al., 2007) and human subjects (Kirby et al., 2008), that the imperfect transmission of language from generation to generation (i.e., *iterated learning*) can explain a bias toward compositionality in communication system without further agent-internal biases. Yet empirical investigation of compositionality in emergent communication literature often uses fixed-population environments[6]. The fact that iterated learning has diverse support as

[6] I.e., environments where the set of agents remains constant throughout the training process. Contrast this with dynamic populations where newly-initialized agents enter the population and older agents leave the

an explanation for compositionality calls into the question the results of compositionality research which does *not* take iterated learning into account, since iterated learning could be a sufficient driver for compositionality in emergent communication, outweighing other potential sources like model capacity (Resnick et al., 2020) or perception (Lazaridou et al., 2018).

Next steps The rederivation of human language in full is a massively complex task which may be impossible in practice or even in principle. Yet even if it is possible only to a limited degree, emergent communication can still fulfill many of its applications. The first step toward rederiving human language is laying down the theoretical foundations: identifying the most salient properties of human language and using these to develop a concrete problem definition of "rederiving human language". The field of linguistics will be especially important for formulating precise notions of "rederiving human language". Such formulations will provide the groundwork for identifying the technical issues with rederiving human language through emergent communication techniques. For example, we speculate that: language will need to be processed by larger neural networks with parameter counts in the billions; agents will also need to have realistic cognitive constraints on producing and understanding language; populations of agents will have to number in the hundreds to mimic even the smallest human language communities; environments will need to be scaled up in terms of both sensory input (e.g., 3-dimensional environments, embodiment) as well as task complexity (e.g., involving multi-step planning); and many advanced techniques from deep reinforcement learning will need to incorporated into the optimization process in order to learn from richer environments (e.g., efficiently learning representations, planning, multi-agent cooperation).

2.2.2 Metrics for emergent communication

Description A metric, for our purposes, is a well-defined method for quantifying a property of or notion about an emergent communication system. Some properties in emergent communication are fairly concrete and are naturally quantitative such as vocabulary size or task success rate. Other properties are more abstract and there is not single, obvious way to quantify them (i.e., they are underspecified in some capacity). For example, compositionality often refers to the idea that "the meaning of a composite message is a function of the meanings of individual parts", but this definition is underspecified. It does not specify if "meaning" rests in the interpretation of the speaker, listener, or both, nor does it specify what limits might exist on functions used to combine meaning—each interpretation would be quantified differently and may be useful in different contexts. Finally, *evaluation* metrics are even more abstract as they try to directly measure how *good*, *useful*, or *desirable* something is in a general sense. For example, F-score is an evaluation metric for classifiers; that is, a better classifier should have a higher F-score (insofar as F-score is an effective evaluation metric), and generally speaking, a classifier with higher F-score will be more useful than one with a lower F-score.

Thus, developing metrics within emergent communication comprises a number of different tasks, including: designing precise formulations of abstract properties, developing

population.

practical computational methods for implementing these formulations, and demonstrating mathematically and empirically that they accurately quantify the particular property.

Applicability Metrics, in general, are a ubiquitous part of research in most any area of science or engineering. They are integral to formulating testable hypotheses since they delineate precisely what is being considered empirically (or theoretically). They are also what enables effective summarization and statistical analysis of the results of experiments beyond mere qualitative analysis. Together, these two factors make principled comparison with prior work possible. Evaluation metrics, in particular, help identify approaches to a given problem are most effective. These are especially important for the long-term development of a field as they help gauge overall progress and direct efforts towards the most promising approaches.

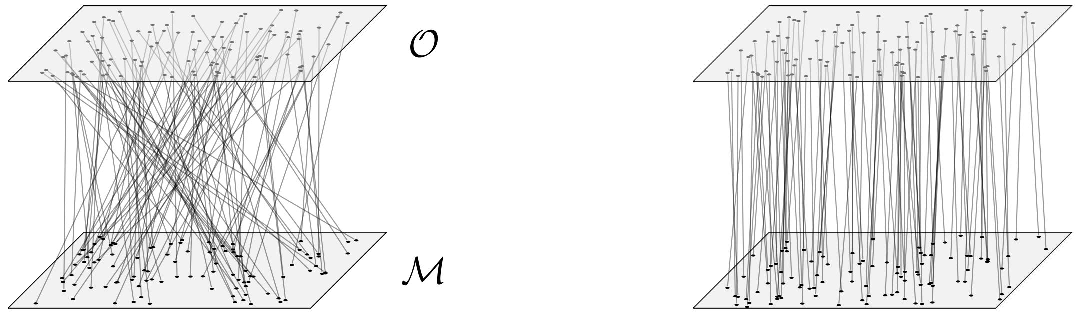
Current state (compositionality) Metrics for compositionality and generalizability comprise the lion's share of literature on this goal while only a few have been developed for other properties. This corresponds with the most common goals of emergent communication papers which are to develop emergent communication which has compositional semantics and generalizes beyond the scenarios seen during training.

Compositionality (or compositional semantics) refers to the general principal that utterances with complex meaning derive their meaning from a combination of the meaning of the components of the utterance (e.g., a "red car" is a car that is red). This is in contrast to "holistic" communication where there is no relationship between the meaning of an utterance and its components.[7] The most popular metric for compositionality is topographic similarity (Brighton et al., 2006; Lazaridou et al., 2018), which quantifies compositionality as the degree of correlation between distances in the referent feature space and distances in the message space (illustrated in Figure 2.2). Specifically, Lazaridou et al. (2018) use the Spearman's rank correlation coefficient (ρ) on the pairwise distances between objects in the feature space (quantified with cosine similarity) and their corresponding messages (quantified with Levenshtein distance). In this sense, toposim is more of a family of metrics since the precise methods of computing correlation and distance in the object and message spaces have a number of concrete realizations.

Representation similarity analysis (Kriegeskorte et al., 2008; Rodríguez Luna et al., 2020) takes a similar approach to quantifying compositionality but measures the correlation in the feature space and agents' internal representations. A handful of other metrics fall under the umbrella of *disentanglement*, where components of the message specify single attributes and do so independent of context. Such metrics include positional and bag-of-words disentanglement (Chaabouni et al., 2020), context independence (Bogin et al., 2018), and conflict count (Kuciński et al., 2020). Tree reconstruction error takes a deeper look at the compositionality of language by measuring how closely an explicitly compositional model of semantics can approximate what the emergent communication agents produce (Andreas, 2019).

In response to the amount of research into measuring compositionality, some papers have provided deeper analyses of metrics of compositionality. Korbak et al. (2020) provide a

[7] For example, a "black swan" can refer (idiomatically) to a rare event—something that is neither black nor a swan.



(a) Mapping with low topographic similarity; low correlation between spaces.

(b) Mapping with high topographic similarity; high correlation between spaces.

Figure 2.2: Illustration of two spaces with different topographic similarities (*toposims*). $\mathcal{O}$ and $\mathcal{M}$ represent embedding spaces for the observations and messages, respectively. A high toposim means that distances between any two points in the observation space correlates well with distances between corresponding points in the message space.

meta-analysis of the above compositionality metrics, showing that while many are sensitive to basic types of compositionality, most fail to recognize more sophisticated methods of composition. Kharitonov et al. (2020a) and Chaabouni et al. (2020) provide evidence against the claims that standard measures of compositionality are also measuring the ability of the language to generalize to describing novel objects.

Current state (other) Generalizability, as in other areas of machine learning, generally refers to the ability to perform well outside of the training conditions. Generalizability is most often operationalized as agents successfully describing objects previously unseen combinations of attributes (i.e., generalizing from training data to test data) (Korbak et al., 2019; Chaabouni et al., 2020; Denamganaï et al., 2020a; Kharitonov et al., 2020a; Resnick et al., 2020; Perkins, 2021a). Apart from this type of generalization, other work has looked at generalizing to new communication partners (Bullard et al., 2021), generalizing over different environments (S. Guo et al., 2021; J. Mu et al., 2021), and generalizing across linguistic structures (e.g., disentangled syntax and semantics) (Baroni, 2020).

Yao et al. (2022) introduce an evaluation metric, that is, one which measures the *overall* quality of an emergent language using data-driven methods. The metric equates the quality of an emergent language with the quality of machine translation from the emergent language to human language. The underlying intuition here is that the more human-like an emergent language is, the more effective substituting it for human language will be in machine learning tasks (i.e., using it as synthetic data, see Section 2.3.1).

Next steps With regard to metrics of phenomena like compositionality, it is critical to incorporate knowledge from linguistics as to how similar notions apply to human language. For example, with compositionality, emergent communication research often use to simple notions of compositionality, focusing individual units of meaning combining arbitrarily to form composite meanings. On the other hand, human language's relationship with compositionality is far more complicated, sometimes exhibiting it while sometimes being non-compositional (e.g., idioms, irregular word forms, grammatical rules limiting "acceptable" sentences). While this cross-disciplinary approach more difficult to incorporate into the research process, it is critical to the long-term trajectory of emergent communication research.

Evaluation metrics, on the other hand, are mostly absent in the emergent communication literature despite their importance to other fields of machine learning like reinforcement learning and natural language processing. Thus, it would be fruitful to develop true evaluation metrics which can determine what emergent languages are "best" or "most human language-like". As these notions are more abstract than "compositionality" or "generalizability", there is more theoretical groundwork that must go with the motivation of evaluation metrics in addition to the engineering efforts in actually designing and implementing them.

2.2.3 Theoretical models

Description A theoretical model of emergent communication is a mathematical or formal system which describes the behavior of an emergent communication system. Generally speaking, a theoretical model will describe a relationship between two or more variables in an emergent communication system. Theoretical models are developed in conjunction with empirical work and represent a refinement and systematization of the knowledge gained from these experiments. Most importantly, their formal representation allows rigorously reasoning about the behavior of a systems without needing to directly run experiments.

Applicability Theoretical models benefit emergent communication research primarily in two ways: they clarify research methods and can predict a system's behavior in compute-intensive situations. For research methods, using a theoretical model to phrase a research question results in a hypothesis which is clear and testable. As a result, the empirical evaluation has a clear relationship with the assumptions and structure of the model, allowing subsequent research to more easily build on previous work. In the absence of theoretical models and their hypotheses, papers must often rely on qualitative hypotheses which are difficult to empirically verify or result in merely pointing out "interesting" observations from the experiments. For this reason, employing theoretical models can move emergent communication research towards systematically scientific investigation instead of less organized trial-and-error.

Second, theoretical models provide a way to predict the behavior of emergent communication systems in situations where directly running the system is computationally expensive. The applicability of theoretical models on this front is discussed in the context of GPT-4 (OpenAI, 2023) and its scaling laws where the extreme computational cost of training the model made it critical that the designers could predict the behavior of the full-scale model ahead of time. In particular, they fit a mathematical equation predicting the loss based on computational input using smaller models. This gave the developers a way to accurately predict the final loss of the full-scale model at a fraction of the computational cost. As emergent communication environments get more complex with more design choices, hyperparameters, and computational cost, it will also be important to be able to predict the behavior of the system without having to run the full environment in every situation.

[8] τ is the Kendall correlation coefficient of the points.

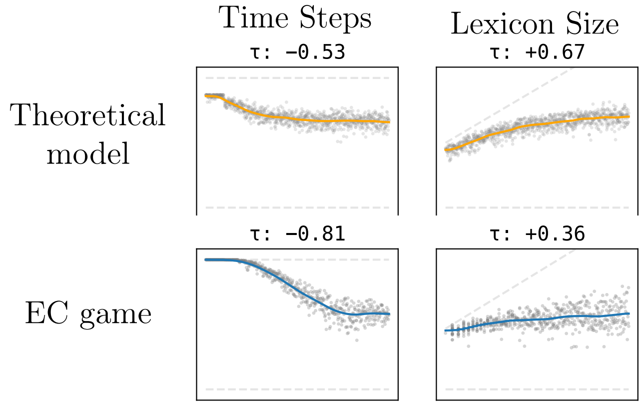


Figure 2.3: Plots of lexicon entropy (y-axis) versus time steps and lexicon size (x-axes) comparing a theoretical model against empirical measurements on navigation game with emergent communication (from Boldt et al. (2022b)).[8] Theoretical models can help predict the outcomes of emergent communication games much more efficiently than simply running the environment while also providing a conceptual understanding the environment's behavior.

Current state Only a handful of papers in the literature use theoretical models, and these models are usually not employed in any subsequent papers. Khomtchouk et al. (2018) study the transition between two degenerate "phases" of language: single-symbol systems and full one-to-one systems, with the synonymy and ambiguity found in human language lying in the middle. This model is then tested with a pair of simple reinforcement learning-based agents. Ren et al. (2020) apply the iterated learning model (Smith et al., 2003) to deep learning-based emergent communication; they use a formal probabilistic model of an iterated learning algorithm to express hypotheses which are empirically tested. Boldt et al. (2022b) formulate a stochastic process which describes the entropy of an emergent language's lexicon based on a handful of hyperparameters of the agent's neural network; the predictions of this model are also empirically tested in four simple emergent communication environments. Rita et al. (2022b) analyze emergent communication environments based on the Lewis signaling game by providing a mathematical decomposition of the loss function. This decomposition explains the different overfitting pressures hidden in the loss function; from this, they suggest measures to counteract such pressures which result in more compositional emergent communication.

Finally, the model presented in Resnick et al. (2020) is a good representative of theoretical models in emergent communication and their attendant difficulties. The model describes the relationship between the capacity of an agent's neural networks and the compositionality of the learned emergent language: if the capacity is too low to capture the regularities in the language (i.e., grammatical rules) the agents "underfit", and if the capacity is too high, the agents "overfit" by simply memorizing individual utterances in the language. The model predicts the compositionality to be low in both the under- and overfitting regimes and higher between them where the neural network learns regularities without memorizing individual examples. The model could then be formalized as follows

$$\text{Capacity}(M_X) \in [C_L, C_U] \land \text{Capacity}(M_Y) \notin [C_L, C_U] \Rightarrow \text{Comp}(M_X) > \text{Comp}(M_Y) \quad (2.1)$$

where M_X and M_Y are the agents' underlying models, $\text{Capacity}(\cdot)$ quantifies a model's capacity, C_L and C_U are the under- and overfitting thresholds respectively, and $\mathrm{Comp}(\cdot)$ quantifies the compositionality of the model's emergent language.[9]

The precise formulation of the model results in a clear hypothesis based on the predictions of the model, allowing the experiments to more directly test the underlying principles of the model. The difficulties that persist, though, are that the model's formulation and its predictions still lack precision. In the formulation, Resnick et al. (2020) do not fully articulate what constitutes "capacity"; a notion of capacity though would be extremely difficult if not impossible to formulate precisely for deep learning models. In the predictions, the paper is only able to articulate general trends and correlations rather than predicting exact values or distributions. These issues, though, are representative of more general issues with theoretical models in emergent communication the use of deep neural networks and reinforcement learning make precision inherently difficult (although approximation is not impossible as shown by the GPT-4 scaling case above). Finally, despite the fact that the model in Resnick et al. (2020) addresses compositionality, the most popular topic within emergent communication, it does not see reuse subsequent papers.

Next steps Theoretical models are difficult to apply deep learning-based emergent communication since deep neural networks themselves are difficult to formalize. Part of this difficulty is inherent while some of it stems from the sparsity of formalization in the applications of deep neural networks. Thus, an important next step would be to address these difficulties with archetypical examples of how theoretical models can be applied to emergent communication as well as recommendations for best practices, taking inspiration from existing work on the theoretical foundations of deep learning and complex systems.

Even given these difficulties, one-off instances of simple theoretical models can be helpful in clarifying the contributions, hypotheses, and results of a given paper. For example, instead of hypothesizing simply that changing X will improve Y, it could be stated instead that there will be a positive correlation between x and y, where x and y are quantitative metrics of X and Y respectively, and correlation is mathematically defined (e.g., Spearman's rank correlation coefficient). This would facilitate experiments which more clearly refute or a support a hypothesis and its underlying claims.

2.2.4 Tooling

Description The central aim of tooling within emergent communication is to develop apparatus that can be used to ease the process of implementing and running experiments. Since emergent communication is under the broad category of computer science, the experimental apparatus are most often programs, their source code, and sometimes datasets. Although any codebase used for an emergent communication experiment can be reused and repurposed by other researchers for new experiments, codebases which are designed to be reused for a broad range of experiments are the focus of this application.

[9] This is a summarization of the model which is more precise in its original formulation. The particular formalization is not used in the original paper and is instead derived from Boldt et al. (2022c).

Applicability The most obvious benefit of shared and standardized tooling is that it saves time for researchers as less time needs to be spent reimplementing the basic features of emergent communication experiments. Furthermore, the incidence of bugs decreases, implementation efforts can be spent improving existing tooling, and comparison across papers is more reliable since more implementation details will be the same. Special care must be taken, though, that the implementation details do not lead to systematic biases in experiments; emergent communication is especially susceptible to this concern since it is difficult to distinguish between effects of structure of the environment (abstractly speaking) and effects of implementation details. Finally, well-designed and easy-to-use tooling is a significant help to emergent communication researchers who do not have a strong software development background. The task of putting ideas into code is much more difficult for such researchers, and decreasing the amount of unnecessary reimplementation can greatly improve their ability to contribute to the field of emergent communication.

Current state Tooling for emergent communication has a small degree of standardization, although the high degree of variety in problems and approaches in the field decrease the practicality of a one-size-fits-all framework. EGG (Emergence of lanGuage in Games) (Kharitonov et al., 2019) is the most widely used framework for deep learning-based emergent communication; it provides a simple Python programming interface for some of the most common emergent communication games, agent architectures, and metrics. Papers which implement new games using EGG further expand the range of games and metrics which are easily accessible through the framework including: Chaabouni et al. (2019a), Dessì et al. (2019), Chaabouni et al. (2020), Auersperger et al. (2022), and Kharitonov et al. (2020b). ReferentialGym (Denamganaï et al., 2020b) is similar to EGG in scope, although it has seen less reuse within the literature. Other tooling may target more specific aspects of experiments in emergent communication. For example, TexRel (Perkins, 2021b) is a synthetic dataset designed specifically for use in emergent communication games; in this case, data (images) are constructed such that *compositional* language could aptly describe them. Additionally, Ikram et al. (2021) introduce HexaJungle, a suite of environments for studying emergent communication. For papers which explore beyond the typical environments (which is a significant portion), it is common to implement the emergent communication game and agents from scratch using more general purpose tools like PyTorch (Paszke et al., 2019) (Evtimova et al., 2017; J. Mu et al., 2021; Noukhovitch et al., 2021).

Next steps The next steps for tooling in emergent communication largely depends on what tasks, problems, and methods receive the most attention going forward. If the field continues to study similar environments, EGG could continue to support such work, but if radically new environments or experimental paradigms appear, current tooling might prove insufficient. In part, this is due to an inherent trade-off between the flexibility of a framework and its convenience; while emergent communication is rapidly changing, the required flexibility often does not provide much convenience, but as the field focuses on fewer problems, frameworks could play a greater role. A possible middle way between these two issues would be developing an interface (in the sense of object-oriented programming) for emergent communication environments similar to OpenAI Gym (Brockman et al., 2016), which can provide some standardization and interoperability while not impeding novel environments and implementations.

2.3 Task-Driven Applications

The task-driven applications of emergent communication center around fields of engineering such as machine learning, natural language processing, and multi-agent systems, and typically involve solving a well-defined, practical problems. These applications have the most immediate impacts and, as such, offer some of the most convincing motivations for developing emergent communication techniques in the short term. The primary challenges in this area come from competing against more established methods in deep learning which are continually advancing through larger and larger scales of data and compute.

2.3.1 Synthetic language data

Description In the context of deep learning, synthetic data refers to data which is generated with a computer program; this is in contrast to "real" or "natural" data which is collected from an actual system being studied. For example, the language we find in books, conversations, speeches, etc. would all be real, in this sense, whereas a corpus of sentences generated by sampling from a probabilistic context-free grammar would be synthetic. For example, within NLP, synthetic data can be used for transfer learning (Papadimitriou et al., 2020; Mirzaee et al., 2022) and model probing (Lake et al., 2018; Y. Zhang et al., 2022). Synthetic data has a number of advantages when applied to deep learning; this includes: controllability, availability in arbitrary quantities, availability in low-resource domains (e.g., endangered languages, multi-modal settings), alleviating concerns about bias, and alleviating privacy concerns (since data is not collected from humans, e.g., through surveillance). Although synthetic data finds niche uses alongside real data in deep learning and NLP, it fails to have widespread applicability because it often does not capture the plethora of nuances and irregularities that appear in real data, that is, the "long tail" of the real data distribution. In natural language, this can manifest as unnatural but valid syntactic structures, uncommon senses of words, idioms, and wordplay.

We consider all kinds of pretraining, data augmentation, analyses, and evaluation of deep learning models with emergent communication as part of this application even if it does not involve generating synthetic datasets *per se*. For example, you might use a trained emergent communication agent itself to pretrain or evaluate a model instead of generating an intermediate dataset.

Applicability Emergent communication could serve as a way to generate synthetic language data which more closely mimics the natural variation found in human language. The distribution of patterns within natural language has a "long tail" insofar as a large proportion of the total mass comprises a large number of infrequent patterns, making it very difficult for something like synthetic data generated by handcrafted programs to sufficiently replicate the distribution (Naik, 2022). This is illustrated by the history of NLP: handcrafted expert systems have been surpassed by learning-based method which can scalably leverage computing power to mine patterns from increasingly large quantities of data. Emergent communication, rather than mining patterns directly from data, seeks to uncover linguistic and behavioral patterns which are latent in the communicative pressures of embodied multi-agent environments. Work such as Artetxe et al. (2020) demonstrates that deep neural networks do learn latent, language-agnostic patterns from their training data; this suggests that even if an emergent language does not have a one-to-one correspondence with some particular human language, having underlying structural similarities with human language would be sufficient to still be useful.

Current state Work towards using emergent communication to generate synthetic data has been at the proof-of-concept level. The papers in our survey (discussed below) showed that emergent communication could indeed improve the performance of neural NLP models when used for pretraining in very low-resource settings. That being said, experiments only cover a narrow selection of datasets/tasks and do not rigorously compare against alternative methods (e.g., traditional synthetic data, cross-lingual transfer). As a result, is difficult to gauge the practical impact of the proposed methods.

Yaoyiran Li et al. (2020) pretrain encoder-decoder few-shot machine translation models with an emergent communication signaling game; in addition to finding improvements in very low-resource settings, the experiments showed that the task success rate in the emergent communication game was well-correlated with the downstream BLEU score. Downey et al. (2023) also tackle machine translation, but instead use an emergent communication game to fine tune a multi-modal model for unsupervised machine translation, finding that emergent communication is more effective than the back-translation baseline. Yao et al. (2022) take a slightly different approach by using the emergent communication game only to generate a synthetic corpus (instead of training the models directly); this corpus is then used to pretrain models for language modeling and image captioning tasks. The experiments compare emergent language corpora against two baselines: Spanish and a synthetic dataset generate by sampling delimiters from a Zipfian distribution to create a hierarchical language with similar structural biases to human language (e.g., `{<()>[]()}`). For the lowest data regimes, pretraining the model on emergent language corpora reliably outperforms models pretrained on the baseline datasets. Finally, Y. Mu et al. (2023) use emergent communication to pretrain an instruction-following embodied control model (e.g., for controlling a robotic arm); the experiments showed that not only does the proposed method outperform the baseline models but also that the emergent language is more effective as training data than pre-trained, static representations derived from video demonstrations.

Next steps The first direction is thoroughly investigating the different ways emergent communication can be used for generating synthetic data. Yaoyiran Li et al. (2020) (using emergent communication agent models directly downstream) and Yao et al. (2022) (using emergent language corpora for pretraining downstream models) take different approaches to the same task of pretraining downstream NLP models. These approaches have different relative merits (e.g., making better use of training data versus decoupling agent architecture from downstream architecture, respectively), and there are likely more ways to approach the same problem with emergent communication. Thus, next steps would consist of finding other promising methods of harnessing emergent communication for model pretraining and comparing these approaches on a common ground. Determining which of the approaches is best is critical to giving emergent communication the best chance of surpassing more traditional methods model pretraining and generating synthetic data.

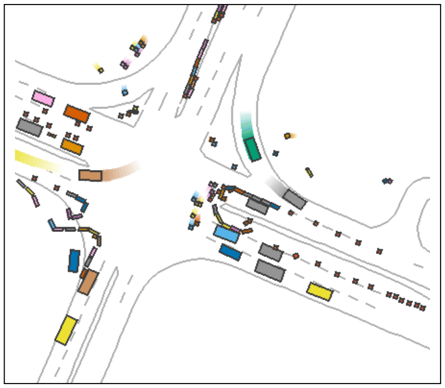
The second direction which can be pursed after or in parallel to the first is rigorously comparing emergent communication for pretraining neural NLP models with more established techniques like cross-lingual transfer and traditional synthetic data (Artetxe et al., 2020). First and foremost, this helps to establish whether or not emergent language data can truly surpass what is already present in the field. In particular, comparison against cross-lingual transfer should highlight how emergent language data is more available, that is, it can be attained in higher quantities with more relevance to the target language than cross-lingual data. Comparison against traditional synthetic data could tease out exactly what properties of emergent communication make it more effective in downstream applications. For example, emergent communication could be compared against increasingly complex synthetic languages: balanced parentheses, context-free grammars, then full-scale grammars (e.g., head-driven phrase structure grammar (Pollard et al., 1994)).

Both directions would entail developing a sort of benchmark for testing the effectiveness of pretraining methods. This would require not only finding suitable data sources and evaluation metrics, as usual, but also determining how to make the variety of methods for pretraining comparable. For example, emergent communication is more computationally expensive than traditional synthetic data and standard NLP pretraining methods, yet it could surpass synthetic data in quality and real data in low-resource settings. Therefore the benchmark would have to take into account data and computational requirements in addition to raw performance.

2.3.2 Multi-agent communication

Description The area of multi-agent communication is concerned with autonomous (computer) agents coordinating their actions through the use of a communication protocol. Most prototypically, this would apply to a team of autonomous robots working together but could also include situations like self-driving cars on the road (illustrated in Figure 2.4) or IoT devices on a local area network. The two typical approaches to developing multi-agent communication protocols are handcrafting them or learning them like a latent variable between agents. Handcrafted protocols (e.g., DHCP for network configuration) are typically well-suited for specific tasks but are also require significant expert design, which hinders much potential for open-domain or general purpose communication. Automatically learned continuous protocols (i.e., messages are learned continuous vectors) solve some of these issues but raise new issues related to deep, learned representations such as low interpretability. This task is distinct from autonomous agents communicating directly with humans, which we discuss in Section 2.3.3.

Applicability Emergent communication addresses these issues in three main ways. First, emergent communication is scalable to more general-purpose tasks since it is developed by computational processes directly from the functional pressures of the task it is applied to. Second, it is more interpretable insofar as it resembles the structure of human language,



(a) Complex scenario with various kinds of agents.

(b) Pixel-based input to approximate real-world diversity in observations.

Figure 2.4: Self-driving vehicles in complex traffic situations are an important application of multi-agent communication. Diversity in both scenarios as well as the observations themselves indicate that open-ended communication systems could be more appropriate than handcrafted protocols where all scenarios are anticipated ahead of time. Screenshots from documentation of MetaDrive (Q. Li et al., 2023) (Apache-2.0 license).

for example, using discrete symbols in its communication channel or having a hierarchical syntactic structure (cf. continuous vectors which do not resemble human language and require mathematical transformation to be analyzed). Finally, human language is the gold standard for communication protocols insofar as it can apply to previously unseen situations and is robust to noise and other hindering factors. Thus, developing communication protocols which deliberately mimic the structural properties of human language could be a way to better attain these desirable functional properties.

For example, the following design elements of an emergent communication system could contribute to recreating some of the above desirable properties of an emergent language. To encourage general purpose language, we can start with an open world, open-ended environment (e.g., Minecraft) and/or one with many distinct situations (e.g., Starcraft, Dota 2). Furthermore, tasks which have adversarial components can especially elicit a diversity of situations since one team of agents is constantly trying innovate to outcompete the other. Towards interpretability, the agents could be constrained to communicate only with discrete symbols at human-scales (e.g., modest vocabulary size and message length). Finally, elements like communication channel noise or constantly cycling out agents in the population can induce a more robust communication protocol since agents cannot as easily overfit to each other.

Current state Work on developing multi-agent communication protocols has experimented with a handful of environments and scenarios but has not established any one task as being definitively helped by emergent communication. Many of the explored environments are a variation on navigation (Mul et al., 2019; S. Li et al., 2022; Masquil et al., 2022) or the signaling game (Bullard et al., 2021; Cope et al., 2021; Y. Wang et al., 2022; Tucker et al., 2021), although some include more abstract environments like a coalition-based voting game (S.

Li et al., 2022) or semantic communication (Thomas et al., 2022). Emergent communication for multi-agent communication has been compared against competing methods, that is, handcrafted protocols (S. Gupta et al., 2020; S. Chen et al., 2022) and learned continuous communication (S. Li et al., 2022; Y. Wang et al., 2022). Although, these comparisons use the competing methods more as "baseline points of reference" rather than comparing them "head-to-head", where both methods are presented in their strongest forms so as to show the real-world superiority of emergent language-based communication. In most cases, increasing the performance of the multi-agent team is the primary interest of the experiments; additionally, papers have also looked at emergent communication's robustness to corruption and noise (Cope et al., 2021; Y. Wang et al., 2022) as well as the potential for communicating with partners not seen during training (Bullard et al., 2021; Cope et al., 2021).

Next steps The first direction of future work on emergent communication for multi-agent communication is to find a niche for emergent communication, that is, presenting a particular task where, in realistic conditions, emergent communication surpasses state of the art non-emergent approaches. Although emergent communication has intuitive advantages (discussed above in "Applicability"), it has yet to be shown in a real-world task. This is a significantly more difficult task than what most of the current literature accomplishes: namely demonstrating a on small scale that multi-agent communication is possible with emergent communication techniques as a proof of concept. Based on the particular advantages of emergent communication, such a task will likely have to be open-domain or demand continual adaptation, rendering hand-crafted protocols impractical, while also needing some element of interpretability, demonstrating an advantage over learned continuous communication. This is a formidable task as presenting effective definitions of "open-domain" and "interpretable" require formalizing rather abstract notions.

In conjunction with this first direction, it will also be necessary to empirically verify the intuitions that (1) emergent communication is more interpretable than continuous communication, and (2) emergent communication's structural similarities to human language confer some actual functional benefit beyond continuous or unconstrained communication. If these intuitions are well-founded, then it will greatly expand the potential applications of emergent communication in multi-agent systems.

2.3.3 Interacting with humans

Description A perennial goal of computer systems has been more naturally interacting and communicating with humans. This is an incredibly difficult task due to the complexities of human communication ranging from nuanced syntax and semantics to pragmatics and conversational dynamics. While deep learning methods have had good success learning syntax and decent success learning semantics, proficiency at the level of pragmatics is not yet present because these higher levels of language and communication tend to be more difficult to learn from purely from text through a language modeling objective. This is demonstrated in Ouyang et al. (2022) by the fact that an InstructGPT model outperforms a GPT-3 model 100× larger when it comes to following a human's instructions (as evaluated by humans). They observe from this that the language modeling objective alone is misaligned with the objective of "follow the user's instructions helpfully and safely"; for example, truthfulness

(a) User interface for interpreting the meaning of an emergent communication method from a visualization of the message's embedding (Tucker et al., 2021).

(b) Example of a "message" from a sketch-based game (from documentation of Mihai et al. (2021a)'s code).

Figure 2.5: Two examples of agent-to-human emergent communication interfaces.

is one dimension that is drastically increased by training on human feedback (Ouyang et al., 2022). Even with extensive training with human feedback, models like ChatGPT still significantly diverge from humans when it comes to pragmatics and communication strategies (Z. Qiu et al., 2023; B. Guo et al., 2023). Thus, despite large language models' fluency, they do not naturally capture critical aspects of interacting with humans, and current methods of addressing it entail relying directly on human supervision (Ouyang et al., 2022).

This application primarily refers to methods of interactively communicating in tasks like dialogue or human-robot collaboration. We distinguish this from creating explainable machine learning models which we address in Section 2.3.4.

Applicability The central argument for using emergent communication to better communicate with humans comes from the fact that an emergent communication agents naturally develop competency with a wide range of linguistic phenomena. The hypothesis here is that the same functional pressures that drive the pragmatic and social aspects of human language could be replicated by sufficiently rich and embodied emergent communication environments. Thus, the emergent communication agents could not only develop the syntax and semantics of the language but also pragmatic elements in response to the environmental and social pressures. In fact, Bisk et al. (2020) argue that embodiment and interaction, beyond simply modeling static corpora, are necessary for learning to use the full depth of language. Emergent communication, then, could be a more compute-driven (and less human-feedback intensive) way of imbuing machine learning models with a full range of linguistic competency that is necessary for seamlessly interacting with humans.

Current state Communicating with humans is an oft-cited potential application of emergent communication techniques, although few papers have directly experimented with it. The papers we found in the survey were proof-of-concept tasks which demonstrated some possible methods for emergent communication agents interacting with humans. One of the characteristic design choices of each paper is deciding how to structure the communication channel between the human and agents.

For the direction of human-to-agent communication, Tucker et al. (2021) map natural language to a joint embedding space with the emergent language, making the embedded natural language understandable to the agents. S. Li et al. (2022) have humans select embeddings directly from a labelled visualization of the embeddings (i.e., t-SNE) of emergent language messages. For the direction of agent-to-human communication, Tucker et al. (2021) visualize the embedding of agent messages in a labelled embedding space, allowing a human to determine which cluster of messages an unlabelled message belongs to (shown in Figure 2.5a). Mihai et al. (2021a) show the human directly with a "message" (i.e., sketch) in a sketch-based signaling game (shown in Figure 2.5b). Apart from direct human-agent interaction, Tucker et al. (2022a) demonstrate machine translation-based approach where human and emergent languages are aligned through an image captioning task.

Next steps The first direction for using emergent communication to augment human-computer interaction is to determine the most the natural and scalable methods and modalities for human and computers to communicate. The existing literature uses a handful of methods some of which are either unnatural and not scalable to more complex communication (e.g., interacting with concept/word embeddings). Work on non-emergent human-computer interaction can inform emergent communication research not only on methods of communication but also concerning what environments would have the potential for complex communication while still being simple enough to work with. For example, Narayan-Chen et al. (2019) present a collaborative building game in a Minecraft environment which could satisfy these criteria.

The second direction for this application is empirically demonstrating the intuitive advantages of emergent communication over more established methods for human-computer interactions. The pragmatics of interacting with humans is one of the areas with the most potential because pragmatics are inherently flexible, tied to extra-linguistic knowledge, and are more difficult to formalize than, say, syntax or semantics. Nevertheless, emergent communication could help in the more difficult regions of syntax and semantics, such as disambiguating utterances which rely on contextual knowledge or common sense reasoning.

2.3.4 Explainable machine learning models

Description Explainable machine learning models are those which can communicate to humans the reasons or factors behind a certain their decision. Such model are a response to deep, black-box neural models which may be able to make accurate decisions but often for opaque or seemingly arbitrary reasons. Instead it is desirable for explanations to be: (1) causally related to actual decision that is made (i.e., not a *post hoc* rationalization); (2) expressed natural language, which is one of the most effective ways to convey ideas to humans; and (3) not impose a significant negative impact on the performance of the model.

Two paradigms of explainable machine learning models illustrate solutions only satisfying some of the criteria. The first paradigm is using language generation models to generate explanations based on the hidden states of a model; while this permits the use of deep neural models, the explanations are decoupled from the actual decision since the explanation is superfluous with respect to the actual decision. The second paradigm is using explicit, interpretable steps in reasoning to the prediction (e.g., decision trees, knowledge graphs);

although these explanations are now causally efficacious with respect to the prediction, it restricts the complexity of model that can be used to make the prediction. While the explanations these models generate are intrinsically related to the decisions made (e.g., the weights of a regression both explain the decision and cause it), they restrict the complexity of the model, and hence, can hamper overall performance.

Explainable machine learning models, in some sense, is a subclass of *Interacting with humans* (Section 2.3.3); here the interaction is always focused on a machine learning model communicating accurate and interpretable explanations for its decision or behavior.

Applicability Emergent communication takes a radical approach to both the causal efficacy and the natural language aspect of explainable models. To illustrate this, we can describe a "deliberative ensemble of emergent communication agents". Such an ensemble would be posed a semi-adversarial game where first each member of the ensemble would generate an output for a given input. After this, the ensemble members would communicate in the emergent language to try to convince the other members of the particular output before aggregating the members' revised decisions. Given that emergent language is designed to resemble human language, the representation mismatch between natural language and the emergent language discourse is far less than natural language and the activations of a monolithic neural network. Furthermore, since the deliberation and communication among agents is critical in the final decision of the ensemble, the explanation has a direct causal link to the decision.

Current state Using emergent communication for creating explainable machine learning models has only seen proof-of-concept exploration in one series of papers. Namely, Santamaria-Pang et al. (2020), Chowdhury et al. (2020a), Chowdhury et al. (2020b), and Chowdhury et al. (2021) implement and experiment with a medical image classification model which, internally, is a Lewis signaling game (Lewis, 1969). This means that the internal representations are themselves the discrete messages of an emergent language. Messages-as-internal representations, here, are intended to be a more natural modality for human working with the system than, for example, the activations of intermediate layers in the neural network.

Next steps The first direction for using emergent communication for explainable machine learning models is exploring methods of generating explanations beyond the signaling game that we see in the current literature. The signaling game, while providing potentially interpretable messages, does not effectively exhibit the multi-step reasoning which (1) is most suited to the complex decisions which we would want explained, (2) is how humans generally explain themselves, and (3) is where emergent communication has the greatest potential to surpass more established methods. Such games or environments might incorporate incentives for agents to collaborate and reason sequentially using the emergent language. This reasoning process would then double as the basis for the decision and the explanation of the decision.

The second direction is incorporating state-of-the-art models into the emergent communication systems. This application, more so than others, requires that the emergent communication-based model perform comparably on downstream tasks to more established explainable machine learning models; even if the emergent communication-based models are highly explainable, they are of little practical use if they are not comparable in performance to traditional approaches. Given the size of current state-of-the-art models and inherent difficulty of training emergent communication models, this incorporation, in the near term, would likely be limited to leveraging pre-trained models which could be, at most, finetuned.

2.4 Knowledge-Driven Applications

The knowledge-driven applications of emergent communication center around the scientific fields of linguistics and cognitive science and typically concern gaining a deeper understanding of phenomena in the natural world. These applications have tend to have more remote impacts than the task-driven applications, but they also present the opportunity to gain novel insights into how humans think and use language. The primary challenges in this area come from creating emergent communication which is realistic enough to legitimately provide insight in areas where there are gaps left by more traditional techniques in linguistics and cognitive science. The first subsection below (Section 2.4.1) provides a summary of common themes in the “Description” and “Applicability” subsections throughout knowledge-driven applications (i.e., it is not itself an application).

2.4.1 General paradigm of knowledge-driven applications

Description Some of the most persistent debates in linguistics are about the degree to which language and its characteristics are the product of very specific biology (the “Chomskyan” nativist position that dominated North American linguistics in the second half of the twentieth century) or can be derived from very general mechanisms of learning (the behaviorist position that dominated North American linguistics in the first half of the twentieth century). This conflict reflects a broader debate within the social and behavioral sciences about the relative importance of “nature” (the inductive biases of the human brain) and “nuture” (operant conditioning from parents, caregivers, and other aspects of the environment) in the cognitive development of human children. Such debates are difficult to resolve because of limited access to the necessary data: the ingredients of language (nature and nuture) are largely fixed, meaning we cannot (ethically) vary them in order to determine their effects on language. This is to say, the relevant data in these debates come largely from observation and only extremely limited experimentation. The lack of true experimentation hinders the type of scientific investigation which would yield more definitive answers to these questions.

Applicability Emergent communication can address these unsolved problems by serving as a proxy for human language whose ingredients can be manipulated and experimented with. Emergent communication makes a suitable proxy because (1) it aims at being a faithful reconstruction of human language, and (2) this reconstruction is a reflection of its ingredients. For example, we can see the “nature vs. nurture” distinction paralleled in the distinction between the systems inside of an agent and the interaction that takes place with other agents.

Deep learning-based emergent communication is uniquely poised to serve as a proxy for human language for two reasons. First, deep learning methods are by far the closest methods to replicating human proficiency in language (as well as vision, planning, and so on). Hence, it would seem a model class of comparable power is necessary to support the emergence of a language with enough complexity to be useful for the most relevant linguistic problems. Second, deep neural networks also introduce minimal inductive bias when compared with traditional simulations and mathematical models. The behaviorist or "nurture" position can only be validated if language learning can take place without language-specific inductive biases and this is only possible in a context in which learning according to very general principles is possible, so deep learning is a natural fit for testing hypotheses about the necessity of language-specific learning mechanisms.

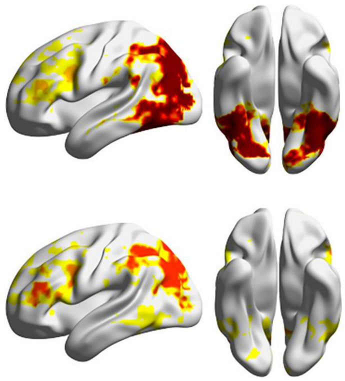
2.4.2 Language, cognition, and perception

Description This goal refers to the two-way relationship between language and cognitive (and perceptive) processes in the human brain: how language is shaped by the cognitive capacities of humans and what goes on in the brain to enable the use of language. By extension, this also includes behavior which proceeds from cognitive phenomena of interest (e.g., adjusting communication strategies based on a theory of mind). Aside from not being able to experimentally modify the brain, a major barrier in studying cognition is being able to merely *observe* the brain.

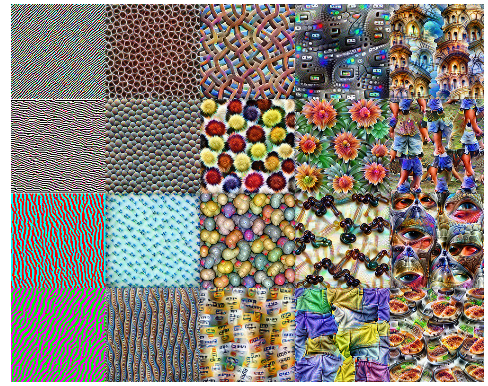
The primary way of studying language and cognition has been through laboratory experiments with humans. While we do have easy access to humans using language, the observation of the actual cognitive processes we are interested has limitations in both its direct and indirect forms. Direct observation includes using apparatus like an EEG, MEG, or fMRI; its primary disadvantages are that it requires specialized instruments, often cannot be done *in situ* and is still limited with what it can observe. Indirect observation includes methods which infer cognitive processes from external observations; for example, we might infer a limit to working memory by seeing how many digits in a long number a person can recall. The primary restrictions with indirect methods is that they, too, are very limited in what they can observe.

Some approaches to simulation for this application investigate the similarity of language models to humans in the cognitive domain (Schrimpf et al., 2020; Misra et al., 2021; Mahowald et al., 2023). These neural networks, though, are typically trained in a standard supervised or self-supervised manner (i.e., not the embodied reinforcement learning of emergent communication). Even if the model is trained with multi-modal data, the relationship between the modalities is more rigid insofar as it is restricted *a priori* by the way the model is optimized; this limits the ability to draw conclusions about human linguistic behavior where relationship between modalities is flexible and dynamic.

Applicability Observing neural networks is easier that observing the results of human-subject experiments. This is because the state and processes of artificial neural networks are completely accessible, even if they are not always easy to interpret. Furthermore, any individual aspect of an artificial neural network can be manipulated, which allows for a far higher granularity in experimentation than human subjects. Compared to using language models, emergent communication agents have a more natural integration of language capabilities with other capabilities such as perception or interpersonal communication goals. This is due to the automatically learned neural-to-neural interface between between language,



(a) Visualization of fMRI scans of the human brain in response to visual stimuli (Bracci et al., 2023).



(b) Visualization of image classification CNN layers (Olah et al., 2017).

Figure 2.6: Measurements and visualizations of artificial neural networks are easier to make and far more flexible compared to biological networks in the human brain. This makes ANNs an attractive proxy for studying the human brain.

cognition, and perception, allowing the resulting use of language to be shaped by embodiment and pressures for useful communication.

Current state The current literature in this area focuses on observing high-level principles from cognitive science and perception in the context of emergent communication systems. While these abstract facts do relate to cognitive science, they are more directly aimed at improving emergent communication techniques themselves (i.e., like an *internal* goal). Work directly applying emergent communication-trained models to particular questions within cognitive science (along the lines of Misra et al. (2021)) is largely absent.

The subtopic with the most attention in this application is the relationship between emergent communication and the agents' perception of the environment. Bouchacourt et al. (2018) establish a simple but important point regarding perception: neural-network based agents may successfully communicate with degenerate perceptual strategies. Namely, they show how agents which learn to play an image discrimination game with natural images are just as successful when playing with random noise images, demonstrating that we cannot simply assume that agents will learn intuitive or interpretable perceptual representations without further investigation.[10] Nevertheless, Dessì et al. (2021) counter this pessimism by demonstrating that it is still possible for emergent communication agents to develop interpretable visual representations on their own.

Choi et al. (2018) and Portelance et al. (2021) study how the balance of visual attributes in training data directly influences what attributes are actually perceived. Feng et al. (2023) look specifically at *relations* between visual elements in a referential game. More generally, Lazaridou et al. (2018), Ohmer et al. (2021b), and Ohmer et al. (2021a) study how the emergent communication is sensitive, in general, to the perception of the environment. While

[10]This is closely related, both technically and methodologically, to adversarial inputs in computer vision research.

most papers address *visual* perception, Khazar Khorrami (2019) looks at the emergent perception of units of sound.

Deeper than perception, some work studies the agents' internal representations themselves. Sabathiel et al. (2022) look at how agents can represent numbers to themselves by interacting with their environment (e.g., an abacus). Santamaría-Pang et al. (2019) compare representations learned with supervised methods (e.g., a convolutional neural network trained on image classification) with those learned with self-supervised learning; supervised learning yields better representations, generally, but self-supervised learning can be augmented to approach the same performance. Garcia et al. (2022) discuss how a mismatch in internal representation severely reduces the effectiveness of communication.

Finally, a handful of papers have addressed cognitive strategies themselves and specifically how human-inspired inductive biases can be beneficial both for task success and for learning intuitive representations. Todo et al. (2020) find that agents restricting their own learning process lead to languages with more interpretable structure; specifically, agents would discard training examples which diverged more than certain threshold from their own representations. Yuan et al. (2020) and Piazza et al. (2023) encourage agents to develop a theory of mind by explicitly modeling the internal states of other agents. This leads to more effective communication by introducing pragmatics into the emergent communication since agents can explicitly infer meaning from the communicative context. Masquil et al. (2022) propose adding intrinsic motivations to agents to improve communication. Finally, Cowen-Rivers et al. (2020) explore the use of world models (Ha et al., 2018) to improve agents' ability to handle environments with longer episodes.

Next steps The next steps for this area of emergent communication are to bring the research which already explores abstract principles of cognition in emergent communication closer to the more concrete questions already present in cognitive science. This would entail using emergent communication techniques in the same vein as Misra et al. (2021) and the other papers mentioned in the "Description" section. In particular, it would be especially important to identify the differences between traditional language models and emergent communication agents in terms of their cognitive realism. This would include both ways in which language models should be limited (e.g., language models having super-human recall) as well as ways in which they need to improve (e.g., discourse coherence, factuality). Incorporating cognitive science will better illuminate where emergent communication techniques diverge from human cognition and behavior and how that might influence the resulting emergent communication.

2.4.3 Origin of language

Description The origin of human language, as a task, comprises studying the environment and processes under which human language, as we recognize it today, emerged from pre-linguistic communication (e.g., methods animals use to communicate, see Figure 2.7). In particular, one of the biggest questions surrounding the origin of human language is whether it occurs gradually or through saltations (discussed in LaCroix (2019)). The gradualist position holds that there was no clear boundary or and no clear discontinuities between pre-linguistic communication and true human language while the saltationist position holds that, at some point, pre-linguistic communication underwent a sudden transition into human

Figure 2.7: Illustration of the continuum of pre-linguistic communication from simple, low-bandwidth communication (e.g., ants leaving pheromone trails) to more complex, high-bandwidth systems (e.g., the vocalizations of Rhesus monkeys) (examples and figure taken from Grupen et al. (2020)). Pre-linguistic communication systems, such as these, are an important component of studying the origin of human language.

language. Addressing this particular question is major step in determining the nature of the processes explaining the origin of human language.

Since language was originally only spoken, there are no direct data which describe what happened when it evolved. Thus, any data for research come from inferential data from animal communication, and contemporary examples of language invention (e.g., creolization, Nicaraguan Sign Language). These are relatively sparse, leaving the origin of language very difficult to study. As a result, simulation is, in a way, the closest source of data to direct observation. Yet critical factors in the origin of language include complex non-linguistic elements such as perception, internal representation, and social dynamics which traditional simulations have difficulty representing.

Applicability Simulation is a natural way to address processes, such as the origination of language, for which we have no (or limited) direct observations. Simulations permit not only observing these processing but counterfactually experimenting with them as well (e.g., answering "If I change variable X, how does Y respond?"). Such experiments are necessary for scientifically distinguishing causation from mere correlation. Yet, the dependence of the origin of language on non-linguistic factors like perception, internal representations, and social dynamics indicates a significant need for simulations which integrate learning methods with a high capacity and flexibility, that is, deep neural networks. Furthermore, learning these linguistic and non-linguistic skill jointly (as opposed to, for example, using a pre-trained vision network) is also an important point of realism which emergent communication provides as it mirrors the fact that humans learn language and other cognitive skills jointly. Additionally, using neural networks allows the simulation to reflect the evolutionary pressures in the environment instead of the stipulations of a handcrafted mathematical model.

Current state Work on language evolution and change comprises a few empirical papers which have used small-scale, simple environments to test specific hypotheses as well as a few position papers. The empirical papers typically use environments and tasks from prior work with the added element of transmission of language from generation to generation. For example, Grupen et al. (2021) look specifically at pre-linguistic communication (e.g., between animals) with emergent communication techniques as a foundation for the emergence of fully linguistic communication. F. Li et al. (2019) and Ren et al. (2020) test the effects of *iterated learning* in emergent communication environments. Iterated learning is a framework introduced by Smith et al. (2003) as a way to reason about and explain the origin of compositionality (among other things) in human language from an evolutionary perspective on language (Kirby et al., 2002; Kirby et al., 2008). The core feature of iterated learning is that when language users transmit only a subset of the language to language learners, the learners have to generalize what they have heard in order to infer the rest of the language, leading to greater systematicity and compositionality over generations.

The position papers on this topic all specifically incorporate relevant work from the linguistics side of language evolution and try to square it with the contemporary approaches of emergent communication. LaCroix (2019) compares the relative merits of gradualist and saltationist approaches to the origin of language and what bearing they have on emergent communication research, specifically arguing that the focus on compositionality might not align with gradualism. Moulin-Frier et al. (2020) highlight the opportunities and challenges of using recent advancements in multi-agent reinforcement learning for studying the origin of language. Galke et al. (2022) specifically identify the elements of current emergent communication research that must change in order to better apply to linguistically-grounded study of the origin of language.

Next steps Achieving realism in emergent communication-based simulations of the origin of language must focus on closing the gap between the two data points we do actually possess: animal communication and behavior (pre-origin of language) and contemporary human language (post-origin). Thus, the pre-origin side of this entails aligning emergent communication settings with what we can currently observe in the more sophisticated varieties of animal communication, along the lines of what Grupen et al. (2021) study. Subsequently, changes to the setting would be made to elicit more sophisticated forms of communication which would ideally result in communication bearing the traits of human language (i.e., rederivation as described in Section 2.2.1). Since the origin of language depends heavily on the aforementioned non-linguistic concepts, simulations will have to take into account the relevant literature in cognitive science and behavioral psychology.

Additionally, empirical implementations of the principled, interdisciplinary recommendations of the position papers (LaCroix, 2019; Moulin-Frier et al., 2020; Galke et al., 2022) also present concrete opportunities for quickly advancing emergent communication's relevance to studying the origin of language.

2.4.4 Language change

Description Languages are perpetually changing, sometimes above and sometimes below the level of conscious awareness. Language change refers to the processes which govern how language changes and develops over time in human populations. In a groundbreaking paper in language change, Weinreich, Labov, and Herzog identified five problems regarding how languages change over time Weinreich et al., 1968:

constraints What constrains the transition of a language from a state s_{t-1} to a successor state s_t? In particular, are there impossible languages that no change could produce?

transition What intervening stages must exist between states s_{t-1} and s_t? For example, do the two language varieties coexist for a time?

embedding How are the observed changes embedded in the matrix of linguistic and extralinguistic concomitants of the forms in question? What other changes co-occur with the change non-accidentally?

evaluation How do members of the language community subjectively evaluate the change that is underway or has occurred?

actuation Why does a particular change occur at a particular point in time and space?

While human laboratory experiments have been useful in addressing some of these problems (Roberts, 2017), as have field studies and other social-scientific methodologies, emergent communication simulations provide an unprecedented means of addressing all of these problems except *evaluation*.

Applicability Emergent languages in multi-agent simulations change over time. If they did not—in some respect—change, they would never develop language-like properties in the first place. Thus we can ask if they reach stable equilibria and, if so, where and why do changes occur, if at all. In answering this question, emergent communication simulations can address the *actuation problem* (one of the most difficult problems in language change). These simulations allow us to dissect the relationships between language changes and changes in the "social" and "physical" environment as well, addressing the *embedding problem*. But because emergent communication simulations give us a kind of omniscience, they also allow us to characterize the stages between stable equilibria, providing a window onto the *transition problem*. Finally, because emergent communication researchers are free to add and remove constraints on possible languages at will, such simulations allow us to address questions about whether human-like language change requires constraints on what languages are "legal" (addressing the *constraints* problem in a way that bears upon the behaviorism-nativism debate).

Current state Language change has not received much attention in the literature; only two papers were found in the survey which approached the topic specifically. First, Graesser et al. (2019) study language contact, where two or more populations of agents who have developed their own language in relative isolation subsequently start communicating with each other. In particular, the experiments replicated a handful of general language contact phenomena that are known to occur with human language. First, while dialects start out as mutually unintelligible, interaction between subsets of to populations can cause convergence of all agents to a mutually intelligible language. Second, when this contact occurs, either the larger population's language will dominate and take over the smaller population's or a type of creole will form with a lower overall complexity. Finally, when there is a linear chain of populations, a continuum of mutual intelligibility emerges where populations with fewer degrees of separation develop more similar dialects. These findings primarily address the *embedding problem* mentioned above.

Dekker et al. (2020) propose a set of of emergent communication experiments studying a historical instance of language change, namely morphological simplification in Alorese, a language of Eastern Indonesia. Specifically, the experiments look to determine if adult language contact can explain the loss of verb inflection in the whole language over time. The proposed approach is based on deep neural networks and proposes leveraging cognitive two cognitive mechanisms: Ullman's declarative/procedural model of language learning (Ullman, 2001b; Ullman, 2001a) and Lindblom's H&H model (Lindblom, 1990).

Next steps Emergent communication studies of the transition problem have the most potential near-term progress. In particular, studies could investigate quantitatively and at scale how transition between two stable states s_{t-1} and s_t takes place. Specifically, one could investigate whether two languages coexist within a community of agents, with one gradually gaining currency or first dominating a subgraph of the social network, or whether changes happen abruptly across the whole population. Such studies with emergent communication could then be compared to historical examples of the transition program to verify and improve the effectiveness of emergent communication approaches.

2.4.5 Language acquisition

Description Language acquisition is the process by which a human acquires the ability to use a new language. For this application, we will focus on first language acquisition because it has weightier scientific implications than second language acquisition and stands to gain more from emergent communication techniques due to how it co-occurs with the acquisition of important non-linguistic behaviors like reasoning and memory. Compared to the origin of language (Section 2.4.3), observational data of first language acquisition data is readily available as it always occurring in a population of humans. Compared to the cognitive and perceptual aspects of language (Section 2.4.2), there is more to be learned from direct observation of external behavior, making the data easier to collect. Nevertheless, data on first language acquisition is predominantly *observational*, that is, not derived from controlled, randomized experiments. Experiments which test anything more than superficial aspects of language acquisition could have drastic negative effects on human subjects and would be wholly unethical. Thus, data from more involved experimental methods on first language acquisition has to come from other sources such as neural networks trained on language data. Neural networks trained purely on *text* language data, though, fall far short of human performance given a similar amount of language data, suggesting the non-linguistic inputs might be key to replicating human language acquisition (Warstadt et al., 2022).

Applicability Emergent communication naturally integrates non-linguistic inputs into language (e.g., embodiment, interaction) into the acquisition of language by the neural network agents instead of stipulating ahead of time how such inputs will impact the emergent communication (Warstadt et al., 2022; Bisk et al., 2020). Furthermore, the ease of observing and experimenting with neural networks vastly surpasses doing so with human subjects. These advantages of using emergent communication as a simulation technique for studying language acquisition are generally similar to those discussed in *Origin of language* (Section 2.4.3) and *Language, cognition, and perception* (Section 2.4.2) (see those sections for further details). While many of the advantages of emergent communication techniques in studying language acquisition could be derived from more traditional machine learning methods used on multi-modal data, these traditional methods are only ever *mimicking* the acquisition process that a human goes through to acquire that human language (as a first language). On the other hand, with emergent language acquisition, we can observe the selfsame acquisition process that has formed the language in the first place and not just an approximation thereof. This direct connection is important since every step in empirical reasoning which involves approximations brings with it more uncertainty in the conclusions.

Current state Current literature has not often investigated language acquisition, so we will address the collected work exhaustively. At the level of individuals, current work has mainly looked at how the process of language acquisition interacts with the emergence of compositionality and other properties of language. Korbak et al. (2019) and Korbak et al. (2021) propose a developmentally-inspired curriculum which breaks down language learning into multiple phases; they then show that this method results in more compositional emergent communication. Cope et al. (2022) present a method by which a new agent could acquire a pre-existing emergent language purely through observation by inferring the intentions of the observed agents. Kharitonov et al. (2020a) investigate a relationship in the opposite direction, looking at how the degree of compositionality of a language factors into the ease and speed of language acquisition. At a population level, F. Li et al. (2019) investigate the same relationship between ease of acquisition and compositionality in a generationally transmitted setting, arguing (in line with Smith et al. (2003)) that the pressure to acquire language from incomplete data can translate to a pressure towards compositional language. Leaving aside compositionality, Portelance et al. (2021) study the origin of shape bias, arguing that it can be explained with communicative efficiency pressures rather than inductive biases in the human or machine agents.

Next steps The next steps for studying language acquisition are to demonstrate how emergent communication techniques build directly on prior work studying deep neural network-based models of language acquisition. Warstadt et al. (2022) mention that neural networks hold potential for studying language learning but also present a number of difficulties; thus future work in emergent communication would do well to follow existing work on the topic closely (at least for the near term). For example, Warstadt et al. (2020) determine that a neural network (namely BERT) is able to make structural generalizations in natural language but only after a observing more data than is developmentally realistic. Similarly, Chang et al. (2022) compare word acquisition in children and language models. In both cases, emergent communication could help determine if the lack of embodiment and interactivity in standard language model training explains part of why language models require significantly more data than humans to acquire the same proficiency with language.

2.4.6 Linguistic variables

Description Linguistic variables are the particular phenomena in language and its use which are the subject of scientific study in linguistics. This is a catch-all application which includes all studies seeking to determine the relationships between linguistic and other linguistic/non-linguistic variables. These variables span all of the various subfields of linguistics, forming a rough low- to high-level hierarchy:

phonology patterns of individual units of sound

morphology patterns of individual units of meaning at the word and sub-word level

syntax organization of words into meaningful structures (e.g., phrases, clauses, sentences)

semantics the inherent meaning of utterances in a language

pragmatics meaning derived from context cues in conjunction with semantics

sociolinguistics properties of language in the context of group and social dynamics

Beyond identifying individual relationships, broader questions within linguistics concern patterns across relationships. In particular, a central question across all of the above fields, has been the degree to which linguistic variables are the product of formal properties of cognition (formalism) and to what extent they are the emergent result of language use in a communicative context (functionalism). For example, is the tendency of vowel systems to be more-or-less maximally dispersed with the formant space a result of formal universals such as a categorical phonological features that impose a straitjacket on the realization of the vowels or a result—in language evolution—of vowel distinctions that are not well-dispersed collapsing (leaving only the well-dispersed vowels behind) (Blevins, 2004).[11]

Likewise, it has been observed that prefixes and suffixes (in words that have more than one) are ordered so that those with the greatest relevance to the meaning of the root are closest to the root. This has been attributed to a formal constraint in which morphological scope mirrors syntactic scope (the Mirror Principle) M. Baker, 1985 or as a functional tendency based on a motivation, on the part of speakers, to distribute information predictably so that units of language are closest to the other units to which they are most relevant (the Relevance Principle) (Bybee, 1985). This distribution is argued to be the result of evolutionary processes emerging from attempts of language users to communicate with one another (Bybee, 1985).

The evolution of pragmatics is even less-well understood. Is contextual meaning a result of inherent principles of inference or is it an emergent property of communicative interaction? Linguists have not been able to resolve these issues experimentally because they involve simulating conversations between speakers over decades and centuries—not interactions that can be observed during an afternoon in the lab.

[11] Or, perhaps, due to a human drive to communicate as clearly as possible, given the same investment of effort (Flemming, 2013).

Applicability In addition to the aforementioned applicable traits of emergent communication, there are two ways in which emergent communication is particularly applicable to studying linguistic variables. First, studying variables in any scientific discipline requires isolating these variables from confounding factors. Within emergent communication, it is possible to strip away confounding factors in ways that are often not possible when studying humans directly.

Secondly, the holistic way in which emergent communication simulates linguistic processes makes it particularly suitable to studying phenomena that span multiple levels of the linguistic hierarchy. For example, the variables relevant to the distinction between "who" and "whom" in modern English span morphology ("-m" as an affix), syntax ("who" functioning as a subject or object and "whom" as solely an object), and sociolinguistic ("whom" being perceived as formal, dated, etc.). Emergent communication, by design, allows for the interaction between many of the levels in the hierarchy without stipulating a particular way in which they interact. On the other hand, more traditional methods of modeling linguistic variables tend to be limited to just the micro or macro scale, and any interaction between these has to be determined ahead of time through handcrafted schemata, limiting the range of potential outcomes.

Current state Linguistic variables, broadly construed, show up frequently in the literature as almost any property of emergent communication can be considered a "linguistic variable". For example, papers studying compositionality or grounding are addressing a relationship between *syntax* and *semantics* while papers looking at how to leverage extra-linguistic context for better communication are addressing *pragmatics*. Nevertheless, we mention papers here which directly tie into the study of human language and "linguistics" in the narrower sense. Given that emergent communication is in the stage of trying to look more like human language (cf. Section 2.2.1), the current literature in this application primarily focuses on recreating established linguistic phenomena in emergent communication settings. The following is list of summarizing the existing literature:

phonology In contrast to most emergent communication environments which have discrete communication channels, Lan et al. (2020) and Eloff et al. (2021) look at continuous channels and discretization pressures analogous to the relationship between phones and phonemes.

syntax Chaabouni et al. (2019b) study whether or not emergent communication displays word-order biases akin to many human languages. Wal et al. (2020) analyze the output of unsupervised grammar induction applied to emergent communication.

semantics Chaabouni et al. (2021), Rita et al. (2020), and Rodríguez Luna et al. (2020) study the conditions under which Zipf's Law of Abbreviation (Zipf, 1950) is present in emergent communication. Kågebäck et al. (2018) and Chaabouni et al. (2021) study the way emergent communication divides up color spaces as compared to human languages. Finally, Steinert-Threlkeld (2019) looks at the emergence of function words in emergent communication as opposed to the exclusively content-based words in most other settings.

sociolinguistics Graesser et al. (2019), Kim et al. (2021), and Fulker et al. (2022) look at the formation of dialects under different conditions in networks of interacting agents. See "Current State" of Section 2.4.4 for Dekker et al. (2020).

Next steps Phonology and morphology are relatively understudied in this area since most emergent communication environments assume a one-to-one correspondence between discrete symbols and "words". The paradigm of discrete symbols-as-words precludes analyzing sub-word components since a discrete symbol has no structure. Thus, breaking away from this paradigm would open new avenues for research into the phonological and morphological aspects of emergent communication. This could be done either be simply analyzing discrete symbols as sub-word units (requiring some other definition for what constitutes a word in an emergent language), or by using a continuous communication channel with some sort of discretization pressure (such that clusters of continuous signals can be analyzed as discrete units).

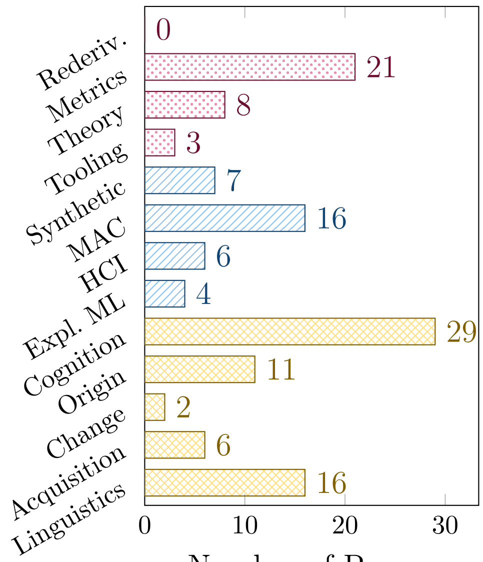
Syntax and semantics are already studied in emergent communication, although this research needs to be more tightly coupled with thoroughly linguistic accounts of these phenomena instead of relying on looser, higher-level analogies with linguistics. This is a non-trivial task insofar as the definitions and models from linguistics will need to be adapted to the unique difficulties of emergent communication. For example, emergent communication can have radically different forms compared human language (or no organization at all); this means that linguistic accounts may make assumptions about the language being studied that do not necessarily hold for emergent communication (e.g., languages are, at most, mildly context sensitive). Thus, operationalizing linguistic definitions for emergent communication will require expanding their scope to account for the numerous edge cases that emergent communication presents.

For pragmatics and sociolinguistics, emergent communication environments will generally have to incorporate more agents, temporality, and embodiment. This is because these linguistic phenomena operate across many instances of language use with a common context across time and among speakers (e.g., conversational, spatial, and cultural context). In contrast, many emergent communication environments currently use single-step, simple observation, two-agent environments which preclude observing almost all pragmatic and sociolinguistic phenomena. The above point about linguistic definitions syntax and semantics requiring adaptation to emergent communication holds true for pragmatics and sociolinguistics as well since many behavior biases and heuristics we observe in humans emergent communication agents may not possess at all.

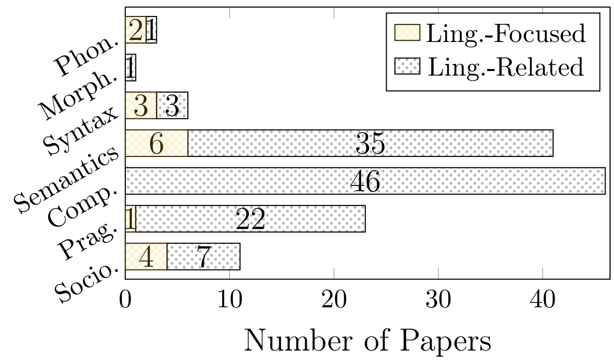
2.5 Discussion

2.5.1 Quantitative summary of results

In Figure 2.8, we present a quantitative summary of the categorization of papers covered in our survey. Figure 2.8a shows at the number of paper falling within the scope of each application, and Figure 2.8b further breaks the down *Linguistic variables* (Section 2.4.6) into the different fields of linguistics. Note that there is not a one-to-one correspondence



(a) Number of papers corresponding to each section of this review.



(b) Breakdown of linguistic variables papers; "Linguistic-Focused" refers to papers discussed in Section 2.4.6, while "Linguistic-Related" includes anything more loosely related to that area of linguistics.

Figure 2.8: Quantitative summary of paper topics in this survey. Abbreviations: *MAC*: multi-agent communication, *HCI*: human–computer interaction, *Expl. ML*: explainable machine learning, and *Comp.*: compositionality.

between papers and applications, a paper may have no applications if its contributions are not properly applications or more than one application if its contribution touches on multiple areas.

2.5.2 Internal goals

The internal goals of emergent communication prove tricky for this survey since they both make up the majority of contributions in emergent communication papers but only loosely qualify as applications. Many of the papers we surveyed listed contributions along the lines of "introducing an environment where we can observe X phenomenon" or "demonstrating a relationship between variable X in the environment and variable Y in the emergent language". The second of these was by the most common in the 245 emergent communication papers surveyed: it appeared 116 times whereas the next highest category was "related to compositionality" with only 46 papers. This is not to say that these contributions are unimportant or unnecessary, but they fail to be true applications in the sense of being a focused "goal" which a line of research can pursue. Thus, such contributions were omitted from this survey.

Aside from these non-application contributions, the topic of *metrics* was the most common. Many of these metrics, though, are not treated as applications or goals in themselves as they are introduced for the needs of the paper and do not see reuse in subsequent papers. Nevertheless, some papers do explicitly aim towards better metrics, comparing the quality of metrics in an effort to refine the tools researchers have for analyzing emergent communication

(Lowe et al., 2019; Korbak et al., 2020).

Finally, *Rederiving human language* (Section 2.2.1) was one goal which we included in this paper, functioning more like a position paper than a survey paper. This goal did was not explicitly pursued by any of the papers we reviewed, although it is implicit in a large number of papers, namely those which seek to align emergent communication with some human language-like quality (e.g., compositionality, Zipf's Law of Abbreviation). Nevertheless, we argue that rederiving human language should receive more attention which addresses it holistically. This is because (1) it is critical to making possible the downstream task- and knowledge-driven applications and (2) it is an effective way to interpret the other contributions falling under the "internal" umbrella.

2.5.3 Task-driven applications

Within the task-driven applications we find that multi-agent communication has received the most attention. This is generally expected as it is one the most natural applications of emergent communication, given its foundation in deep multi-agent reinforcement learning. Although some papers have addressed using emergent communication for synthetic data, it is somewhat surprising that the number is not higher since it is probably the application with the most potential for near-term success, especially in low-resource domains as a replacement for traditional synthetic data. Interacting with humans, while an important long-term goal, does not hold as much short-term promise because it is more difficult to conduct scientific studies with humans and anything short of near-human language-like emergent communication is not going to surpass other methods for interfacing with humans through natural language.

2.5.4 Knowledge-driven applications

Within the knowledge-driven the applications, we find the cognitive and core linguistic aspects of emergent communication to the most addressed. The number shown for "Cognition" in Figure 2.8a includes paper using a broader sense of "cognition" and "cognitive science" including topics like perception, internal representations, and neural architectures.[12] As shown in Figure 2.8b, core linguistics, when given this broader interpretation is far more prevalent with over 100 in total. By comparison, language change and language acquisition of language are more niche and have fewer papers associated with them.

Within the umbrella of linguistic variables, we can see a handful of trends. First, we see generally in Figure 2.8b that the number of papers which address variables directly relevant to linguistics is dwarfed by the number of papers which take only a loose inspiration from linguistics. Compositionality is especially interesting in this regard as it is, by far, the most written-about topic under the broad umbrella of linguistics, yet we did not find any papers addressing it from a strongly linguistic and human language-oriented perspective.

This may be due, in part, to the fact that human languages are universally compositional and generally have similar methods of composing meaning at a broad level in comparison

[12] Although we did not perform the same focused-versus-related breakdown with cognition as was done with linguistic variables, we expect we would have found a similar divide with many cognition-related papers and only a handful of papers which focus on issues directly relevant to cognitive science (as illustrated in Figure 2.8b).

to many ways in which emergent communication may or may not be compositional. Aside from compositionality, semantics and pragmatics are the most studied topics. These areas of linguistics naturally line up with the most foundational aspects of emergent communication, namely figuring out what emergent languages are actually communicating (semantics) and how this meaning derives from communication strategies and environmental pressures (pragmatics). Finally, phonology and morphology have the least amount of work focused on them. One potential reason for this is that emergent communication systems are typically structured in a way to preclude phonology by using discrete communication channels and morphology by assuming discrete symbols to already be individual units of meaning (i.e., morphemes) without investigating potential subword structure further.

2.6 Conclusion

In this paper, we have given a comprehensive summary of the goals and applications of deep learning-based emergent communication research. The applications of emergent communication can roughly be categorized into those which aim at: improving emergent communication techniques themselves (internal); solving well-defined, practical problems (task-driven); and expanding human knowledge of the natural world (knowledge-driven). Each of these applications has been accompanied by a description of its scope, an explication of emergent communication's unique role in addressing it, a summary of the extant literature working towards the application, and brief recommendations for near-term research directions. Finally, we identify general trends observed in the course of surveying the applications of emergent communication.

This work has three primary goals. First, it is meant to inspire future emergent communication research by compiling the most salient areas of research into a single document with relevant work cited. Second, this work is meant to accessibly illustrate the potential applications of emergent communication to practitioners who are not as familiar with the multi-agent reinforcement learning or deep learning in general. Finally, defining the ultimate aims of emergent communication is critical to guiding the field of research itself through practices like evaluation metrics and benchmarks. Evaluation metrics require explicitly defining what a *good* or *desirable* emergent language is, and understanding what emergent communication can be used for is a foundational step. While this paper does not come close to exhausting the nuances of each of these applications, it highlights the nature and importance of applications as a whole in order to serve the future of emergent communication research.

Chapter 3

Building a Library of Emergent Languages

Abstract

This chapter introduces the Emergent Language Corpus Collection (ELCC): a collection of corpora generated from open source implementations of emergent communication systems across the literature. These systems include a variety of signaling game environments as well as more complex environments like a social deduction game and embodied navigation. Each corpus is annotated with metadata describing the characteristics of the source system as well as a suite of analyses of the corpus (e.g., size, entropy, average message length, performance as transfer learning data). Currently, research studying emergent languages requires directly running different systems which takes time away from actual analyses of such languages, makes studies which compare diverse emergent languages rare, and presents a barrier to entry for researchers without a background in deep learning. The availability of a substantial collection of well-documented emergent language corpora, then, will enable research which can analyze a wider variety of emergent languages, which more effectively uncovers general principles in emergent communication rather than artifacts of particular environments. We provide some quantitative and qualitative analyses with ELCC to demonstrate potential use cases of the resource in this vein.[1]

3.1 Introduction

In a typical paper studying emergent language, one of the major components involves implementing a particular emergent language environment which will be used for empirically testing the properties of emergent communication. This requires either creating a codebase from scratch or adapting a previously existing code, and in both cases, this requires a significant amount of time or may be out of reach for those without a background in computing. One result of the substantial effort required to even generate emergent languages is that many studies only look at a handful of emergent languages with little diversity in environment design. This, in turn, obscures whether the observed phenomena are general

[1]Based on "ELCC: the Emergent Language Corpora Collection" available on arXiv (Boldt et al., 2024b).

features of emergent communication or simply artifacts of the particular environment, and any sort of comparative analysis between varieties of emergent languages is unavailable as well.

We present an initial solution to this problem, namely the Emergent Language Corpus Collection (ELCC): a collection of 73 corpora generated from 7 representative emergent communication systems (ECSs).[2] Each corpus is accompanied by metadata describing the environment, summary statistics for the generated corpus, and a ready-to-run codebase for reproducing the corpus. This collection allows investigators, even those with very limited software engineering knowledge, to analyze a wide range of emergent languages straightforwardly, removing a barrier that has held back comparative emergent language research from its inception. ELCC is published at `https://huggingface.co/datasets/bboldt/elcc` with data and code licensed under the CC BY 4.0 and MIT licenses, respectively.

We discuss related work in Section 3.2. Section 3.3 lays out the design of ELCC while Section 3.4 describes the content of the collection. Section 3.5 demonstrates some of the types of analyses enabled by ELCC. Section 3.6 presents some brief analyses, discussion, and future work related to ELCC. Finally, we conclude in Section 3.7.

Contributions The primary contribution of this paper is as a first-of-its kind data resource which will enable broader engagement and new research directions within the field of emergent communication. Additionally, code published for reproducing the data resource also improve the reproducibility of existing ECS implementations in the literature, supporting further research beyond just the data resource itself. Finally, the paper demonstrates some of the analyses uniquely made possible by a resource such as ELCC.

3.2 Related Work

Emergent communication There is no direct precedent for this work in the emergent communication literature that we are aware of. Perkins (2021b) introduces the TexRel dataset, but this is a dataset of observations for training ECSs, not data generated by them. Some papers do provide the emergent language corpora generated from their experiments (e.g., Yao et al. (2022)), although these papers are few in number and only include the particular ECS used in the paper. At a high level, the EGG framework (Kharitonov et al., 2019) strives to make emergent languages easily accessible, though instead of providing corpora directly, it provides a framework for implementing ECSs. Thus, while EGG is useful for someone building new systems entirely, it is not geared towards research projects aiming directly at analyzing emergent languages themselves.

Data resources At a high level, ELCC is a collection of datasets, each of which represent a particular instance of a phenomenon (emergent communication, in this case). On a structural level, ELCC is analogous to a collection of different human languages in a multi-lingual

[2]Emergent communications systems are more commonly referred to as simply "environments"; we choose to use the term "system" in order to emphasize that what goes into producing an emergent language is more than just an environment including also the architecture of the agents, optimization procedure, datasets, and more.

```
systems/ ........................ top-level directory
  ecs-1/ ........................ directory for a particular ECS
    metadta.yml ................. metadata about the ECS
    code/ ....................... directory containing files to produce the data
    data/ ....................... directory containing corpus and metadata files
      hparams-1/ ................ directory for run with specific hyperparameters
        corpus.jsonl ............ corpus data
        metadata.json ........... metadata specific for corpus (e.g., metrics)
      hparams-2/ ................ as above
      hparams-n/ ................ as above
  ecs-2/ ........................ as above
  ecs-n/ ........................ as above
```

Figure 3.1: The file structure of ELCC.

dataset. ELCC, though, focuses more on a particular phenomenon of scientific interest, and, in this way, would be more analogous to work such as Blum et al. (2023), which presents a collection of grammar snapshot pairs for 52 different languages as instances of diachronic language change. Similarly, Zheng et al. (2024) present a dataset of conversations from Chatbot Arena, where "text generated by different LLMs" is the phenomenon of interest. Furthermore, insofar as ELCC documents the basic typology of different ECSs, it is similar to the World Atlas of Language Structures (WALS) (Dryer et al., 2013).

3.3 Design

3.3.1 Format

ELCC is a collection of ECSs, each of which has one or more associated *variants* which correspond to runs of the system with different hyperparameter settings (e.g., different random seed, message length, dataset). Each variant has metadata along with the corpus generated from its settings. Each ECS has its own metadata as well and code to generate the corpus and metadata of each variant. The file structure of ELCC is illustrated in Figure 3.1.

ECS metadata Environment metadata provides a basic snapshot of a given system and where it falls in the taxonomy of ECSs. As the collection grows, this structure makes it easier to ascertain the contents of the collection and easily find the most relevant corpora for a given purpose. This metadata will also serve as the foundation for future analyses of the corpora by looking at how the characteristics of an ECS relate to the properties of its output. These metadata include:

- Source information including the original repository and paper of the ECS.
- High-level taxonomic information like game type and subtype.
- Characteristics of observation; including natural versus synthetic data, continuous versus discrete observations.

- Characteristics of the agents; including population size, presence of multiple utterances per episode, presence of agents that send *and* receive messages.
- Free-form information specifying the particular variants of the ECS and general notes about the ELCC entry.

A complete description is given in Section B.1. These metadata are stored as YAML files in each ECS directory. A Python script is provided to validate these entries against a schema. See Section B.2 for an example of such a metadata file.

Corpus Each *corpus* comprises a list of *lines* each of which is, itself, an array of *tokens* represented as integers. Each line corresponds to a single episode or round in the particular ECS. In the case of multi-step or multi-agent systems, this might comprise multiple individual utterances which are then concatenated together to form the line (no separation tokens are added). Each corpus is generated from a single run of the ECS; that is, they are never aggregated from distinct runs of the ECS.

Concretely, a *corpus* is formatted as a JSON lines (JSONL) file where each *line* is a JSON array of integer *tokens* (see Figure 3.3 for an example of the format). There are a few advantages of JSONL: (1) it is a human-readable format, (2) it is JSON-based, meaning it is standardized and has wide support across programming languages, and (3) it is line-based, meaning it is easy to process with command line tools.[3] Corpora are also available as single JSON objects (i.e., and array of arrays), accessible via the Croissant ecosystem (Akhtar et al., 2024).

Corpus analysis For each corpus in ELCC we run a suite of analyses to produce a quantitative snapshot. This suite metrics is intended not only to paint a robust a picture of the corpus but also to serve as jumping-off point for future analyses on the corpora. Specifically, we apply the following to each corpus: token count, unique tokens, line count, unique lines, tokens per line, tokens per line stand deviation, 1-gram entropy, normalized 1-gram entropy, entropy per line, 2-gram entropy, 2-gram conditional entropy, EoS token present, and EoS padding. *Normalized* 1-*gram entropy* is computed as 1-*gram entropy* divided by the maximum entropy given the number of unique tokens in that corpus.

We consider an EoS (end-of-sentence) token to be present when: (1) every line ends with token consistent across the entire corpora, and (2) the first occurrence of this token in a line is only ever followed by more of the same token. For example, `0` could be an EoS token in the corpus `[[1,2,0],[1,0,0]]` but not `[[1,2,0],[0,1,0]]`. EoS padding is defined as a corpus having an EoS token, all lines being the same length, and the EoS token occurs more than once in a line at least once in the corpus.

Additionally, each corpus also has a small amount of metadata copied directly from the output of the ECS; for example, this might include the success rate in a signaling game environment. We do not standardize this because it can vary widely from ECS to ECS, though it can still be useful for comparison to other results among variants within an ECS.

[3] E.g., Creating a 100-line random sample of a dataset could be done with `shuf dataset.jsonl | head -n 100 > sample.jsonl`

Source	Type	Data source	Multi-agent	Multi-step	n corp.
Kharitonov et al. (2019)	signaling	synthetic	No	No	15
Yao et al. (2022)	signaling	natural	No	No	2
J. Mu et al. (2021)	signaling	both	No	No	6
Chaabouni et al. (2022)	signaling	natural	Yes	No	5
Unger et al. (2020)	navigation	synthetic	No	Yes	18
Boldt et al. (2022a)	navigation	synthetic	No	Yes	20
Brandizzi et al. (2022a)	conversation	—	Yes	Yes	7

Table 3.1: Taxonomic summary the contents of ELCC.

Reproducibility ELCC is designed with reproducibility in mind. With each ECS, code is included to reproduce the corpora and analysis metadata. Not only does this make ELCC reproducible, but it sometimes helps the reproducibility of the underlying implementation insofar as it fixes bugs, specifies Python environments, and provides examples of how to run an experiment with a certain set of hyperparameters. Nevertheless, in this code, we have tried to keep as close to the original implementations as possible. When the underlying implementation supports it, we set the random seed (or keep the default) for the sake of consistency, although many systems do not provide a way to easily set this.

3.4 Content

ELCC contains 73 corpora across 8 ECSs taken from the literature for which free and open source implementations were available. With our selection we sought to capture variation across a three distinct dimensions:

1. Variation across ECSs generally, including elements like game types, message structure, data sources, and implementation details.
2. Variation among different hyperparameter settings within an ECS, including message length, vocabulary size, dataset, and game difficulty.
3. Variation within a particular hyperparameter setting that comes from inherent stochasticity in the system; this is useful for gauging the stability or convergence of an ECS.

Table 3.1 shows an overview of the taxonomy of ELCC based on the ECS-level metadata. In addition to this, Table 3.2 provides a quantitative summary of the corpus-level metrics described in Section 3.3.1. We separate the discussion of particular systems into two subsections: signaling games (Section 3.4.2) and its variations which represent a large proportion of system discussed in the literature and other games (Section 3.4.3) which go beyond the standard signaling framework.

3.4.1 Scope

The scope of the contents of ELCC is largely the same as discussed in reviews such as Lazaridou et al. (2020) and Section 2.1.1. This comprises agent-based models for simulating the formation of "natural" language from scratch using deep neural networks. Importantly, *from scratch* means that the models are not pretrained or tuned on human language. Typically, such simulations make use of reinforcement learning to train the neural networks, though this is not a requirement in principle.

One criterion that we do use to filter ECSs for inclusion is its suitability for generating corpora as described above. This requires that the communication channel is discrete, analogous to the distinct words/morphemes which for the units of human language. This excludes a small number of emergent communication papers have approached emergent communication through constrained continuous channels like sketching (Mihai et al., 2021a) or acoustic-like signals (Eloff et al., 2021). Other systems use discrete communication but have episodes with only a single, one-token message (e.g., Tucker et al. (2021)), which would have limited applicability to many research questions in emergent communication.

3.4.2 Signaling games

The *signaling game* (or *reference game*) (Lewis, 1969) represents a plurality, if not majority, of the systems present in the literature. A brief, non-exhaustive review of the literature yielded 43 papers which use minor variations of the signaling game, a large number considering the modest body of emergent communication literature (see Section B.3). The basic format of the signaling game is a single round of the *sender* agent making an observation, passing a message to the *receiver* agent, and the receiver performing an action based on the information from the message. The popularity of this game is, in large part, because of its simplicity in both concept and implementation. Experimental variables can be manipulated easily while introducing minimal confounding factors. Furthermore, the implementations can entirely avoid the difficulties of reinforcement learning by treating the sender and receiver agents as a single neural network, resulting in autoencoder with a discrete bottleneck which can be trained with backpropagation and supervised learning.

The two major subtypes of the signaling game are the *discrimination game* and the *reconstruction game*. In the discrimination game, the receiver must answer a multiple-choice question, that is, select the correct observation from among incorrect "distractors". In the reconstruction game, the receiver must recreate the input directly, similar to the decoder of an autoencoder.

Vanilla For the most basic form of the signaling game, which we term "vanilla", we use the implementation provided in the Emergence of lanGuage in Games (EGG) framework (Kharitonov et al., 2019, MIT license). It is vanilla insofar as it comprises the signaling game with the simplest possible observations (synthetic, concatenated one-hot vectors), a standard agent architecture (i.e., RNNs), and no additional dynamics or variations on the game. Both the discrimination game and the reconstruction game are included. This system provides a good point of comparison for other ECSs which introduce variations on the signaling game. The simplicity of the system additionally makes it easier to vary hyperparameters:

for example, the size of the dataset can be scaled arbitrarily and there is no reliance on pretrained embedding models.

Natural images "Linking emergent and natural languages via corpus transfer" (Yao et al., 2022, MIT license) presents a variant of the signaling game which uses embeddings of natural images as the observations. In particular, the system uses embedded images from the MS-COCO and Conceptual Captions datasets consisting of pictures of everyday scenes. Compared to the uniformly sampled one-hot vectors in the vanilla setting, natural image embeddings are real-valued with a generally smooth probability distribution rather than being binary or categorical. Furthermore, natural data distributions are not uniform and instead have concentrations of probability mass on particular elements; this non-uniform distribution is associated with various features of human language (e.g., human languages' bias towards describing warm colors (Gibson et al., 2017; Zaslavsky et al., 2019)).

Concept-based observations "Emergent communication of generalizations" (J. Mu et al., 2021, MIT license) presents a variant of the discrimination signaling game which they term the *concept game*. The concept game changes the way that the sender's observation corresponds with the receiver's observations. In the vanilla discrimination game, the observation the sender sees is exactly the same as the correct observation that the receiver sees. In the concept game, the sender instead observes a set of inputs which share a particular concept (e.g., red triangle and red circle are both red), and the correct observation (among distractors) shown to the receiver contains the same concept (i.e., red) while not being identical to those observed by the sender. The rationale for this system is that the differing observations will encourage the sender to communicate about abstract concepts rather than low-level details about the observation. This ECS also presents the vanilla discrimination game as well as the *set reference game*, which is similar to the reference game except that the whole object is consistent (e.g., different sizes and locations of a red triangle).

Multi-agent population "Emergent communication at scale" (Chaabouni et al., 2022, Apache 2.0-license) presents a signaling game system with populations of agents instead of the standard fixed pair of sender and receiver. For each round of the game, then, a random sender is paired with a random receiver. This adds a degree of realism to the system, as natural human languages are developed within a population and not just between two speakers (cf. idioglossia). More specifically, language developing among a population of agents prevents some degree "overfitting" between sender and receiver; in this context, having a population of agents functions as an ensembling approach to regularization.

3.4.3 Other games

Considering that the signaling game is close to the simplest possible game for an ECS, moving beyond the signaling game generally entails an increase in complexity. There is no limit to the theoretical diversity of games, although some of the most common games that we see in the literature are conversation-based games (e.g., negotiation, social deduction) and navigation games. These games often introduce new aspects to agent interactions like: multi-step episodes, multi-agent interactions, non-linguistic actions, and embodiment.

These kinds of systems, as a whole, are somewhat less popular in the literature. On a practical level, more complex systems are more difficult to implement and even harder to get to converge reliably—many higher-level behaviors, such as planning or inferring other agent's knowledge, are difficult problems for reinforcement learning in general, let alone with discrete multi-agent emergent communication. On a methodological level, more complexity in the ECS makes it harder to formally analyze the system as well as eliminate confounding factors in empirical investigation. With so many moving parts, it can be difficult to prove that some observed effect is not just a result of some seemingly innocent hyperparameter choice (e.g., learning rate, samples in the rollout buffer) (Boldt et al., 2022a). Nevertheless, we have reason to believe that these complexities are critical to understanding and learning human language as a whole (Bisk et al., 2020), meaning that the difficulties of more complex systems are worth overcoming as they are part of the process of creating more human-like emergent languages, which are more informative for learning about human language and more suitable for applications in NLP.

Grid-world navigation "Generalizing Emergent Communication" (Unger et al., 2020, BSD-3-clause license) introduces an ECS which takes some of the basic structure of the signaling game and applies it to a navigation-based system derived from the synthetic Minigrid/BabyAI environment (Chevalier-Boisvert et al., 2018; Chevalier-Boisvert et al., 2023). A sender with a bird's-eye view of the environment sends messages to a receiver with a limited view who has to navigate to a goal location. Beyond navigation, some environments present a locked door for which the receiver must first pick up a key in order to open. What distinguishes this system most from the signaling game is that it is multi-step and embodied such that the utterances within an episodes are dependent on each other. Among other things, this changes the distribution properties of the utterances. For example, if the receiver is in Room A at timestep T, it is more likely to be in Room A at timestep $T + 1$; thus if utterances are describing what room the receiver is in, this means that an utterance at $T + 1$ has less uncertainty given the content of an utterance at T. Practically speaking, the multiple utterances in a given episode are concatenated together to form a single line in the corpus in order to maintain the dependence of later utterances on previous ones.

Continuous navigation "Mathematically Modeling the Lexicon Entropy of Emergent Language" (Boldt et al., 2022a, GPL-3.0 license) introduces a simple navigation-based ECS which is situated in a continuous environment. A "blind" receiver is randomly initialized in an obstacle-free environment and must navigate toward a goal zone guided by messages from the sender which observes the position of the receiver relative to the goal. The sender sends a single discrete token at each timestep, and a line in the dataset consists of the utterances from each timestep concatenated together. This system shares the time-dependence between utterances of the grid-world navigation system although with no additional complexity of navigating around obstacle, opening doors, etc. On the other hand, the continuous nature of this environment provides built-in stochasticity since there are (theoretically) infinitely

	min	25%	50%	75%	max
Token Count	48616	67248	110000	1061520	42977805
Line Count	999	5765	10000	10000	2865187
Tokens per Line	5.87	7.00	11.00	33.53	7212.72
Tokens per Line SD	0.00	0.00	2.31	13.81	445.84
Unique Tokens	2	7	10	20	902
Unique Lines	18	1253	2440	4911	309405
1-gram Entropy	0.36	2.12	2.80	3.37	6.60
1-gram Normalized Entropy	0.16	0.71	0.82	0.90	1.00
2-gram Entropy	0.42	3.16	4.11	5.88	12.88
2-gram Conditional Entropy	0.06	0.85	1.41	2.54	6.29
Entropy per Line	4.38	21.23	30.80	71.85	30233.52

Table 3.2: Five-number summary of the analyses across corpora of ELCC. Entropy in bits.

many distinct arrangements of the environment that are possible, allowing for more natural variability in the resulting language.

Social deduction "RLupus: Cooperation through the emergent communication in The Werewolf social deduction game" (Brandizzi et al., 2022a, GPL-3.0 license) introduces an ECS based on the social deduction game *Werewolf* (a.k.a., *Mafia*) where, through successive rounds of voting and discussion, the "werewolves" try to eliminate the "villagers" before the villagers figure out who the werewolves are. In a given round, the discussion takes the form of all agents broadcasting a message to all other agents after which a vote is taken on whom to eliminate. As there are multiple rounds in a given game, this system introduces multi-step as well as multi-speaker dynamics into the language. Furthermore, the messages also influence distinct actions in the system (i.e., voting). These additional features in the system add the potential for communication strategies that are shaped by a variety of heterogeneous factors rather than simply the distribution of observations (as in the signaling game).

3.5 Analysis

In this section we give present a brief set of analyses that demonstrate some of the possible insights that can be gained from ELCC. Table 3.2 shows the five-number summary of the corpus-level metrics in ELCC. The corpora come in all shapes and sizes, so to speak, demonstrating a wide range of token counts, vocabulary sizes, entropies, and so on. The variety, in large part, comes from the diversity of systems included in ELCC rather than variation within a system. Thus research focusing on a single or narrow range of emergent communication systems—the norm prior to ELCC—restricts itself to a limited diversity of corpus "shapes"; ELCC, in turn, provides an easy opportunity to expand the breadth of many such approaches.

The range of analyses ELCC enables is greatly multiplied by a resource like XferBench (Chapter 4), a deep transfer learning-based evaluation metric for emergent languages. This

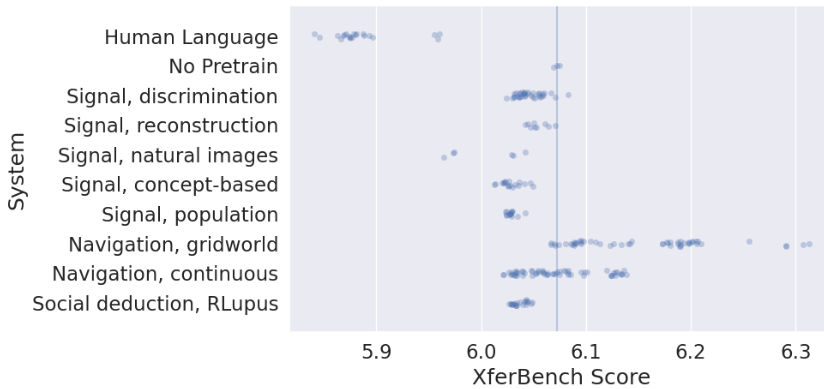


Figure 3.2: XferBench score across ELCC and human language baselines; lower is better. "No pretrain" baseline illustrated with the line on the plot.

```
[47, 2466, 47, 3923, 3325, 3107, 3350, 3923,
    1216, 3980, 1617, 3350, 1897, 556, 0]
[3925, 3925, 3925, 3325, 1172, 2530, 3925, 1209,
    3493, 665, 512, 3923, 2432, 309, 0]
[2128, 2128, 2371, 3925, 946, 512, 1962, 1288,
    2250, 1722, 1722, 1962, 3755, 2695, 0]
```

(a) Best-performing: signaling game (Yao et al., 2022) with the COCO dataset.

```
[3, 3, 3, 3, 3, 3, 3, 3, 7, 7, 7, 7, 7, 7, 7, 7]
[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
[3, 3, 3, 3, 3, 3, 3, 3]
```

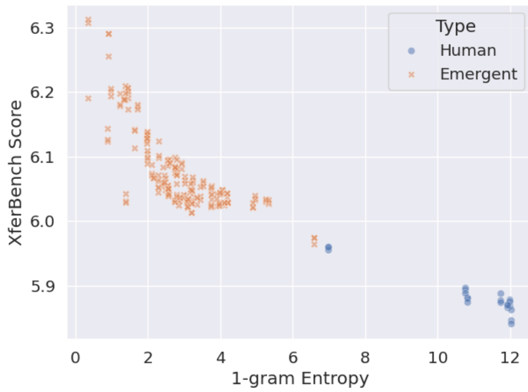
(b) Worst-performing: BabyAI-based navigation game (Unger et al., 2020) (hyperparameters in text).

Figure 3.3: Sample utterances from the best and worst performing emergent language corpora on XferBench from ELCC.

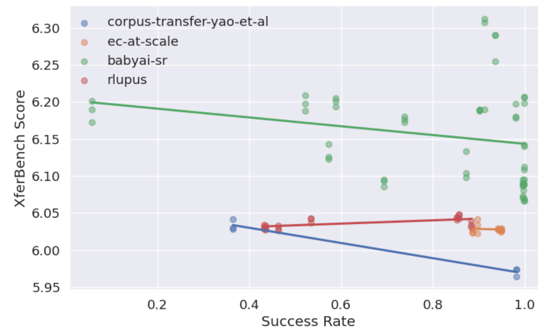
metric quantifies how good a corpus is as pretraining data for a human language-based downstream task, specifically language modeling (thus a lower score is better). XferBench proves to be particularly powerful for analyzing ELCC because it works in an environment-agnostic way, taking only a corpus of tokenized utterances as input. In fact, ELCC and XferBench permit the first large-scale comparison of emergent language systems with an *evaluative* metric.

Explaining XferBench's performance In addition to the purely descriptive metrics discussed above, we also present evaluative metrics via XferBench in Figure 3.2. We run XferBench three times for each corpus since there inherent stochasticity in XferBench. We see that most of the emergent languages occupy a band which slightly outperforms the baselines (i.e., no pretraining at all) while significantly underperforming human languages (exception discussed below). Notably, two of the environments with the worst-performing corpora are the grid-world (Unger et al., 2020) and continuous (Boldt et al., 2022a) navigation environments, while the signaling games perform better consistently.

Inspecting some utterances from the best- and worst- performing corpora, we can see a qualitative difference in Figure 3.3. The best-performing corpus uses a variety of tokens derived from a large vocabulary (given the high token IDs), while the worst-performing corpus repeats the same two tokens with little variation (this sample is representative of the whole corpus). We hypothesize that pretraining on repetitive strings of a small variety of



(a) Plot of XferBench score versus unigram entropy for emergent languages and baseline human languages from XferBench.



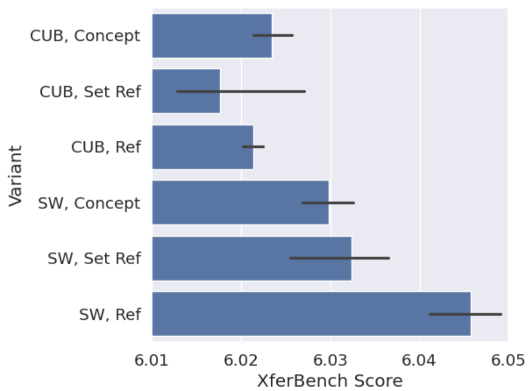
(b) Plot of XferBench score versus success rate, separated by emergent communication system.

Figure 3.4

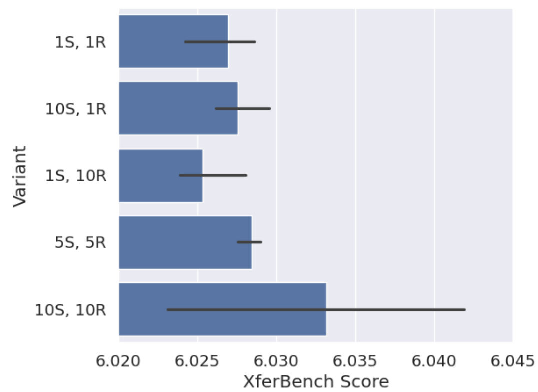
tokens poorly conditions the model used in XferBench, supported by the fact that the lowest entropy corpora perform the worst on XferBench.

The qualitative analysis suggests that something along the lines of variation or information content might be correlated with XferBench score. To investigate this, we plot two possible explanatory variables against XferBench scores: unigram entropy and task success rate Figure 3.4. Immediately, we can see that there is a strong correlation between entropy and XferBench score. In fact, this plot gives some insight into the anomalously low score on "Signal, natural images" (Yao et al., 2022) and anomalously high score for Hindi (an unresolved quandary of the XferBench paper): both of these corpora perform as expected given their entropies. On the other hand, success rate does not seem to be well-correlated with score on XferBench; surprisingly enough, the worst-performing corpus shown above still sported a >90% task success rate!

Evaluating improvements in ECS design Finally, we are also able to use XferBench and ELCC to evaluate some of the innovations in emergent communication system design made by papers contributing to ELCC. Namely, we look at J. Mu et al. (2021) and "Emergent Communication at Scale" (Chaabouni et al., 2022). J. Mu et al. (2021) introduce (as discussed in Section 3.4.2) a more sophisticated, concept-focused version of the signaling game, comparing it against a vanilla signaling game ("reference") and an intermediate form of the concept version ("set reference"), finding that the introduced games promote more systematic and interpretable emergent languages. On the other hand, Chaabouni et al. (2022) introduces multi-agent populations to the signaling game but does not find that larger populations have a beneficial effect on communication. Looking at the systems' performance XferBench (Figure 3.5), we can see that the proposed improvements to the signaling game do not have an appreciable effect on XferBench performance in either case. These results do not detract from the original findings; instead, evaluating the design changes with XferBench better contextualizes work, highlighting to what degree certain desirable features of emergent



(a) Expected order: concept, set reference, reference (J. Mu et al., 2021).



(b) # of senders, # of receivers; more agents expected to perform better than fewer (Chaabouni et al., 2022).

Figure 3.5: XferBench scores compared to expected order; lower is better.

languages (e.g., interpretability, robustness) correspond with suitability for deep transfer learning.

3.6 Discussion

Work enabled by ELCC In the typical emergent communication paper, only a small amount of time and page count is allocated to analysis with the lion's share being taken up by designing the ECS, implementing it, and running experiments. Even if one reuses an existing implementation, a significant portion of work still goes towards designing and running the experiments, and the analysis is still limited to that single system. While this kind of research is valid and important, it should not be the only paradigm possible within emergent communication research. To this end, ELCC enables research which focus primarily on developing more in-depth analyses across a diverse collection of systems. Furthermore, removing the necessity of implementing and/or running experiments allows researchers without machine learning backgrounds to contribute to emergent communication research from more linguistic angles that otherwise would not be possible.

In particular, ELCC enables work that focuses on the lexical properties of emergent communication, looking at the statical properties and patterns of the surface forms of a given language (e.g., Zipf's law (Zipf, 1950)). Ueda et al. (2023) is a prime example of this; this paper investigates whether or not emergent languages obey Harris' Articulation Schema (HAS) by relating conditional entropy to the presence of word boundaries (Harris, 1955; Tanaka-Ishii, 2021). The paper finds mixed evidence for HAS in emergent languages but only evaluated a handful of settings in a single ECS, yet it could be the case that only systems with certain features generate languages described by HAS. The variety of systems provided by ELCC could, then, provide more definitive empirical evidence in support or against the presence of HAS in emergent languages. Additionally, ELCC can similarly extend the range of emergent languages evaluated in the context of machine learning, such as Yao et al. (2022) and Chapter 4 which look at emergent language's suitability for deep transfer learning to downstream NLP tasks or Wal et al. (2020) which analyzes emergent languages with unsupervised grammar induction.

ECS implementations and reproducibility In the process of compiling ELCC, we observed a handful of trends in the implementations of emergent communication systems. A significant proportion of papers do not publish the implementations of experiments, severely limiting the ease of reproducing the results or including such work in a project such as ELCC, considering that a large amount of the work in creating an ECS is not in the design but in the details of implementation. Even when a free and open source implementation is available, many projects suffer from underspecified Python dependencies (i.e., no indication of versions) which can be difficult to reproduce if the project is older than a few years. Furthermore, some projects also fail to specify the particular hyperparameter settings or commands to run the experiments presented in the paper; while these can often be recovered with some investigation, this and the above issue prove to be obstacles which could easily be avoided. For an exemplar of a well-documented, easy-to-run implementation of an ECS and its experiments, see J. Mu et al. (2021) at `https://github.com/jayelm/emergent-generalization/` which not only provides dependencies with version and documentation how to download the data but also a complete shell script which executes the commands to reproduce the experiments.

Future of ELCC While ELCC is a complete resource as presented in this paper, ELCC is intended to be an ongoing project which incorporates further ECSs, analyses, and taxonomic features as the body of emergent communication literature and free and open source implementations continues to grow. This approach involves the community not only publishing well-documented implementation of their ECSs but also directly contributing to ELCC in the spirit of scientific collaboration and free and open source software. ELCC, then, is intended to become a hub for a variety of stakeholders in the emergent communication research community, namely a place for: ECS developers to contribute and publicize their work, EC researchers to stay up-to-date on new ECSs, and EC-adjacent researchers to find emergent languages which they can analyze or otherwise use for their own research.

Limitations Emergent communication research is primarily basic research on machine generated data; thus, ELCC has few, if any, direct societal impacts. From a research point of view: while ELCC attempts to provide a representative sample of the ECSs present in the literature, it is not comprehensive collection of all of the open source implementations let alone all ECSs in the literature. This limitation is especially salient in the case of foundational works in EC which have no open source implementations (e.g., Mordatch et al. (2018)). Thus, the contents of ELCC could potentially result in an over-reliance on the particular systems included resulting in an unfamiliarity with the data and limiting research on those currently not included in ELCC. Including the data-generating code and metadata describing the systems in ELCC has partially addressed this issue, and future work adding more open source implementations and reimplementing seminal papers could continue to ameliorate this limitation.

Beyond the variety of systems, in its design ELCC only provides unannotated corpora without any reference to the semantics of the communication, which limits the range of analyses that can be performed. For example, measures of compositionality, such as topographic similarity (Brighton et al., 2006; Lazaridou et al., 2018), are precluded because they fundamentally a relationship between surface forms and their semantics. In terms of compute resources, we estimate that on the order of 150 GPU-hours (NVIDIA A6000 or equivalent) on an institutional cluster were used in the development of ELCC, and additional 1000 GPU-hours were used to generate the results of XferBench on ELCC. This research could be difficult to reproduce without access to institutional resources.

3.7 Conclusion

In this paper, we have introduced ELCC, a collection of emergent language corpora annotated with taxonomic metadata and suite of descriptive metrics derived from free and open source implementations of emergent communication systems introduced in the literature. ELCC also provides code for running these implementations, in turn, making those implementations more reproducible. This collection is the first of its kind in providing easy access to a variety of emergent language corpora. Thus, it enables new kinds of research on emergent communication which involve a wide range of emergent communication, focusing directly on the analysis of the emergent languages themselves.

Chapter 4

Evaluation with Deep Transfer Learning

Abstract

In this chapter, we introduce XferBench, a benchmark for evaluating the overall quality of emergent languages using data-driven methods. Specifically, we interpret the notion of the "quality" of an emergent language as its similarity to human language within a deep learning framework. We measure this by using the emergent language as pretraining data for a downstream NLP tasks in human language—the better the downstream performance, the better the emergent language. We implement this benchmark as an easy-to-use Python package that only requires a text file of utterances from the emergent language to be evaluated. Finally, we empirically test the benchmark's validity using human, synthetic, and emergent language baselines.[1]

4.1 Introduction

Neural language models learn many things in pretraining, but research suggests (Artetxe et al., 2020) that a substantial part of that knowledge is not simply knowledge of a particular language or domain, but rather knowledge of "how to language." We currently teach models to "language" using vast quantities of text dredged from the dark recesses of the Web—text that is full of bias, toxicity, and potential intellectual property violations. Ideally, we would be able to teach models to "language" without such compromises through the use of synthetic data, but mainstream approaches to synthesizing data produce outputs that do not have the same structural and social properties as human language.

Emergent communication (EC), also called emergent language (EL), is a potential solution to this problem (Yao et al., 2022; Downey et al., 2023; Y. Mu et al., 2023). Emergent languages are communication systems developed *de novo* among multiple agents in a reinforcement learning simulation. Because the conditions under which they develop mirror, reductively, the conditions under which languages develop among humans, there is reason to believe that ELs will ultimately be more like human language than other sources of synthetic data. However, up to this point, there is no way of quantifying—in a holistic way—how much like human

[1] Based on "XferBench: a Data-Driven Benchmark for Emergent Language" appearing in the *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (Boldt et al., 2024c).

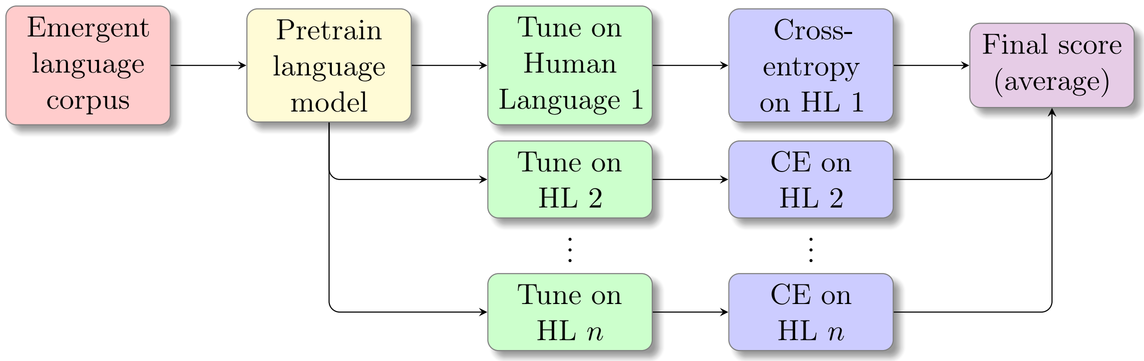


Figure 4.1: Illustration of the architecture of XferBench.

languages any particular EL really is, or to what extent it may provide useful pretraining signals.

Research on deep learning-based emergent communication has seen the introduction of many metrics to measure various aspects of the language. These metrics quantify notions such as compositionality (Brighton et al., 2006; Lazaridou et al., 2018), expressivity (Y. Guo et al., 2023), ease-of-teaching (F. Li et al., 2019), and zero-shot transfer (Bullard et al., 2020), to name a few. Despite this proliferation of metrics, emergent language largely lacks *evaluation* metrics. An evaluation metric is specifically one that measures the *overall quality of an emergent language* and not simply a particular property. Thus, we introduce XferBench, a data-driven benchmark for evaluating the overall quality of emergent languages using transfer learning with deep neural models.

Evaluation metrics are critical in gauging progress in technical fields since they quantify otherwise vague notions of improvement over time. Benchmarks, in particular, pair evaluation metrics with specific data and evaluation procedures to compare various systems on common ground. Benchmarks and shared tasks have been critical to the development of NLP from the Penn Treebank (Marcus et al., 1993) to the WMT datasets (O. Bojar et al., 2014) to GLUE (A. Wang et al., 2018).

In the field of emergent communication specifically, Yao et al. (2022) introduced the idea of using *corpus transfer* as means of practically applying emergent communication to deep learning-based NLP via transfer learning. In corpus transfer, a language model is pretrained on a corpus of emergent language utterances before being tuned on real data for a human language-based downstream task. As a corollary, they suggest that the effectiveness of this transfer can serve as a means of evaluating the quality of the emergent in a more general sense. This is based on the intuition that the more similar two language are, the better transfer learning works from one to the other (observed in Zoph et al. (2016), for example).

This chapter takes the transfer learning-as-an-evaluation metric idea from Yao et al. (2022) and expands it into a full benchmark, XferBench, for emergent languages (illustrated in Figure 4.1). An evaluation metric for emergent languages in a benchmark format is the first of its kind. Additionally, XferBench is unique within emergent communication for being primarily data-driven instead of relying on particular handcrafted algorithms for quantifying a given phenomenon. This means that XferBench can be easily scaled up in the future as the field of emergent communication advances and requires expanded means of evaluating emergent languages. Finally, XferBench is distributed as a user-friendly Python package, allowing researchers from across the field of emergent communication to apply XferBench to their own work on emergent communication.

Contributions This chapter makes the following contributions: (1) Introduces XferBench, a data-driven benchmark for evaluating the overall quality of an emergent language, the first of its kind in emergent communication. (2) Provides a analysis of the quality human, synthetic, and emergent language according to XferBench. (3) Provides an easy-to-use Python implementation of XferBench.

4.2 Related Work

Emergent Communication This chapter is situated in the field of emergent communication (a.k.a. emergent language) which is generally covered by the review Lazaridou et al. (2020). The field centers around the invention of language by deep neural networks typically using multi-agent reinforcement learning techniques. The study of emergent communication is intended to (1) shed light on the origin and nature of the human language (LaCroix, 2019; Moulin-Frier et al., 2020; Galke et al., 2022) and (2) provide an alternative approach to problems in NLP and multi-agent reinforcement learning which relies on constructing language from the ground up and not just pre-existing (human) languages alone (Yaoyiran Li et al., 2020; Yao et al., 2022; Y. Mu et al., 2023; Downey et al., 2023).

Transfer Learning Transfer learning for deep neural networks is a key component of XferBench and follows in general tradition of Zoph et al. (2016). Specifically, this chapter draws heavily from Yao et al. (2022) (see also Papadimitriou et al. (2020) and Artetxe et al. (2020)) which introduce the technique of *corpus transfer* for emergent language, that is, pretraining a neural model on an emergent language corpus before tuning it on a downstream human language task. In particular, this chapter takes Yao et al. (2022)'s idea of using corpus transfer as a metric and adapts it into a benchmark pipeline which can easily be applied to new emergent languages.

Benchmarks Work such as Y. Guo et al. (2023) and Perkins (2022) have looked at benchmarking particular aspects of emergent languages, but XferBench is the first of its kind in benchmarking the overall quality of an emergent language. Yao et al. (2022) also explicitly provide a metric for emergent language quality, but this metric is restrictive in that it can only be applied to emergent languages derived from a model that takes images (that have captions available) as input; this conflicts with the design goals of XferBench discussed below.

Outside of emergent communication, XferBench is more analogous to benchmarks for generative models (e.g., Fréchet Inception Distance (Heusel et al., 2017) for image generation) than more traditional NLP benchmarks like GLUE (A. Wang et al., 2018) or SQuAD (Rajpurkar et al., 2016). This is because emergent communication is a generative enterprise, where one of the main goals is to create samples (emergent languages) which resemble a target distribution (human languages) either generally or in some particular respect. Furthermore, metrics like FID are primarily self-supervised, data-driven measures of similarity in the same vein as XferBench. This is in contrast to more traditional NLP benchmarks which combine data-driven methods with many human judgments (i.e., through labeled examples).

4.3 XferBench

4.3.1 Design Goals

We frame the primary design goals of the benchmark as three desiderata:

- **D1** Quantitatively capture a meaningful notion of the overall quality[2] of an emergent language from a data-driven perspective.
- **D2** Be applicable to as wide a variety of emergent languages as possible, not restricted to a specific game, environment, or agent architecture.
- **D3** Be relevant and accessible to the broader EC/EL community, by being: (a) easy to interpret, (b) minimally biased with regards to language typology, (c) runnable with minimal coding experience, and (d) runnable on modest hardware.

While there are other consideration in the benchmark, these form the bulk of the motivation. In the following paragraphs we expand upon the motivation for each design goal.

D1: Quantifying quality D1 is the core of what a benchmark seeks to do: to quantify a desirable property of a given system such that it can be compared directly to other systems (i.e., be an *evaluation* metric). There are two distinct senses in which XferBench strives towards this goal. First, XferBench measures how good an emergent language is from a specifically machine learning perspective; that is, it addresses the question, "How useful would this emergent language be for practical machine learning tasks?" The second sense is more general: XferBench addresses the question, "How similar is an emergent language to human language according to how deep neural networks process language?" That is, it uses data-driven techniques to quantify the similarity between emergent language and human language in some general sense.

D2: Wide applicability D2 is intended to make XferBench practically applicable to a wide range of EC research. The field of EC has an especially diverse set of possible approaches, environments, agents, games, etc. Thus, it is especially salient that the benchmark be designed with interoperability in mind, having minimal assumptions as to the nature of the EC system being evaluated.

The influence of this design goal is primarily seen through the use of a textual corpus as the sole input to the benchmark: the vast majority of EC systems generate utterances which can be represented as sequences of discrete tokens.[3] EC presents the opportunity for much richer representations of its language: leveraging the grounded semantics of the communication, incorporating non-verbal behavior, and even directly interacting with the agents themselves. Yet such richer representations also limit the range of EC systems to

[2]We are aiming for *a* meaningful notion of overall quality: we are not claiming that this is the only meaningful notion nor that it is the best among all possible notions of "quality".

[3]In the minority case, there are EC methods which use communication channels that are, for example, continuous (Eloff et al., 2021) or even pictorial (Mihai et al., 2021a).

which XferBench could apply. Even if it is possible to define some universal EC interface that could allow for richer representations, the implementation cost for each and every EC system to be tested is significant compared to the ease of producing a corpus of utterances from the emergent language.

D3: Easy-to-use D3 is critical to the success of XferBench as a practical tool for diverse field of researchers—a benchmark is expressly *for* the broader research community, and, as such, should be widely accessible. In particular, D3a demands that XferBench be conceptually simple with results that can easily be reported, compared, and incorporated into a research program. D3b is relevant to both aspects of D1. First, if XferBench is to gauge an EL's practical use in machine learning, it should seek to use a typologically diverse set of human languages in the downstream tasks. Second, since XferBench is trying to capture a notion of "similarity to human language generally", it is important to test this against a wide range of language typologies so as not to unnecessarily narrow the criteria for "similar to human language". D3c is particularly important for incorporating interdisciplinary researchers into the field of EC who might not have a background in computer programming. Finally, D3d ensures that XferBench is accessible not only to labs and researchers with fewer financial resources but also makes it much easier to incorporate into the fast-paced research and development cycles prevalent in contemporary ML reserach.

4.3.2 Methods

The following procedure describes the benchmark (illustrated in Figure 4.1):

1. Initialize a causal language model.
2. Train the model on the corpus of utterances from the EL being evaluated.
3. Re-initialize the input and output (i.e., language modeling head) embedding layers; this is the *base model*.
4. For each downstream human language:
   a) Train the base model on the human language data.
   b) Evaluate the cross-entropy on a held-out test set of the human language.
5. Average the cross-entropies across the downstream human languages; this is the corpus's score on the benchmark (lower is better).

The structure of the benchmark is derived from the *corpus transfer* method presented in Yao et al. (2022).

Task For XferBench's evaluation task, we choose causal language modeling for a few different reasons. In principle, language modeling is a component of a wide variety of NLP tasks, especially generative tasks; the prevalence of language modeling is in line with the benchmark providing a very general notion of quality that will be familiar to anyone acquainted with NLP. On a practical level, language modeling is easy to acquire data for—especially helpful for evaluating against low-resource languages—and there are fewer hyperparameters and confounding variables compared to other downstream tasks like machine translation or question-answering. The main limitation from using language modeling is that it itself is not a widespread downstream task and so cannot guarantee direct correlation with metrics on more concrete downstream tasks (e.g., accuracy on a QA task).

For the pretraining task we also use causal language modeling. Due to requiring a wide applicability across emergent languages (Design Goal 2), we select causal language modeling for our pretraining task since it requires only a corpus without any additional annotations or stipulations.

Data The data for the transfer learning targets (viz. human languages) comes from Wikipedia dumps (Wikimedia Foundation, n.d.) (under the GFDL and CC-BY-SA 3.0 License) hosted by Hugging Face[4]. This dataset provides a diverse set of languages each with sufficient amounts of data. For our downstream human languages, we use the same 10 languages presented in Yao et al. (2022), namely: Basque, Danish, Finnish, Hebrew, Indonesian, Japanese, Kazakh, Persian, Romanian, and Urdu. Having a variety of languages reduces the likelihood that XferBench will be biased toward specific typologies of human language (Design Goal 3b).

We use 15 and 2 million tokens for the pretraining and fine tuning phases, respectively following Yao et al. (2022). Datasets are always repeated or truncated to fit the required size so that the number of training steps stays constant.

Tokenization For tokenization we use byte pair encoding (BPE) (Gage, 1994) with a vocabulary size of 30 000 for all human languages. Using BPE across all human languages is done primarily to simplify the implementation and keep tokenization methods consistent across all of the selected human languages. Emergent languages are generally considered to be pre-tokenized since most communication channels consist of one-hot vectors; thus, no additional tokenization or preprocessing is applied.[5]

Model For our model, we use a small configuration of GPT-2 (Radford et al., 2019), similar to that used in Yao et al. (2022): 6 attention heads, 6 layers, context length of 256, and hidden size of 768 with the remainder of the model parameters being the same as the defaults in the Hugging Face Transformers implementation.[6] This yields 65 million parameters in total. We kept the model on the smaller size to better suit it for the generally small amounts of data emergent languages corpora provide as well as to be more accessible (Design Goal 3d). Further details are listed in Section C.1.1.

Metric Given the use of language modeling for our evaluation task, we use token-level cross-entropy as the evaluation metric on the downstream task. This is a very common metric, making the outputs easy to interpret (Design Goal 3a). Although perplexity is more common as an evaluation of language models, the exponential nature of perplexity leads to more

[4]`https://huggingface.co/datasets/wikimedia/wikipedia/tree/97323c5edeffcf4bd6786b4ed0788c84abd24b03`

[5]Whether the tokens of an EL should be treated as words or subword units is an open question, although tokens as words is more common (but see Ueda et al. (2023) for tokens as subword units). Practically speaking, many emergent languages are small enough that applying a 30 000-item BPE model would severely reduce the corpus size.

[6]`https://huggingface.co/docs/transformers/v4.36.1/en/model_doc/gpt2#transformers.GPT2Config`

circuitous analyses and interpretation in our case, whereas cross-entropy is comparatively linear and additive (loosely speaking).[7] For the final score of the benchmark, we take the arithmetic mean of the cross-entropy across the 10 downstream human languages. That is, we define the benchmark's score for a given source language s as as h_s:

$$h_s = \operatorname*{mean}_{t \in T} (h_{s,t}) \tag{4.1}$$

where $h_{s,t}$ is the test cross-entropy of a model trained on source language s and finetuned and tested on target language t; T is the set of target languages. Since the score is based on cross-entropy, a lower score means better performance.

4.3.3 Implementation

XferBench is implemented as a small Python codebase which relies primarily on Hugging Face Transformers (Wolf et al., 2020) (Apache-2.0 license) and PyTorch (Paszke et al., 2019) (BSD-3-Clause license) libraries. To run the benchmark, all that is required is to install the environment with either pip or conda, and run `python -m xferbench path/to/corpus.jsonl` (Design Goal 3c). The input corpus is simply formatted as a newline-separated list of integer arrays, specifically in the JSON Lines format (see Section C.2 for an example); a Hugging Face dataset (backed by Apache Arrow) can also be used for larger input corpora. The script executes all of the steps of the benchmark and yields a single floating point number which is that corpus's score on XferBench (the benchmark also saves the individual score across target languages for further analysis). Finer-grained functionalities are available and documented in the codebase. The benchmark takes about 5.5 hours to run on a single NVIDIA GeForce RTX 2080 Ti: 90 minutes to train the base model and 30 minutes for tuning and testing on each of the target languages (Design Goal 3d). Since the model is tuned independently on each target language, it is easy to parallelize this step and drastically shorten the wall-clock time of XferBench.

The implementation is available at `https://github.com/brendon-boldt/xferbench` under the MIT license.

4.4 Experiments

4.4.1 Procedures

XferBench The causal language modeling experiment is simply running XferBench as described in Section 4.3.2 on the reference and emergent languages discussed in Sections 4.4.2 and 4.4.3.

Machine translation The machine translation experiment is structured similarly to XferBench except with the downstream task being English-to-French translation (using the WMT 2014 dataset (O. Bojar et al., 2014)). The primary purpose of this experiment is to

[7] For example, it would make more sense to use logarithmic scales and geometric means to average and compare perplexities, but this would just be reverting back to cross-entropy!

determine how well XferBench correlates with a more concrete downstream task (especially one that incorporates language modeling). We choose this language pair in part to gauge the relative differences between the task languages and the baseline human languages (in contrast to XferBench which we want to be largely agnostic to human languages). Looking at our reference human languages, we have: French, the target language itself; Spanish, closely related to French; Russian and Hindi, distantly related to French; and Chinese, Korean and Arabic, not related to French. Instead of using a GPT-2–based model, we use a BART-based model since MT is a conditional generation task (see Section C.1.2 for details). The pretraining dataset size is increased to 100 million due to the increased difficulty of this task compared to language modeling. We evaluate the translation performance with chrF (Popović, 2015) and BLEU (Papineni et al., 2002) using the default Hugging Face Evaluate metrics (derived from sacreBLEU (Post, 2018)). Evaluation is performed with beam sizes of 1, 3, and 5, and the resulting values are averaged.

We present three settings for this experiment. The first is *Full* which tunes on 50 million source tokens at a higher learning rate ($1 \cdot 10^{-4}$ for training and $2 \cdot 10^{-4}$ for the AdamW optimizer (Kingma et al., 2015)), which we found empirically to lead to the best performance. The second is *Frozen*, in which we use the same configuration as *Full* but freeze all but the embedding layers before tuning the model for translation (as in Papadimitriou et al. (2020) and Artetxe et al. (2020)). Finally, we also present *Reduced* which uses a smaller tuning dataset of 10 million tokens and lower learning learning ($2 \cdot 10^{-5}$); the lower rate helped the random baselines converge better as well as showed better distinction between languages.

4.4.2 Reference languages

The following reference languages serve as a way to contextualize the results of XferBench as well as to validate that it is capturing some notion of the quality of the emergent languages (cf. Section 4.4.4).

Human languages For our baseline line human languages, we selected French, Spanish, Russian, Chinese, Korean, Arabic, and Hindi.[8] Like the evaluation languages, the data is derived from Wikipedia articles (same source as the target languages).

Synthetic languages For synthetic languages, we follow Yao et al. (2022) and use “Zipfian parentheses” from Papadimitriou et al. (2020). This synthetic dataset—referred to as *Paren, real*—is hierarchically balanced “parentheses” where each parenthesis is the token ID sampled from the unigram distribution of a human language (hence “Zipfian”). This datasets mimics both the unigram distribution of a human language as well as the basic recursive hierarchical structure. This yields a reasonably strong yet simple baseline for synthetic data.

We also test a fully synthetic dataset (*Paren, synth*) which uses the same hierarchical parenthesis generation script from Papadimitriou et al. (2020), replacing the data-derived

[8]The main reason for choosing the high-resource language is due to the higher data requirements of machine translation experiment discussed below.

Setting	Observ.	$\lvert V\rvert$	$\lvert M\rvert$	$\lvert C\rvert$
Disc, small	one-hot	6	11	700
Disc, large	one-hot	100	31	100 M
Recon, large	one-hot	100	31	31 M
Mu+, CUB	embed	20	10	1.3 M
Mu+, SW	embed	14	7	1.2 M
Yao+	embed	4028	15	43 M

Table 4.1: Summary of key hyperparameters in the tested emergent languages. Observations are either one-hot vectors or embeddings. $|V|$, $|M|$, and $|C|$ refer to the vocabulary, message, and corpus size respectively.

unigram distribution with Zipf–Mandelbrot distribution:

$$f(w_i) = \frac{1}{(i+\beta)^\alpha} \tag{4.2}$$

where $f(w_i)$ is non-normalized probability weight of word w with 1-based index (rank) i, $\alpha = 1$, $\beta = 2.7$ (Mandelbrot et al., 1953; Piantadosi, 2014).

Random baselines We use two random baselines. The first is simply a uniform unigram distribution across the whole vocabulary with no additional structure (referred to as *Random*). This baseline sheds light on whether the optimization itself, no matter training data, primes the network in some way for transfer learning. The second "random" baseline is no pretraining at all (*No pretrain*); that is, a network which has been freshly initialized at the tuning stage. This baseline helps establish whether or not pretraining on other languages has any impact beyond tuning alone.

4.4.3 Emergent languages

We present a summary of the key hyperparameters of emergent languages in Table 4.1. The emergent language corpora below come from reproductions from existing codebases with the exception of Yao et al. (2022), whose emergent language corpus is available for download. Emergent languages which have a corpus size smaller than the required size are simply repeated and shuffled as many times as necessary so that the model receives the same number of optimization steps.

Generic signaling game The first set of emergent languages we test are generic versions of the of the signaling game (reference game) as implemented in EGG (Kharitonov et al., 2019) (MIT license). These games use one-hot vectors to represent attribute–value observations, that is, observations are elements of the set $V^{|A|}$ where V is the set of values and $|A|$ is the number of attributes. The signaling game is one of the simplest and most used games in emergent communication research.

The first two language are *Disc, small* and *Disc, large* which are two configurations of the discrimination version of the signaling game. Here, the sender makes an observation and sends a message; then, the receiver must select the corresponding observation from a small set of potential observations (like a multiple-choice question). The *small* configuration consists of 4 attributes and 4 values with a small vocabulary size and medium message length; this setting is intended to represent a toy environment that one might find in an emergent communication paper. The *large* configuration consists of 12 attributes and 8 values with a larger vocabulary and longer message length. Both environments show 5 distractor observations to the receiver (i.e., 6-way multiple choice). Both settings converge to a success rate >95% compared to a random baseline of 17%.

The *Recon, large* environment is based on the reconstruction version of the signaling game. In this version, the receiver does not make any observations and instead must recreate the sender's observation based on the message alone (similar to an autoencoder). The observation space has 8 attributes and 8 values with other settings identical to that of *Disc, large*. Since the reconstruction game considerably harder, the game does not converge but does reach an overall accuracy of 0.014% and per-attribute accuracy of 24% compared to a random baseline of 0.000006% and 13% random baseline, respectively. For details, see Section C.1.3.

J. Mu et al. (2021) present the second pair of emergent languages which we test XferBench on (code under MIT license). The emergent communication game is also a discriminative signaling game but with (1) richer observations and (2) more abstract information needing to be communicated. In one setting, the observations are images from ShapeWorld (Kuhnle et al., 2017) (*Mu+, SW*), a synthetic data of various geometric shapes, and the other setting is CUB (Wah et al., 2011) (*Mu+, CUB*) which contains labeled images of birds; both settings encode features with a CNN which is the passed to the sender and receiver. In the basic discriminative game, the observation made by the sender is the exact same one seen by the receiver. J. Mu et al. (2021) instead uses a "concept game" where the sender must communicate some abstract concept shared by a set of input images which the receiver will then have to a pick out from a different set of images, some sharing the same concept (e.g., isolating the concept of "triangle" or "bird size"). The ShapeWorld and CUB games had test accuracies of 71% and 66% respectively compared to a random baseline of 50%, comparable to the reported values in the paper. All messages were taken from observations seen in training.

Yao et al. (2022) present a standard discrimination game which uses natural images (Conceptual Captions (Sharma et al., 2018) (images only)) as inputs to the sender and receiver (code unlicensed but distributed on GitHub with paper). The accuracy for the particular emergent language corpus is not reported in the paper, but comparable experiments from the paper would suggest that it converged to an accuracy of >90% compared to a baseline of 0.4% (i.e., 255 distractors).

4.4.4 Hypotheses

The following hypotheses are directly relate to determining whether or not XferBench is quantifying some meaningful notion of the quality of a language (i.e., Design Goal 1).

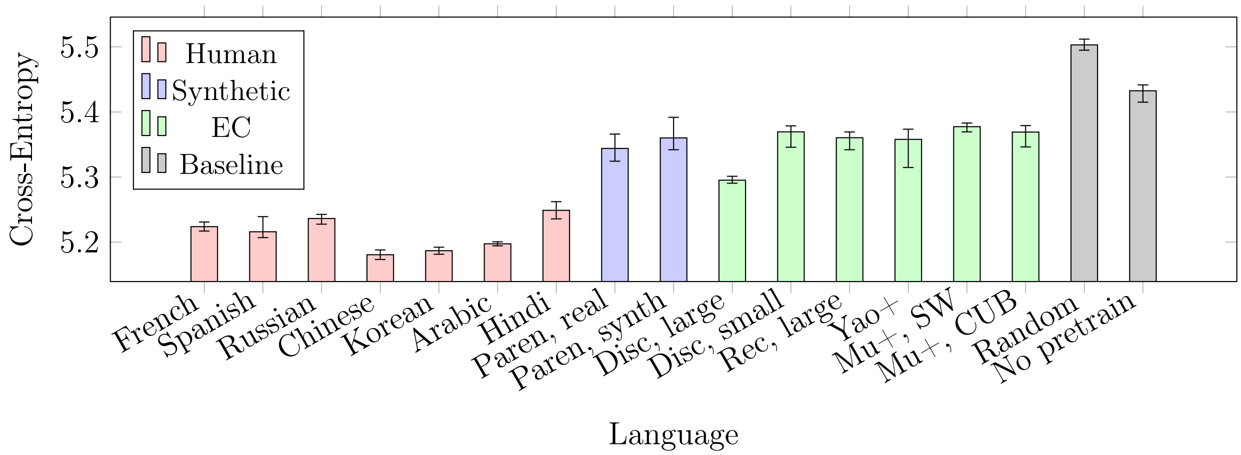


Figure 4.2: Average cross-entropy on target language datasets for each source language. Lower is better. Error bars represent 95% confidence intervals.

(H1) Human languages will perform best, followed by the synthetic and emergent languages, followed by the random baselines.

(H2) Human languages will have similar performance on XferBench (also key for Design Goal 3b); the intuition here is that human languages share deep structural similarities. This hypothesis is supported, in part, by Artetxe et al. (2020). For the MT experiment, we expect to see the following order of performance based on language relatedness: {*French*}, {*Spanish*}, {*Russian*, *Hindi*}, {*Chinese*, *Korean*, *Arabic*}.

(H3) Languages with a larger vocabulary, longer message length, and larger corpora will perform better. In particular, we expect *Disc, large* will perform better than *Disc, small* since the former is a more "complex" version of the latter. This hypothesis (for vocabulary size and message length) is supported by some experiments in Yao et al. (2022, app. B.4).

(H4) XferBench will correlate well with scores on the machine translation task (i.e., cross-entropy will correlate negatively with chrF).

4.5 Results

4.5.1 XferBench

In Figure 4.2 we show the results of the benchmark (i.e., causal language modeling) on the various baselines. Each mean is displayed with error bars showing the 95% confidence interval of mean as calculated with bootstrapping (details in Section C.5). For reference, the cross-entropies range from about 5.2 to 5.5 corresponding to perplexities of 180 to 240.

The human languages show the best score (lowest cross-entropy) on the benchmark with *Chinese*, *Korean*, and *Arabic* performing the best in one cluster and *French*, *Spanish*, *Russian*, and *Hindi* performing slightly worse in their own cluster (based on confidence intervals). The synthetic and emergent languages all show similar performance with only small variations with the exception of the *Disc, large* language which is better than the rest of the emergent languages but still worse than the human languages. Finally, the random baselines perform

Source	Full	Frozen	Reduced
French	47.8	31.4	35.8
Spanish	48.0	27.9	34.8
Russian	47.6	29.0	37.2
Chinese	47.5	22.2	35.2
Korean	47.7	23.3	35.6
Arabic	47.8	27.6	36.6
Hindi	47.5	26.0	31.7
Paren, real	47.5	10.5	35.0
Paren, synth	48.2	12.0	34.3
Disc, large	47.7	24.7	30.7
Disc, small	14.3	16.2	17.3
Rec, large	22.5	18.4	25.4
Yao+	4.0	20.1	25.6
Mu+, SW	3.3	18.4	23.3
Mu+, CUB	47.6	21.6	24.6
Random	1.8	3.0	19.7
No pretrain	11.4	4.3	28.1
Correl. with XferBench	−0.75	−0.84	−0.79

Table 4.2: chrF scores across three English-to-French machine translation settings. Correlation measured with the Pearson correlation coefficient. Colors normalized by column.

worse than the rest of the tested languages. *No pretrain*'s performance is worse than the cluster of synthetic and emergent languages but better than the fully random language (*Random*).

4.5.2 Machine Translation

The chrF scores of the machine translation experiment are given in Table 4.2 (BLEU scores in Section C.4.1). Additionally, we give Pearson correlation coefficients between each setting and the scores generated by XferBench (scatter plots shown in Section C.4.4). In all settings, we see that XferBench is strongly correlated with the results of the machine translation experiment.

For the *Full* setting, the results are somewhat inconclusive. Human languages perform the best and similarly to each other. *Paren, real*, *Paren, syn*, *Disc, large*, and *Mu+, CUB* all match the performance of human languages as well. The rest of the language perform significantly worse than the aforementioned languages, especially *Yao+* and *Mu+, SW* (see Section C.6 for sample outputs). In the case of *Random*, the training loss did not decrease during training likely due to the high learning rate.

In *Frozen*, we see the best correlation with the hypothesis regarding human languages (as well as with XferBench itself). *Disc, large* performs comparably to the worst human languages and better than the rest of the languages. The remainder of the synthetic and emergent languages perform worse than the human languages but better than the random baselines.

Finally, *Reduced* (i.e., lower learning rate and tuning data) displays better separation than *Full*, but not as significant as *Frozen*. Human languages still perform the best, although they are matched by the *Paren* languages. *Disc, large* underperforms the human languages but still outperforms all other emergent languages. All emergent languages, apart from *Disc., large* underperform the *No pretrain* baseline. The better half of languages performed better (compared to themselves) with a higher learning rate while the lower half performed better with a reduced learning rate.

4.6 Discussion

4.6.1 Experiments

The basic ordering of the language by XferBench follows basic *a priori* assumptions: random baselines perform the worst, human languages perform the best, and emergent and synthetic languages are bounded above and below by these (supporting Hypothesis 1). Human languages cluster together in XferBench although there is still variation with non-overlapping confidence intervals (partially supporting Hypothesis 2).

Intra-EL differences Generally speaking, there is very little variation shown by XferBench on the emergent languages; nevertheless, we can still draw a handful of conclusions. First, *Disc, large* outperforms *Disc, small* while sharing the same codebase, task, etc. and differing only in message length, vocabulary size, observation space, and corpus size (supporting Hypothesis 3). This result matches the trend seen in Yao et al. (2022) that larger vocabularies and message lengths in an emergent language lead to better performance on downstream data. On the other hand, *Disc, small* performs similarly to other languages with larger vocabularies and longer message lengths (contradicting Hypothesis 3).

Second, it seems that the underlying complexity of the emergent communication game does not directly correlate with XferBench score: the abstract visual reasoning of *Mu+, SW* and *Mu+, CUB* does not lead to it outperform *Disc, small*. Additionally, the richer observations (i.e., image embeddings) of *Mu+, CUB* and *Yao+* also do not, by their mere presence, confer an advantage to the emergent language with respect to XferBench.

Finally, *Disc, large* and *Recon, large* both share hyperparameters in terms of the vocabulary size, message length, and corpus size, yet *Disc, large* shows significantly better performance on XferBench. This indicates that XferBench is not *solely* concerned with surface-level features as we see that the nature of the game (e.g., discrimination versus reconstruction, success rate) is relevant as well.

Correlation with MT The results from the machine translation experiment show strong, though not perfect, (negative) correlation with XferBench (supporting Hypothesis 4). For example, in all cases, *Disc, large* outperforms all other emergent languages. This strongly supports the notion that XferBench performance is predictive of downstream performance on more concrete NLP tasks.

The results from the *Full* setting of the MT experiment do show some correlation with XferBench but fail to show expected trends in other ways. For example, there is no clear ordering among the human languages (e.g., *French* does *not* outperform *Arabic*). Additionally *Yao+* and *Mu+, SW* drastically underperform the other emergent languages and the *No pretrain* baseline. We suspect that these aberrations from expected results come in part due to the high learning rate which cause unstable training or generation. On the other hand, the *Frozen* setting gives us the clearest ordering of human languages that matches with *a priori* expectations; this setting also has the strongest correlation with XferBench scores. The *Reduced* setting shows better correlation than *Full* but is not as clear as *Frozen*.

Random baselines In all of our experiments, the pretraining on random tokens (*Random*) performed notably worse than not pretraining at all (*No pretrain*), suggesting that ill-conditioning the neural network can be a significant hindrance to performing well on XferBench. This is important to note in light of the fact that a perfectly one-to-one compositional language describing uniformly sampled attribute–value vectors would yield a corpus with a uniformly random unigram distribution. This is to say, a fully compositional language, which is often seen as desirable in emergent communication research, could make for a very poor source of pretraining data as shown by *Random*'s performance on XferBench.

This fact along with the observations about sensitivity to learning rate indicates that performance on XferBench is not simply a function of the particular features of the emergent language in relation to the downstream human languages but also a function of the dynamics of optimization (i.e., priming the model to adapt well). Although this increases the difficulty of developing and interpreting a tool like XferBench, it is almost an unavoidable part of deep learning methods.

4.6.2 Future work

We identify three main directions for future work with XferBench. The first direction is determining what XferBench is measuring and how its scores correlate with the different factors of emergent languages. Yao et al. (2022, app. B.4) pursued this on a small scale with factors like vocabulary size and message length, but there exist a host of other factors worth exploring: speaker model size, game design, language entropy, observation modality, etc.

The second direction is more extensively investigating the correlation of XferBench with downstream tasks. We would expect that tasks that rely heavily on a language model—such as automatic speech recognition, abstractive summarization, and generative question-answering—to correlate well with XferBench. On the other hand, tasks that are more focused on classification—such as named entity recognition, sentiment analysis, and multiple choice question-answering—might not correlate as well.

Finally, XferBench would benefit greatly from improved compute efficiency. For example, if the results of XferBench could be replicated with a fraction of the training steps, it could (1) allow for a larger number of downstream languages to be tested which would reduce the size of the confidence intervals, allowing more more precise scoring. And (2), it would open the door to using larger models which would better capture the deeper structures of language and likely correlate better with realistic downstream tasks.

4.7 Conclusion

In this paper we have introduced XferBench, a first-of-its-kind benchmark for evaluating the quality of an emergent language corpus based on its transfer learning performance on human languages. This approach to evaluating emergent language scales with data and compute as opposed to requiring increasingly complex handcrafted rules to measure the desirable qualities of emergent language. We provide empirical results of XferBench across human, synthetic, and emergent languages and demonstrate that these results correlate with downstream performance on a machine translation task. XferBench is implemented as an easy-to-use Python package that will permit researchers in the field to easily apply XferBench to new emergent languages.

4.8 Limitations

The first limitation of XferBench is that it relies on a restricted interface with the emergent communication system. With emergent communication we have access not only to the grounding of all of the utterances of the emergent language but also full access to the agents themselves. Language is fundamentally a contextual phenomenon, so only a small part of it can be understood from looking at corpora in isolation. Thus, although XferBench is much more broadly applicable because of this restricted interface, it is also quite limited in what it can detect from a theoretical point of view.

The other set of limitations we will discuss have to do with the model and data size. First, the model and data size (60 M parameters and 15 M tokens) are quite small by contemporary standards, limiting the direct applicability of results from XferBench to relevant downstream tasks involving large language models, for example. On the other hand, scaling up the models, data, and methods of XferBench comes with its own difficulties. First, it would start to bias the benchmark towards high-resource languages, as only those could provide the necessary data to accommodate larger models. Second, it would make XferBench, which is already relatively slow as a metric (6 GPU-hours) even slower. This would decrease the speed of the iterative design process of emergent communication systems and, thus, the utility of the metric as a whole.

Chapter 5

Optimizing for Statistical Similarity Human Language

Abstract

In this chapter, we design a signaling game-based emergent communication environment to generate state-of-the-art emergent languages in terms of similarity to human language. This is done with hyperparameter optimization, using XferBench as the objective function. XferBench quantifies the statistical similarity of emergent language to human language by measuring its suitability for deep transfer learning to human language. Additionally, we demonstrate the predictive power of entropy on the transfer learning performance of an emergent language as well as validate previous results on the entropy-minimization properties of emergent communication systems. Finally, we report generalizations regarding what hyperparameters produce more realistic emergent languages, that is, ones which transfer better to human language.[1]

5.1 Introduction

Emergent language has tremendous potential to generate realistic human language data for deep learning methods without the need to collect data directly (or indirectly) from humans (Chapter 2). This stems from the fact that emergent language aims to replicate the communicative pressures that drive the development of human language and are hypothesized to explain various patterns observed in linguistics (Scholz et al., 2024). Yet little work has been done to date designing emergent communication systems to generate languages with high statistical similarity to human languages. Such languages could better serve as synthetic human language data for pretraining and evaluating NLP models. Thus, in this paper, we generate emergent languages with a signaling game that have a high degree of similarity to human languages, demonstrating state-of-the-art performance on emergent-to-human language deep transfer learning. Specifically, we use Bayesian hyperparameter search to optimize a signaling game on the XferBench benchmark (Chapter 4).

[1]Based on "Searching for the Most Human-like Emergent Language" appearing in the *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing* (Boldt et al., 2025b).

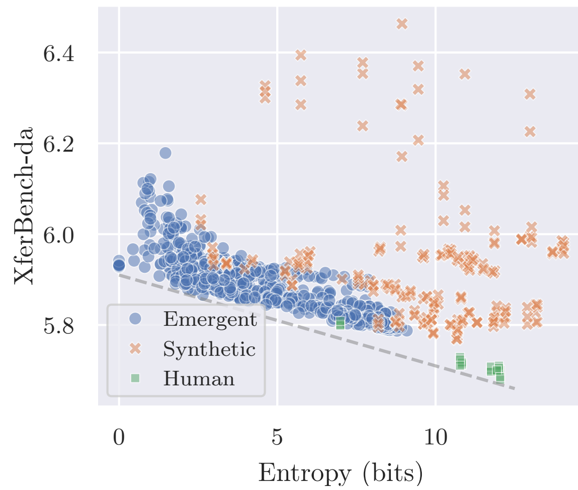


Figure 5.1: Hyperparameter search shows that emergent and human languages tend towards the Pareto frontier of minimizing entropy and minimizing XferBench score (lower is better) while non-emergent synthetic languages less reliably follow this trend. Dashed gray line represents a lower bound on entropy versus XferBench score.

First and foremost, this moves the field of emergent language measurably closer to the goal of providing realistic, fully synthetic data for NLP. On a methodological level, hyperparameters in emergent communication research are often selected arbitrarily or based on convenience. Instead, hyperparameters ought to be selected, we suggest, such that they maximize emergent language's similarity to human language. For example, vocabulary sizes in emergent languages are often very small (only one of eight emergent language environments surveyed in Chapter 3 exceeds a vocabulary size of 70) while our research suggests that the optimal vocabulary size is in the 1k to 10k range. Increasing vocabulary sizes, then, not only improves transfer learning performance but also makes it possible for emergent languages to replicate the long-tailed, Zipfian word distribution that is characteristic of human language (Zipf, 1950; Piantadosi, 2014), for example.

Our experiments also confirm a significant relationship between transfer learning performance and corpus entropy. Not only does it appear that the entropy of a corpus determines a lower bound on XferBench score (lower is better) but that emergent languages minimize entropy with respect to a given XferBench score in a way that procedurally generated (i.e., non-emergent, synthetic) languages do not (see Figure 5.1). Such minimization is, significantly, an *emergent* phenomenon as neither entropy nor transfer learning performance are directly involved in the optimization of the emergent communication system (and neither entropy nor XferBench incorporate each other). This observation is significant in two regards: First, it suggests that transfer learning and, consequently, statistical similarity to human language can be (partially) explained with information theory. Second, it aligns closely with prior work that finds that emergent communication minimizes entropy with respect to task success within the environment (Kharitonov et al., 2020b; Chaabouni et al., 2022).

We discuss related work in Section 5.2. Methods are discussed in Section 5.3, and the experiments are presented in Section 5.4. An analysis of the results is performed in Section 5.5 with discussion and conclusion in Sections 5.6 and 5.7.

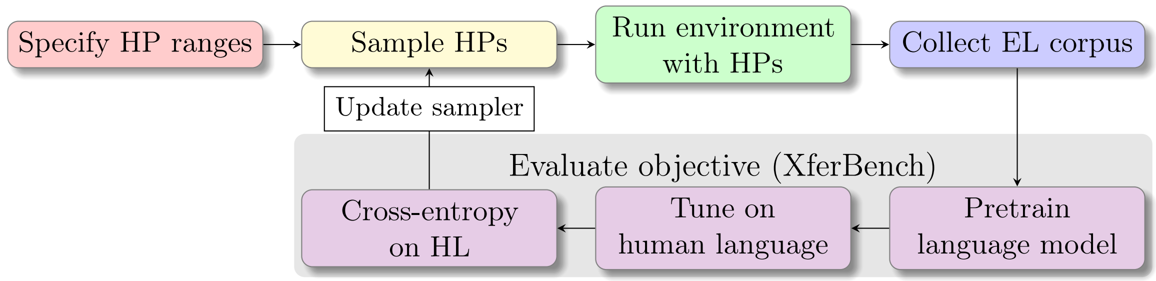


Figure 5.2: Illustration of hyperparameter optimization with XferBench.

Contributions We (1) introduce emergent communication environments which produce the most human language-like emergent languages to date, as shown by state-of-the-art performance on a deep transfer learning task using the XferBench benchmark; (2) provide concrete recommendations on better hyperparameter settings for emergent language, making them more statistically similar to human language; and (3) provide evidence that entropy minimization is a general property of emergent communication systems, showing that it is minimized with respect to transfer learning performance.

5.2 Related Work

For a general overview of deep learning-based emergent communication research, see Lazaridou et al. (2020). This paper shares the goal of producing emergent language corpora that are suitable for transfer learning to human languages with Yao et al. (2022), which also introduces the *corpus transfer* method for applying emergent communication techniques to pretraining deep learning models used in this paper. Boldt et al. (2022a), similarly to this paper, investigate the effect of hyperparameters on emergent communication, although their study focuses primarily on the effects of individual hyperparameters on entropy instead optimizing an entire system for an evaluation metric. Finally, this paper scales up emergent communication game hyperparameters in a way that overlaps with Chaabouni et al. (2022), although the latter focuses on addressing the practical challenges of scaling up certain facets of the signaling game (e.g., number of agents) rather than directly optimizing a particular objective.

The task of generating emergent languages for pretraining NLP models falls within the broad category data augmentation with synthetic data but differs from most other approaches due emergent language's unique nature as an *emergent* phenomenon. First, emergent language differs from procedurally generating data from rules because emergent techniques preclude stipulating the exact process for generating the data; expert knowledge is incorporated into designing the system which generates the data, not generating the data itself. On the other hand, emergent language differs from using pretrained language models to generate synthetic data since emergent communication is derived from scratch, again precluding any (pre)training on human language data.

5.3 Methods

5.3.1 Objective: XferBench

The ultimate objective that we are optimizing for is transfer learning performance on downstream human language tasks. This objective is quantified by XferBench (Chapter 4), which measures how much pretraining on an emergent language corpus decreases cross-entropy on a limited-data, downstream language modeling task on human languages (illustrated in the gray box of Figure 5.2). Since the output of XferBench is mean cross-entropy across human languages, a lower score better. XferBench takes as input a corpus of 15 million tokens, which is used for the pretraining stage and finetunes on 2 million tokens of the (human) evaluation language. The language model used for XferBench is based on GPT-2 (Radford et al., 2019) and has ∼60 million parameters. Since XferBench has a long runtime, we use a modified version only during hyperparameter search termed *XferBench-da* which only evaluates on one human language (viz. Danish) which we found to have high correlation ($R^2 > 0.95$) with the complete XferBench; see Section D.1 for details.

5.3.2 Environment: signaling game

The environment we use in our experiments is the signaling game. In particular we use the discrimination variant of the signaling game based on the implementation in EGG (Kharitonov et al., 2019, `https://github.com/facebookresearch/EGG`, MIT license). The discrimination variant of the signaling game consists of two agents, a sender and a receiver interacting for a single round. In a given round, the sender observes an input, sends a message to the receiver, and the receiver selects an observation out of a number of candidates based on the message. Of the candidate observations, one is correct (i.e., the same as the sender's input), and the rest are "distractors". In the implementation used in this paper:

- Observations are concatenations of a fixed number of one-hot vectors.
- Messages are sequences of integers represented by one-hot vectors.
- Agents are feed-forward neural networks with one hidden layer and GRU-based RNNs to generate/read the message.
- The sender–receiver system is trained end-to-end with backpropagation using a Gumbel-Softmax layer (Maddison et al., 2017; Jang et al., 2017) to generate the message.

Overall, this emergent communication system is about as "vanilla" as is studied in the literature. This is advantageous for a number of reasons:

- The environment is fast to run, requiring 10 to 120 minutes depending on the hyperparameters.
- It has a (comparatively) limited number of hyperparameters making hyperparameter search more tractable and reducing potential confounding variables.
- It serves as "lower bound" for optimizing emergent communication environments since we can determine the maximum performance possible in a system with minimal complexity.
- The training is stable, converging to a high success rate for most hyperparameter combinations.

The data is generated for the input corpus to XferBench by sampling from the dataset and feeding these observations into the sender which generates the message.

#	|Trials|	|Attrs.|	|Vals.|	|Distrs.|	Temp.	|Embed.|	|Hidden|	LR	|Vocab|	Length	|Epochs|
1	578	$[3,7]$	$[3,7]$	$[1,127]$	$[0.1,10]$	$[8,128]$	$[8,128]$	[500µ, 50m]	[10, 20k]	$[1,40]$	500
2	171	$[5,10]$	$[5,10]$	—	$[0.5,4]$	$[64,512]$	$[64,512]$	[500µ, 5m]	[300, 30k]	—	—
3	140	—	—	—	—	—	—	—	—	—	[500, 5k]
4	282	$[6,20]$	6	23	2	128	256	[1m, 3m]	[500, 30k]	—	—
4*	1	11	6	23	2	128	256	1.79m	9721	16	1715

Table 5.1: All hyperparameters were treated as log-scale hyperparameters. $|\cdot|$ refers to cardinality. "—" means unchanged from the previous run. µ, m, and k refer to the SI prefixes micro ($\times 10^{-6}$), milli ($\times 10^{-3}$), and kilo ($\times 10^{3}$), respectively.

5.3.3 Variables: hyperparameters

The hyperparameters are the independent variable of the primary experiments presented in this paper; that is, the hyperparameters will be varied in order to optimize the system for the objective function. Some hyperparameters manipulated in this study are unique to the signaling game (e.g., how many attributes and values in the signaling game observations) while others come from deep learning-based architectures more generally (e.g., learning rate, neural network architecture).

We primarily investigate the following hyperparameters:

Learning rate Multiplication factor for the weight updates for parameters in the neural network.

Embedding size Size of embedding layer in both the sender and the receiver networks; these are independent layers, but their sizes are varied in unison for hyperparameter search.

Hidden size The size of hidden layer in both the sender and the receiver networks; values are varied in unison.

***n* attributes** Number of one-hot vectors in each observation.

***n* values** Size of one-hot vectors in observations.

***n* distractors** Number of incorrect observations shown to the receiver (in addition to the correct one).

***n* epochs** Number of training examples seen.[2]

Temperature Temperature of the Gumbel-Softmax layer which the sender uses to generate messages during training.

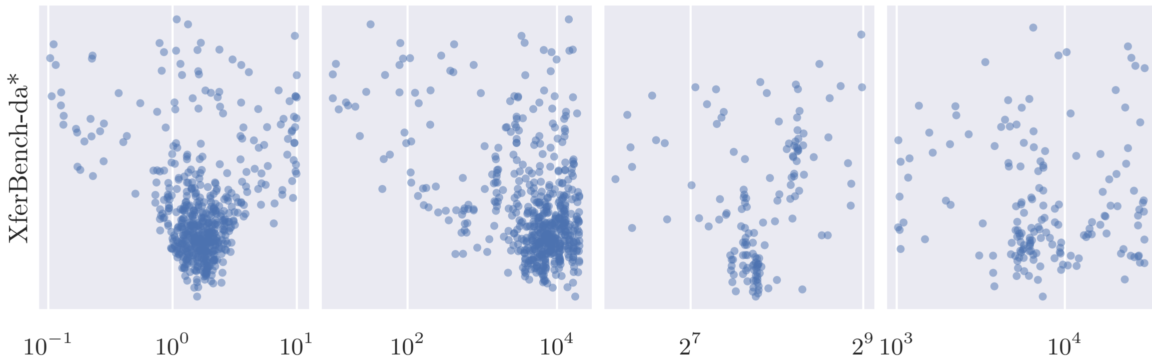
Vocabulary size Dimension of the one hot vectors which comprise the message.

Message length Number of one-hot vectors in a message.[3]

Other hyperparameters that were either not discussed or not investigated are documented in Section D.2.

[2]Since the data is procedurally generated, a new dataset of 1024 observations is sampled for each epoch.

[3]Technically, the implementation allows for variable length messages, but optimization led to all messages always being the max length.



(a) S1, Temperature (b) S1, Vocab Size (c) S3, Hidden Size (d) S2, Vocab Size

Figure 5.3: Examples of different hyperparameter–objective relations observed in the various searches and hyperparameters. From left-to-right, we have: (a) a clear best value, (b) a clear trend outside the provided range, (c) a weak trend toward a particular value, and (d) no definite trend. The y-axis based on different "sizes" of XferBench-da normalized to similar scales.

5.3.4 Optimization: hyperparameter search

Finally, we discuss the method used for optimizing the hyperparameters of the emergent communication system (the parameters system itself are optimized with backpropagation, as mentioned above). The simplest of all hyperparameter search methods is grid search, where each element of the Cartesian product of every set of hyperparameter values is evaluated. Even using a modest 3 values per aforementioned hyperparameter would require $3^{10} \approx 60\,000$ trials, taking 5 GPU-years (at 1 hour per trial). Thus, we employ Bayesian parameter optimization to more efficiently select hyperparameter combinations to evaluate; this additionally allows us to specify a range of hyperparameter values instead of individual values. This process is illustrated in Figure 5.2.

We specifically use a Tree-structured Parzen Estimator (TPE) (Bergstra et al., 2011) as implemented in Optuna (Akiba et al., 2019, MIT license). At a basic level, TPE works by partitioning hyperparameter combinations into a "good" set and a "bad" set based on the objective function value and selects the next combination of hyperparameters by maximizing the probability of the hyperparameters being in the good set divided by the probability of them being in the bad set. These probability estimates use multivariate kernel density estimators and permit discrete, categorical, and conditional hyperparameter values. After running the environment with the hyperparameters and the objective function on the result, the sampler's probability estimates are updated in accordance with the objective function's value. For a more detailed explanation, see Watanabe (2023).

5.4 Experiments

The code to run the experiments and analyses is publicly available at [supplementary material for review] under the MIT license.

5.4.1 Hyperparameter searches

In this paper, we present four main searches (Searches 1–4, parameters given in Table 5.1) with two additional searches (Searches 5r and 6e) for use in later analyses (Section 5.5). The following is a summary of the hyperparameter searches:

Search 1 Large number of hyperparameters varied with a wide range; used small version of XferBench-da (1M train tokens for 1 epoch, 200k test tokens for 2 epochs).

Search 2 Same number of hyperparameters varied with smaller or larger ranges depending on results of Search 1; used medium version of XferBench-da (4M train tokens for 2 epochs, 1M test tokens for 3 epochs)

Search 3 Same parameters as Search 2 while allowing number of epochs to go higher and using the full version of XferBench-da (15M train tokens for 5 epochs, 2M test tokens for 10 epochs).

Search 4 Reduces ranges or fixes parameters from Search 3 to maximize exploitation of good parameters; 4* in Table 5.1 is the best-performing trial from Search 4.

Search 5r Most parameters varied with wide ranges except using *random sampling* to remove sampling bias; similar to Search 1 with narrower ranges on learning rate. Discussed in Section 5.5.2.

Search 6e Optimized for maximizing entropy after a number of previous searches (not discussed in the paper); similar to Search 4 in this regard. Discussed in Section 5.5.2.

The parameters of Searches 1–4 are given in Table 5.1 (for complete table, see Table D.2). The implementation defaults for other hyperparameters were used unless otherwise specified. Optuna's default parameters for TPE were used across all experiments.

The signaling game takes 5 to 40 minutes to run (depending primarily on the number of epochs, and, to a lesser extent, the message length), and the full version of XferBench-da takes approximately 40 minutes to run. Thus, the average trial (for the latter searches) takes approximately $[0.75, 1.5]$ hours. Parallelization was used to run multiple trials within a search at a time. See Section D.4 for a discussion of computing resources used.

Search design For each iteration of the primary searches (i.e., 1–4), we changed the search parameters based on their correlation with the objective function. We observed four main univariate patterns[4], illustrated in Figure 5.3. For parameters with a clear trend toward the center (Figure 5.3a), we narrowed the range to encourage exploiting good values. Some parameters trended to one side of the range (Figure 5.3b), which indicated needing to extend the range. Parameters with weak to no trend (Figures 5.3c and 5.3d) were left unchanged for the initial searches and given an arbitrary value for the final search to reduce additional noise. Full hyperparameter plots given in Section D.7.

Searches 1 and 2 used a reduced version of XferBench to execute more trials quickly and prune the less promising hyperparameter ranges; nevertheless, caution was exercised in pruning since scaling up XferBench could change optimal hyperparameter values. The irregular number of trials per search were due to executing as many trials as possible within a certain time (rather than aiming for a particular number of trials).

[4]While we did look for multivariate effects (i.e., hyperparameters that are *not* independent), we did not observe any notable trends.

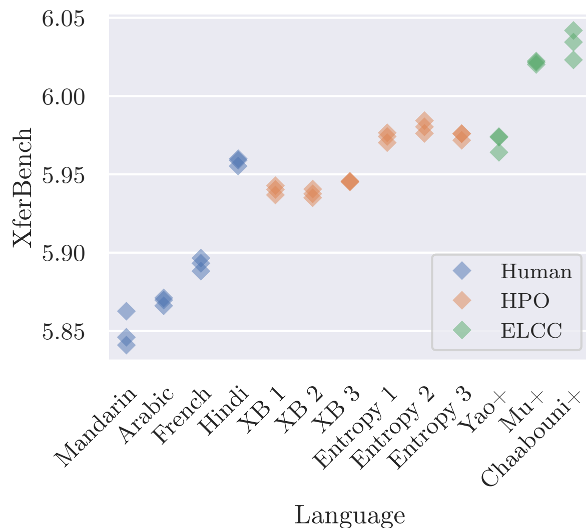


Figure 5.4: Bar chart of XferBench scores on emergent and human languages. XB 1–3 are emergent language corpora derived from Search 4 and Entropy 1–3 from Search 6e.

5.4.2 Languages evaluated

We select three categories of languages to evaluate with XferBench: human languages, those generated with the hyperparameter search discussed above, and extant emergent language corpora from ELCC (Chapter 3). The primary goal is for the search-derived languages to outperform all existing emergent languages and get as close to human language performance as possible. For the human languages, we use a subset of the baselines provided in Chapter 4. In particular, we use Mandarin and Hindi because they were the best- and worst-performing human languages, respectively, and French and Arabic to round out the language families represented.

For the search-derived languages, we selected the three best languages from the final primary run of hyperparameter search (Search 4) and evaluate them on the full set of evaluation languages in XferBench. We additionally include the three highest entropy languages from the entropy-maximizing search (Search 6e, discussed further in Section 5.5.2).

Finally, for the emergent language-based points of comparison, we select three of the best performing languages from ELCC. Most notably, this includes Yao+ (`corpus-transfer-yao-et-al/coco_2014` (Yao et al., 2022)) which performed far better than all other emergent languages on XferBench. Mu+ (`generalizations-mu-goodman/cub-reference` (J. Mu et al., 2021)) and Chaabouni+ (`ec-at-scale/imagenet-10x10` (Chaabouni et al., 2022)) were also included as more typical high-performing emergent languages on XferBench.

5.4.3 Results

Figure 5.4 shows 3 randomly seeded runs of the full XferBench score for each corpus. For the emergent languages from hyperparameter search, the models restored from checkpoints saved during the search, but the corpora were generated independently of the search. First, we see that the emergent languages from the XferBench-based search (XB 1–3) outperform all other emergent languages and even the Hindi corpus. While it is indeed significant that these emergent languages outperform a human language corpus, this corpus is also an outlier, and the emergent languages are still relatively far from matching the performance of the rest of the human language corpora. Nevertheless, these figures show that the XB 1–3 languages achieve state-of-the-art levels of similarity to human language. The corpora from the entropy-based search (Entropy 1–3) perform well, comparably to Yao+, but significantly worse than the XferBench-search languages.

5.5 Analysis

5.5.1 Importance of hyperparameters

Vocabulary size The most notable hyperparameter trend we found was with vocabulary size, where the best-performing languages had unique token counts of on the order of 1000 and vocabulary sizes closer to 10 000 (see Figure D.5); that is, the model could use up to 10 000 unique words but only uses 1000 after training. For reference, it is common practice in emergent communication research to use vocabulary sizes well under 100 (e.g., only 1 out of the 8 systems in ELCC produce corpora with >70 unique tokens).

Scaling up Similarly to vocabulary size, we observe indications to scale up message length, neural network layer size, and task information (i.e., number of attributes, values, and distractors): the most human like emergent languages require longer training, larger networks, and higher-information tasks than are often used in the emergent communication literature. Along with vocabulary size, these hyperparameter are most often trivial to adjust, meaning there is little reason not to adjust standard practice in emergent communication research to using hyperparameters in these ranges.

Learning rate Finally, in terms of raw importance with respect to XferBench score, learning rate was most significant; this result is not surprising as learning rate is significant in any deep learning algorithm. Nevertheless, part of the difficulty with learning rate is that there is no one best learning rate, and so performing at least some hyperparameter tuning with learning rate will be necessary for optimal performance.

Summary of recommendations We recommend the following hyperparameters as a rule of thumb: vocabulary size: 10 000, hidden layer size: 256, embedding layer size: 128, message length: 20, observation diversity: the higher the better (e.g., $6^{12} \approx 2$ trillion unique observations), epochs: train until task success plateau (not just until arbitrary threshold), learning rate: tune on final setting.

5.5.2 Entropy and XferBench

The most striking correlation we observe in our experiments is between XferBench score and unigram token entropy, which is illustrated in Figure 5.1 (Pearson's $r = -0.57$ for Search 5r only). The emergent languages pictured are all those generated by Searches 4 and 5r, while the human languages are taken from Chapter 4. We see that low entropy languages tend to score poorly on XferBench while high scoring languages have higher entropy; this aligns with the observed correlation between XferBench and entropy in Chapter 3. Furthermore, this correlation follows the same trend we see in human languages with respect to entropy.

Entropy's lower bound In particular, we have illustrated a lower bound of low entropy–low XferBench score that describes both emergent and human languages (the gray dashed line in Figure 5.1). This suggests that given a certain entropy, there is a hard limit on the performance XferBench that can be achieved. While further theoretical and empirical analysis would be required to verify that this a true lower bound, this aligns with the notion of language models as entropy-minimizers: Language models, in order to reduce the entropy on a target language, require a certain degree of entropy (i.e., information) in the pretraining data. Hence, low-entropy, low-information pretraining data leads to low entropy reduction (higher cross-entropy) language models.

Entropy minimization Looking again at Figure 5.1, we also see that the high-entropy, high-XferBench quadrant (upper right) is also sparsely inhabited. In fact, emergent and human languages seem to lie primarily near the Pareto frontier of low-entropy, low-XferBench score mentioned above. This comes in contrast to the XferBench scores of a variety of synthetic languages (descriptions of which are given in Section D.5) which often do not demonstrate this Pareto efficiency, even for synthetic languages performing well on XferBench.

This result is concordant with the related claim that entropy is "minimized" inside of emergent communication systems (Kharitonov et al., 2020b; Chaabouni et al., 2021). Such work has shown that emergent communication systems tend to find Pareto efficient solutions in terms of maximizing task success and minimizing entropy (this correlation in the hyperparameter search is discussed briefly in Section D.6).

Optimizing on entropy directly The correlation between entropy and XferBench naturally leads to a potential performance improvement: Why not use entropy as the hyperparameter objective instead of XferBench? Entropy takes seconds to compute instead of close to an hour. This is the experiment performed in Search 6e which was successful in producing languages with good XferBench scores but which still performed significantly worse than optimizing on XferBench directly (see Figure 5.4).

Given that the lower bound of entropy versus XferBench score is tighter than the upper bound, it is roughly the case that low entropy implies poor XferBench performance, but high entropy does not necessarily imply good XferBench performance. Thus, the fact that the entropy-based search finds good but not optimal emergent languages fits with the earlier observation about bounds of entropy and XferBench score. With these observations in mind, a refinement to the hyperparameter search algorithm would be to prune low-entropy trials before running XferBench while fully evaluating the trial on XferBench if has a high entropy.

Task success The correlation between task success and XferBench score (Figure 5.5, Pearson's $r = -0.40$) is not as dramatic as with entropy. Nevertheless, the negative correlation (better task success, better XferBench score) matches the expectation that the realism of emergent language is positively correlated with the efficacy of the language. This

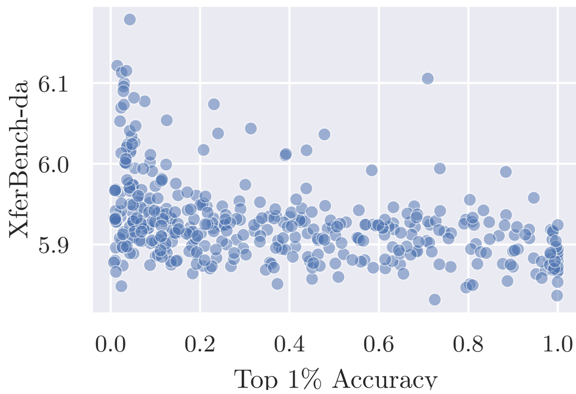


Figure 5.5: Accuracy versus XferBench for Search 5r. Accuracy is measured as proportion of rounds for which the correct observation is ranked in the top-1 percentile among all distractors.

relationship is a foundational assumption of emergent communication techniques generally: the realism of simulation-derived language comes, in part, from its development out of the functional pressures to communicate.

5.6 Discussion

Similarity to human language The primary motivation for optimizing emergent communication systems on XferBench is to create more human language-like emergent languages. In this way, this environment and the recommended hyperparameters provide a better baseline environment for future emergent communication research to work from. This similarity to human language is critical for nearly every application of emergent communication research, not only related to machine learning and NLP but also areas with more linguistic focus (Chapter 2). Although XferBench quantifies a decidedly more deep learning, data-driven notion of similarity, this account is complimentary with more explicitly linguistic notions of similarity to human language.

For example, linguistic phenomena such as parts of speech fundamentally concern whole classes of words behaving predictably in a variety of environments. Thus, trivially small languages are not suitable for addressing such phenomena as there are not classes of words and no variety to generalize over. Even something as fundamental as the Zipfian distribution of words in human language presupposes a large vocabulary size (Zipf, 1950; Piantadosi, 2014). Furthermore, smaller-scale emergent languages are a greater risk for overfitting since the capacity of a neural network quickly enters the overparameterization regime when the language has as small vocabulary, message length, etc. (A. Gupta et al., 2020).

Emergent properties The relationship between entropy, task success, and XferBench score demonstrated in the hyperparameter searches emphasizes the presence of *truly emergent* properties and processes in emergent communication: Neither entropy nor transfer learning performance are directly optimized for (cf. task success). Just as Pareto efficient entropy has been found for task success in emergent languages (Kharitonov et al., 2020b), we find some degree of Pareto efficiency with entropy and XferBench performance (and to a limited degree with task success and XferBench). What this shows is that the communicative pressures and information theoretic considerations are a key ingredient in emergent language's similarity to human language. Thus, task success and entropy serve as additional ways to reason about emergent language and how to apply it to human language. Nevertheless, the limited correlation we find among these properties also tells us that emergent language is not trivially explained by these factors either.

Future work On the front of creating more human language-like emergent languages, a next step is to introduce new variations of the signaling game, entirely new environments, or more sophisticated neural architectures and optimize them on a metric like XferBench in order to progress towards the long-term goal of producing realistic emergent languages for transfer learning. Because this paper has wrung as much performance as is possible from the basic signaling game environment, there can be greater certainty that innovations producing higher-performing languages are actually causing the improvement. Otherwise, more trivial factors like better learning rate tuning could become confounding variables.

As far as investigating the entropy minimization pressure in emergent languages, further theoretical work needs to build models and generate testable hypotheses; theoretical models are the key to scientific explanation beyond merely showing the existence of correlations. Nevertheless, this paper has shown that hyperparameter turning can be an effective tool for producing a large variety of emergent language that preclude hyperparameters being confounding variables. Such methods of generating datasets will be invaluable in empirically testing theoretical models of emergent language.

5.7 Conclusion

In this paper we have used hyperparameter search to generate the most human language-like emergent language to date, as quantified by XferBench. Not only does this represent a step forward for using emergent languages as realistic synthetic data for transfer learning but also provides insight into how hyperparameters can be better addressed in future emergent communication research. Finally, the hyperparameter search reveals further importance of the role of entropy in emergent language. High entropy appears to be a necessary condition for good transfer learning performance while at the same time, emergent language appears to minimize entropy for a given level of transfer learning performance. Furthermore, this entropy minimization is not replicated in synthetic languages suggesting that emergent language is more than just "synthetic languages with extra steps".

Limitations

In terms of finding the most human language-like emergent language, this study is limited in terms of the simplicity of the environment. A single round signaling game with a fixed sender and receiver and uniform, synthetic observations is a no-frills environment which, while good for stability and simplicity, is limited in the richness of information to be communicated, and as a result, the languages it can produce.

Regrading the investigation of the link between entropy and XferBench score and task success, we were not able to build any theoretical models to scientifically test particular hypotheses about the relationships between the variables; instead, we are only able to offer empirical evidence that there are trends warranting further investigation. Finally, the recommendations we can given regarding the hyperparameters of emergent communication systems are limited because hyperparameter search is relatively "messy"; it is geared toward maximizing performance more than uncovering generalizable trends. Additionally, we perform our experiments with a signaling game which provides only limited evidence for the behavior of emergent communication systems with different tasks.

Chapter 6

Discovering Morphemes in Rich Corpora

Abstract

We introduce CSAR, an algorithm for inducing morphemes from emergent language corpora of parallel utterances and meanings. It is a greedy algorithm that (1) weights morphemes based on mutual information between forms and meanings, (2) selects the highest-weighted pair, (3) removes it from the corpus, and (4) repeats the process to induce further morphemes (i.e., Count, Select, Ablate, Repeat). The effectiveness of CSAR is first validated on procedurally generated datasets and compared against baselines for related tasks. Second, we validate CSAR's performance on human language data to show that the algorithm makes reasonable predictions in adjacent domains. Finally, we analyze a handful of emergent languages, quantifying linguistic characteristics like degree of synonymy and polysemy.[1]

6.1 Introduction

Emergent languages—communication systems invented by neural networks via reinforcement learning—are fascinating entities. They give us a chance to experiment with the processes underlying the development of human language to which we would not otherwise have access. A perennial problem in this field, though, is that emergent languages are difficult to interpret. The strategies emergent languages use to convey meaning do not always align with those known from human language (Kottur et al., 2017; Chaabouni et al., 2019a; Kharitonov et al., 2020a). Yet a lack of general-purpose methods for investigating the structure of emergent communication prevents us from systematically investigating how they encode meaning.

As an essential step towards understanding emergent languages, we introduce CSAR, an algorithm for *morpheme induction* on emergent language. That is, given an input corpus of parallel data: utterances and their associated meanings, find the smallest meaningful components of utterances with their accompanying meaning. Simply put, this task is to jointly segment utterances and align them with their meanings. The output of this algorithm, then, is a mapping between the forms and meanings of the emergent language (example shown in Fig. 6.1). Furthermore, the proposed algorithm is easily applicable to almost any

[1]Based on "Morpheme Induction for Emergent Language" appearing in the *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing* (Boldt et al., 2025a).

Form	Meaning
3, 6	{not, gray}
7, 7	{not, blue}
32	{circle}
4, 5	{not, yellow}
6, 12, 6, 12	{green, or, yellow}
3, 12, 3	{blue, or, yellow}

Figure 6.1: Example of morphemes extracted from a signaling game with pixel observations.

emergent language due to the simplicity of the input format. In fact, the algorithm is general purpose enough to produce reasonable results in other domains, as we demonstrate with human language-based image captioning, machine translation, and word segmentation data. Validating CSAR's performance on human language data as well as synthetic data is critical to demonstrating its effectiveness since there is no way of obtaining ground truth morphemes for emergent languages.

An inventory of the morphemes of an emergent language is the foundation of many further linguistic analyses. Existing studies of compositionality (Korbak et al., 2020), word boundaries (Ueda et al., 2023), and grammar induction (Wal et al., 2020) could be validated and augmented with information on the morphology of emergent languages, and new directions would also be made possible, including analyses of the morphosyntactic patterns and typological properties of emergent languages. Ultimately, such studies form one of the pillars of emergent communication research: learning what emergent language can tell us about human language (Chapter 2).

In Section 6.2 we define the task of morpheme induction and discuss related work. Section 6.3 presents the proposed algorithm, CSAR. Section 6.4 validates CSAR's performance on procedurally generated and human language data. Section 6.5 applies CSAR to emergent languages. And we discuss the paper's findings and limitations and conclude in Sections 6.6 to 6.8.

Contributions This work: (1) Introduces an algorithm for inducing morphemes, applicable to a wide variety of emergent language corpora. (2) Offers an easy-to-use Python implementation for executing the proposed algorithm on arbitrary emergent language data. (3) Provides a first look into the morphology of emergent languages including phenomena such as polysemy and synonymy.

6.2 Problem Definition

In this section we give a precise definition of the problem and the terms we will use throughout the paper.

6.2.1 Task: morpheme induction

We define the task of *morpheme induction* as follows: Given a corpus of *utterances* and their corresponding *complete meanings*, identify *minimal, well-founded form–meaning pairs* (i.e., *morphemes*) present in the corpus. This collection of pairs is the *morpheme inventory* of the corpus.

form a sequence of (form) tokens; represented as an integer sequence in emergent language.

utterance a complete sequence of tokens produced by an agent; forms are subsequences of utterances.

meaning a set of meaning tokens (i.e., atomic meanings grounded in the environment).

complete meaning a meaning which represents the entire meaning of an utterance.[2]

well-founded a form–meaning pair is well-founded when the particular form corresponds with a particular meaning.

minimal a well-founded form–meaning pair is minimal when there is no way to decompose the pair while maintaining continuity of meanings.

It is important to note that we make two assumptions about the complete meanings. First, complete meanings are assumed to be *abstracted* already, hence the reason we can represent them as a set of atomic meanings. That is to say that the "raw semantics" of the utterances are already broken down into individual components of interest; this task does not entail automatically finding meaning in arbitrary data (cf. clustering). Second, since complete meanings are sets, they are not able to represent more complex phenomena that might require graph structures, for example (cf. abstract meaning representations).

Additionally, we note that *well-founded correspondence* is a concept subject to a variety of philosophical accounts. Sometimes these accounts hold that the meaning is derived from either the behavior or state of mind of a language user. Yet in this task, we only have access to a corpus, not to the language users themselves; thus, we employ a notion of "well-founded correspondence" most akin to a statistical view semantics (e.g., as in the distributional hypothesis).

6.2.2 Related work

Emergent language Lipinski et al. (2024) serves as the inspiration for this paper through its application of normalized pointwise mutual information to probe emergent languages for certain kinds of form–meaning relationships, though it stops short of providing full morpheme inventories over arbitrary data. Ueda et al. (2023) introduces a method of form-only segment induction for emergent language based on token-level entropy patterns in utterances.

Finally, Brighton (2003) introduce methods for inducing morphemes from simulations of language evolution. In particular, the algorithm is based on finite state transducers and the minimum description length principle. The key difference, though, is that the FST-based method assumes a strict form–meaning correspondence that does not appear to hold in emergent languages generated by deep neural networks. Thus, CSAR is designed to handle noise and looser form–meaning correspondence.

[2] atomic meaning $\in$ meaning $\subseteq$ complete meaning

Statistical word alignment The task of morpheme induction resembles the task of statistical word alignment for machine translation insofar as it involves learning a mapping between two modalities. Well-known algorithms for this task include the IBM alignment models (Brown et al., 1993). While morphemes can be extracted from the alignments, the alignments themselves are not intended to represent morphemes as such.

Segment induction Segment induction is similar to morpheme induction, except that it deals only with the forms; these methods, then, cannot provide a mapping between form meaning because they are meaning-unaware. Sometimes this task is called "morpheme induction" since the segments are supposed to correspond to morphemes, but they are not morphemes in the particular sense we use for this paper, that is: explicit form–meaning pairs. An example of an algorithm which addresses this task is Morfessor (Creutz et al., 2002; Virpioja et al., 2013) or the submissions to the SIGMORPHON 2022 Shared Task Batsuren et al. (2022). Narasimhan et al. (2015) introduce a semi-supervised segment induction algorithm that uses semantic features to guide segmentation (viz. groups of morphologically related words), although meanings are not modelled explicitly and "word" is not a well-defined concept for emergent languages. The discovery of valid segments by tokenization methods based on statistics—such as BPE (Sennrich et al., 2016; Gage, 1994) and Unigram LM (Kudo, 2018)—is largely an epiphenomenon, not a design goal.

6.3 Algorithm

In this section we introduce the algorithm for morpheme induction: CSAR (Count, Select, Ablate, Repeat). CSAR comprises the following steps:

1. Collect form and meaning candidates from the corpus.
2. While form and meaning candidates remain.
   a) Count co-occurrences of form and meaning candidates.
   b) Select form–meaning pair with the highest weight.
   c) Remove instances of the form–meaning pair from the corpus.
3. Selected form–meaning pairs constitute the morpheme inventory of the corpus.

The code implementing CSAR as well as the experiments discussed later is available under a free and open source license at `https://github.com/brendon-boldt/csar`.

6.3.1 Representation and preprocessing

Input data The input data to CSAR is a parallel *corpus* of utterances and their meanings. Each record in the corpus is a tuple of form and meaning where a form is a list of (form) tokens and a meaning is a set of (meaning) tokens.

Candidate collection Given the corpus, we can identify and count the *form* and *meaning candidates* to produce their corresponding *occurrence matrices*. A form candidate is any substring of form tokens under consideration for inducing morphemes. A meaning candidate is any subset of meaning tokens under consideration for inducing morphemes. The most straightforward approach is to simply consider every non-empty substring of forms and subset meanings, although CSAR is not constrained to this approach in theory (cf. Section E.1.1).

Having defined the universe of forms and meanings, we can build a binary *occurrence matrix* for forms and one for meanings, where each row corresponds to a record and each entry corresponds to the presence (1) or absence (0) of a form/meaning in that record. Thus, the form occurrence matrix has the shape $O_{\mathcal{F}} : |\mathcal{R}| \times |\mathcal{F}|$ and the meaning matrix $O_{\mathcal{M}} : |\mathcal{R}| \times |\mathcal{M}|$, where $\mathcal{R}$ is the list of records, $\mathcal{F}$ is the set of all forms candidates, and $\mathcal{M}$ is the set of all meanings candidates.

Example If we had a simple corpus with records ("s", $\square$), ("st", $\boxtimes$), ("ct", $\otimes$), the corresponding occurrence matrices would be:

$$\mathcal{O}_{\mathcal{F}} = \begin{bmatrix} \cdot & \text{s} & \cdot & \cdot & \cdot \\ \cdot & \text{s} & \text{t} & \cdot & \text{st} \\ \text{c} & \cdot & \text{t} & \text{ct} & \cdot \end{bmatrix} \quad \mathcal{O}_{\mathcal{M}} = \begin{bmatrix} \square & \cdot & \cdot & \cdot & \cdot \\ \square & \times & \cdot & \boxtimes & \cdot \\ \cdot & \times & \bigcirc & \cdot & \otimes \end{bmatrix}, \tag{6.1}$$

where entries with value 1's are shown with the occurring symbols, and entries with value 0's with $\cdot$ for clarity.

6.3.2 Main loop

Weighting and co-occurrences Given the occurrence matrices, the next step is to compute the weights of all potential pairs. The pair with the highest weight will be selected and added to the morpheme inventory. The weight of a form–meaning pair is the mutual information of the binary variables representing the corresponding form and meaning. The mutual information of a particular form–meaning pair is given by

$$I(F; M) = \sum_{x \in F} \sum_{y \in M} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}, \tag{6.2}$$

where $F = \{f, \neg f\}$, $p(f)$ is the probability of f appearing in a record, $p(\neg f)$ is the probability of f not appearing, and the rest are defined analogously. The key term of the mutual information expression is the joint probability between a form and a meaning, $p(f, m)$: since f and m are binary variables, all other joint probabilities can be computed from their joint probability and the marginal probabilities. The joint probability can be computed by normalizing the sum of co-occurrences of given forms and meanings, namely:

$$p(f, m) = \frac{1}{|\mathcal{R}|} \sum_{j=1}^{|\mathcal{R}|} O_{\mathcal{F}}[j, i_f] \wedge O_{\mathcal{M}}[j, i_m] \tag{6.3}$$

where i_f and i_m are the indices of f and m in their respective matrices. More succinctly, co-occurrences can be computed with matrix multiplications, yielding

$$p(f, m) = \frac{1}{|\mathcal{R}|} \cdot \left(O_{\mathcal{F}}^{\top} O_{\mathcal{M}} \right) [i_f, i_m] \tag{6.4}$$

Other weighting methods were explored including joint probabilities, pointwise mutual information, and normalized pointwise mutual information, though mutual information was found to perform best empirically.

The above weighting function results in ties which we break with the following heuristics: (1) higher initial weight, (2) fewer selected pairs with this form, (3) larger form size, and (4) smaller meaning size.

Remove pair from corpus The final step of the algorithm's main loop is ablating the pair from the corpus. That is, once we select a form–meaning pair, we want to remove all co-occurrences of the form and meaning in order to determine what form–meaning correspondences remain to be explained. For example, after ablating the pair ("t", $\times$), the corpus from above would comprise ("s", $\square$), ("s", $\square$), and ("c", $\bigcirc$); the occurrence matrices would then be updated to reflect this. In cases where ablating a pair is ambiguous, we apply a heuristic (see Section E.1.2).

Repeating and stopping After ablating the selected form–meaning pair, the algorithm repeats the main loop, beginning again at the weight-computation step (with the updated occurrence matrices). The one difference is that—in subsequent weight computations—the weight of a pair cannot go up, preventing spurious correlations from arising in later steps.

This loop continues until form or meaning occurrences are exhausted or some other criterion is met (e.g., time limit, inventory size limit). In this way, CSAR is an "anytime" algorithm since it can be stopped after an arbitrary number of iterations and still produce a sensible result. This is because the most heavily weighted morphemes can be considered the highest *confidence* morphemes, meaning that stopping the algorithm before completion will only leave out the lowest confidence morphemes.

6.3.3 Implementation

The implementation of CSAR introduced in this paper is written in Python making use of sparse matrices from `scipy` (Virtanen et al., 2020, BSD 3-Clause license) and JIT compilation with `numba` (Lam et al., 2015, BSD 2-Clause license) to speed up execution. CSAR is conceptually simple. Most of the implementation complexity lies in efficiently handling the occurrence matrices, especially when removing a form–meaning pair from the corpus. For example, the co-occurrence matrix has the shape $|\mathcal{F}| \times |\mathcal{M}|$ which is massive considering that $\mathcal{F}$ and $\mathcal{M}$ are already accounting for the universes of all possible forms and meanings in the corpus. Nevertheless, there are a wide range of heuristics that can be applied to greatly speed up execution while maintaining performance (see Section E.1.3).

6.4 Empirical Validation

To validate the ability of CSAR to find well-founded morpheme inventories, we test it against procedurally generated datasets as well as human languages. Since we do not have access to ground truth morphemes for emergent languages, we gauge the effectiveness of CSAR's morpheme induction in the next best way: by testing its performance in these adjacent domains. Procedurally generated datasets (described in Section 6.4.1) both give us access to the "ground truth" morphemes and allow us to vary particular facets of the languages. Having ground truth morphemes allows us to quantitatively evaluate CSAR against baseline methods

(Section 6.4.2). Fine-grained control over the facets of the languages permits us to identify particular phenomena that are challenging for CSAR to induce correctly (Section 6.4.3). We also test CSAR against human language data (Section 6.4.4) in order to give a qualitative sense of the effectiveness of the algorithm.

6.4.1 Procedural datasets

The dataset-generating procedure has the following basic structure: (1) Meanings are sampled according to some structure (viz. a fixed attribute–value vector). (2) An utterance is produced from this meaning according to a mapping of meaning components to form components. (3) The form–meaning pairs that were used to generate the utterance are added to the set of ground truth morphemes. In the basic case, for example, each particular attribute and value is associated with a unique sequence of tokens which are concatenated to form an utterance, creating a one-to-one mapping from meanings to forms.

Variations Such languages are trivial to induce morphemes from, so we introduce the following variations to produce more complex datasets:

Synonymy Multiple forms may correspond to the same meaning.
Polysemy Multiple meanings may correspond to the same form.
Multi-token forms A form may comprise more than one token, possibly overlapping with other forms.
Vocab size Number of unique tokens.
Sparse meanings Meanings occur independently of each other with no additional structure (i.e., not structured as attribute–value pairs).
Distribution imbalance Meanings are sampled from non-uniform distributions.
Dataset size Number of records in the dataset.
Number of meanings Total number of meanings (e.g., varying number of attributes and values).
Noise forms Form tokens not corresponding to any meanings are added.
Shuffle form Inter-form order is varied randomly (while maintaining intra-form order).
Non-compositionality A given form may correspond to multiple meanings simultaneously.

For the following analyses, we report values for a collection of procedural datasets built from the Cartesian product of two values for each of the above variations (one where the variation is inactive and one where it is). See Section E.2.1 for details.

Evaluation metric We use F_1 score (harmonic mean of precision and recall) to assess the quality of an induced morpheme inventory given the ground truth inventory. We define precision as

$$\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\max_{g\in\mathcal{G}} s(i,g), \tag{6.5}$$

where $\mathcal{I}$ is the set of induced morphemes, $\mathcal{G}$ is the set of ground truth morphemes, and s is the similarity function. For exact F_1, the similarity function is 1 if the morphemes

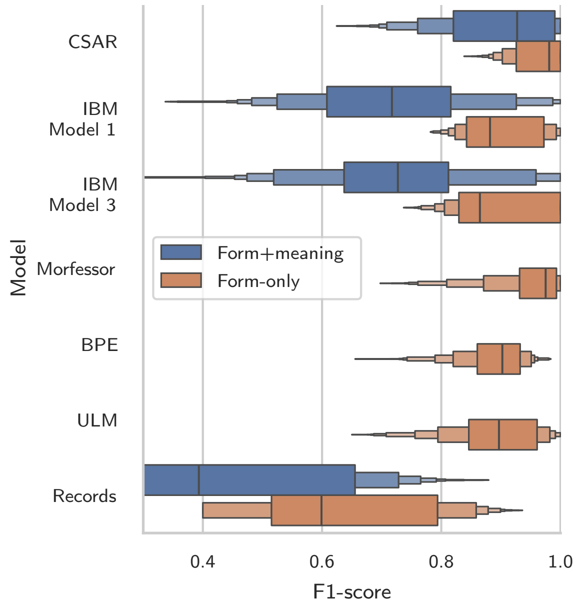


Figure 6.2: Fuzzy F_1 scores for CSAR and baseline methods across procedural datasets. Results reported for form–meaning inventories and form-only inventories.

are identical and 0 otherwise. In fuzzy F_1, the similarity function is the minimum of form similarity (normalized insertion–deletion ratio[3]) and meaning similarity (Jaccard index). Recall is defined similarly to precision except that the roles of $\mathcal{I}$ and $\mathcal{G}$ from Eq. (6.5) are reversed.

6.4.2 Comparison with baselines

Below we describe the baseline methods we use for comparison.

IBM Model 1 Simple expectation-maximization approach to machine translation primarily through aligning words in a sentence-parallel corpus. (Brown et al., 1993)

IBM Model 3 Built on top of the IBM Model 1 to handle phenomena such as allowing a form to align to no meaning.

Morfessor A form-only segmentation algorithm built to handle human language; also uses an EM algorithm.

Byte pair encoding A greedy form-only tokenization method which recursively merges frequently occurring pairs of tokens. Vocabulary size is selected according to a simple heuristic (see Section E.2.2). (Gage, 1994; Sennrich et al., 2016)

Unigram LM An EM-based form-only tokenization method which starts with a large vocabulary and iteratively removes tokens contributing least to the likelihood of the data. Vocabulary size is selected according to a simple heuristic (see Section E.2.2). (Kudo, 2018)

Record A trivial baseline where the inventory is just the set of all records.

[3] $1 - (\text{insertions} + \text{deletions})/(|s_1| + |s_2|)$

Dataset	Induced Morpheme
Morphology	(“ed$”, {PAST}) (“ ’ ”, {POSSESSIVE})
Image captions	(“stop sign”, {stop sign}) (“woman”, {person}) (“skier”, {person, skis})
Translation	(“Member States”, {Mitgliedstaaten})

Figure 6.3: Examples of morphemes induced from various human language datasets and tasks.

For the baseline methods which do not handle meanings and only produce forms, we report the form-only F_1 score (i.e., $s(i,g)$ only takes the form into account), though CSAR and IBM models still have access to meanings. For form-only metrics, we exclude datasets which include noise forms as form-only methods cannot identify which forms are noise.

Results The results of CSAR and the baselines on the procedural datasets are presented in Fig. 6.2, which shows the distributions of mean scores for each hyperparameter setting for the procedural datasets. Each setting was repeated over 3 random seeds. Additional results are given in Section E.2.3. For inducing full morphemes (form and meaning), CSAR performs the best by a large margin over the baselines (and even greater margin when considering exact F_1). The IBM alignment models perform better than the trivial record-based baseline but still perform noticeably worse than CSAR. While CSAR yields roughly equal precision and recall, the IBM models' precision is lower than their recall suggesting that they are more prone to inducing spurious morphemes than CSAR.

When evaluating the forms only, we find that CSAR is the best method with Morfessor exhibiting comparable performance. The IBM alignment models exhibit roughly the same performance as the tokenization methods (BPE and Unigram LM). As with the full morpheme results, CSAR is the only method to achieve comparable precision and recall with all other baselines having precisions lower than their recalls.

6.4.3 Error Analysis

For the most part, the errors CSAR makes are “edit errors”: identifying a correct morpheme but adding or removing a form or meaning token. This is reflected in the near-parity between precision and recall. This is in contrast to the baseline methods which are more prone to inducing too many morphemes, leading to lower precision.

Generally speaking, as more variations are added to a dataset, the performance degrades further. In particular, CSAR's performance decreases the most with small corpus sizes, overlapping multi-token forms, and non-compositional mappings. On the other hand, using sparse meanings, shuffling the forms, and using a non-uniform meaning distribution have relatively little effect.

6.4.4 Human language data

In this section we discuss the results of running CSAR on three different human language datasets. While these datasets are not the intended domain of CSAR—and CSAR is certainly not the best algorithm for the tasks—the point of these experiments is to demonstrate the general effectiveness of the algorithm qualitatively (examples shown in Fig. 6.3). Since these datasets are larger, we employed heuristic optimizations to CSAR to reduce their runtime (described in Section E.1.3). The top 100 induced morphemes for each human language dataset are given in Section E.5.1.

Morpho Challenge The first human language dataset we use is from the Morpho Challenge (Kurimo et al., 2010). This dataset is a human language approximation of the task of morpheme induction for emergent language. Concretely, the utterances are single English words, divided up at the character level, while the meanings are the constituent morphemes.

CSAR is able to recover a wide variety of morphemes including: roots like ("ˆfire", {fire}), prefixes like ("ˆre", {re-}), suffixes like ("ed$", {PAST}), and other affixes like (" ' ", {POSSESSIVE}). While the vast majority of morphemes CSAR induces are accurate, a handful of the lowest-weighted morphemes are spurious (e.g., ("s$", {boy})) likely due to inaccurate decoding earlier in the process (i.e., part of the true form for a given meaning was included in a prior meaning).

Image captions The next dataset we employ is the MS COCO dataset (T.-Y. Lin et al., 2015, CC BY 4.0). In particular, we take the image captions to be the utterances, treating words as atomic units, and the meaning to be the labeled objects in the image (e.g., person, cat).

The bulk of highest weighted induced morphemes are direct equivalents of the objects they describe (e.g., ("cat", {cat})). We find instances of synonymy (e.g., ("bicycle", {bicycle}) and ("bike", {bicycle})) as well as polysemy (e.g., ("animals", {cow}) and ("animals", {sheep})). Finally, we also observe compound forms like ("stop sign", {stop sign}) as well as compound meanings such as ("skier", {person, skis}). As we go beyond the top 100 or so, the associations between forms and meanings remain reasonable but become looser such as ("bride", {dining table, tie}) or ("sink", {toothbrush}).

Machine translation For machine translation, we use the WMT16 dataset and the English–German split, in particular (O. r. Bojar et al., 2016). In this case, the English text is considered to be the utterance and the German text to be the meaning, with words being the atomic units on both sides.

As with the image caption results, the bulk of induced morphemes are direct equivalents (e.g., ("and", {und})). Beyond these simple one-to-one mappings, CSAR induces the polysemic relationship ("the", {der}) and ("the", {die}). Finally, CSAR also picks up on multi-token forms like ("Member States", {Mitgliedstaaten}).

6.5 Analysis of Emergent Languages

Here we apply CSAR to a handful of emergent language environments and observe various metrics which can be derived from the induced morpheme inventories. Although we do not have ground-truth morphemes for these environments, we can still observe patterns in the induced morphemes which demonstrate the diagnostic power of CSAR.

6.5.1 Environments

The selected emergent language environments span three different observation spaces: one-hot vectors, synthetic images of shapes, and natural images of birds. Within each of the environments, further signaling game variations are described below.

Vector observations In the vector observation signaling game (based on the EGG framework (Kharitonov et al., 2019)) the agents directly observe one-hot vectors without having to further extract representations. Specifically, we use two variants of the observation space: (1) the standard attribute–value setting where each of 4 attributes can take on 4 distinct values and (2) the "sparse" setting where there are 8 binary attributes which are either present or absent. During the training of the emergent language agents, the sparse variant is identical to the attribute-value variant (except for the number of attributes and values), but when the corpora are passed to CSAR, the attributes with a value of 0 are filtered out, yielding variable length meanings. Hyperparameters for both environments are given in Section E.3.1.

For both observation spaces, we also test two variations of the signaling game. The first is the *discrimination game*, where the receiver must answer a multiple-choice question where the correct observation is accompanied by multiple (incorrect) distractors. The second is the *reconstruction game*, where the receiver must recreate the original observation similar to the decoding segment of an autoencoder.

Image observations The second environment is introduced by J. Mu et al. (2021) and uses both synthetic and natural images for its observation space. The synthetic images come from ShapeWorld, a tool for generating images of shapes with varying properties (Kuhnle et al., 2017) while the natural images are from the CUB-200-2011 dataset documenting various kinds of birds (Wah et al., 2011). The meanings for these environments are discrete attributes describing the images (e.g., "red" for ShapeWorld or "wing color: brown" for CUB) provided with the datasets. In addition to a more standard discrimination game (termed *reference game* in this paper), J. Mu et al. (2021) also introduce the *set-reference* and *concept* variants of the discrimination game. These variations increase the level of abstraction in the game in order to encourage more generalizable (and compositional) languages to emerge.

Specifically, the reference game functions like the discrimination game described above except that there are multiple target images and the sender sees both the target images and the distractors. The receiver, in this case, must classify the each image as being target or distractor (i.e., multi-label instead of multi-class). In the reference game, the target images are all identical while in the set-reference game, the target images are variations of the same object (e.g., a red triangle in different positions/rotations). Finally, in the concept game, the target images comprise different objects sharing one or more concepts (e.g., triangular).

Obs Type	Subtype	Game Type	Inventory Ent.	Form Length	Meaning Size	Synonymy	Polysemy	Bijectivity	Toposim
Vector	Attr–val	Discrim	6.2	2.2	1.2	1.5	0.6	0.45	0.46
Vector	Attr–val	Recon	4.2	1.2	1.0	0.2	0.1	0.92	0.84
Vector	Sparse	Discrim	5.9	1.8	1.3	2.0	0.9	0.34	0.44
Vector	Sparse	Recon	5.4	1.3	1.2	1.7	0.8	0.44	0.56
Image	Synth	Ref	4.5	1.6	1.1	1.2	0.7	0.07	0.00
Image	Synth	Set-ref	6.4	1.2	1.7	0.8	2.2	0.28	0.12
Image	Synth	Concept	6.2	1.3	1.4	1.2	2.0	0.25	0.12
Image	Natural	Ref	7.1	1.1	1.9	0.3	3.6	0.09	0.05
Image	Natural	Set-ref	6.9	1.1	1.9	0.3	3.2	0.25	0.30
Image	Natural	Concept	6.9	1.2	2.2	0.2	2.8	0.47	0.34

Table 6.1: Metrics (described in Section 6.5.2) across the morpheme inventories of various emergent languages (averaged across 10 runs).

6.5.2 Metrics

We present a handful of quantitative metrics to act as summary statistics for the morpheme inventories induced from the above emergent languages.

Inventory entropy Entropy (in bits) of the morphemes according to their prevalence (probability of occurring out of all morpheme occurrences identified during induction). We use the inventory entropy to give a rough sense of the breadth of the morpheme inventory as CSAR can induce a large number of low-prevalence, low-quality morphemes (whereas entropy is more robust to this). A rough translation to size is $H^2 \approx$ equivalent number of equally-probable morphemes.

Morpheme bijectivity Weighted mean normalized pointwise mutual information (NPMI) of morphemes in the inventory corresponding to how "one-to-one" the morphemes identified are; higher bijectivity corresponds to higher quality morphemes/inventory. See Section 7.4.4 for the introduction of this metric.

Topographic similarity Correlation (Spearman's ρ) between distances in the utterance space and complete meaning space (Brighton et al., 2006; Lazaridou et al., 2018) of the corpus (i.e., not based on the morpheme inventory).

Synonymy Entropy across forms for a given meaning; computed using prevalence of forms normalized within the particular meaning.

Polysemy Entropy across meanings for a given form; as with synonymy, *mutatis mutandis*.

Form size Mean number of tokens in a form, weighted by morpheme prevalence.

Meaning size Mean number of tokens in a meaning, weighted by morpheme prevalence.

With the exception of inventory size and toposim, the above metrics are weighted by *prevalence* which is the proportion of records from which the morpheme was ablated.

6.5.3 Results

The morpheme inventory metrics are given in Table 6.1 where they averaged across 10 runs of each environment; some examples of morphemes from the inventories are given in

Section E.5.2.

Inventory, form, and meaning size Inventory entropy along with form and meaning size give sense of the general "shape" of the morpheme inventories—how large they are as a whole and what their components look like. With respect to inventory entropy, we find that generally, the image-observation environments results in higher entropy morphemes, which is to be expected with richer input source. On the other end of the interval, the (Vector, Attribute–value, Reconstruction) environment shows the lowest almost matching the lower bound of 4 bits.[4]

While both form length and meaning size remain small, their is a notable presence of sizes greater than 1. For form length, this suggests that the assumption that individual form tokens can be treated as words is not well founded; instead, the smallest meaningful units can comprise multiple tokens. For meanings, this indicates some degree of "fusionality" where the minimal unit of meaning corresponds to multiple atomic meanings at once and inseparably.

Inventory quality Insofar as morpheme bijectivity measures the quality of the morpheme inventories induced, we find that the quality across most environments is medium-to-low with only the reconstruction game with attribute–value vector observations averages above 0.5.[5] Generally speaking, the bijectivity values for the vector-based environments are higher than the those of the image-based environments, suggesting that the noiseless, pre-disentangled observations result in languages which are more easily captured by CSAR. The reference games in the image-based environments show especially low bijectivity potentially, in part, because the communication can reference low-level observation features without significant pressure towards generality.

Synonymy and polysemy Synonymy and polysemy, taken together, naturally correlate with morpheme bijectivity as the former metrics test for unidirectional correlation (e.g., how informative is a form for various meanings) while NPMI (the basis for bijectivity) takes bidirectional correlation (i.e., do a form and meaning correlate with each other exclusively). Empirically, the vector-based environments have more synonymy than polysemy while the image-based have this pattern reversed. Our initial hypothesis would be that synonymy would exceed polysemy insofar as polysemy incurs a cost with respect to communicative efficacy (i.e., ambiguity) while synonymy does not. That being said, the fuzzier relationship between the observations and the semantics in image environments (since it is a lossy conversion) compared to the one-to-one relationship between meanings and observations in the vector-based environments may explain part of the unexpected behavior.

Compositionality We find that morpheme bijectivity and toposim are relatively well correlated in our quantitative analysis, as is expected since they both aim to be measures of compositionality. On the other hand, we do not find any notable correlation between

[4]Since this environment as 4 attributes which can each taken on 4 values, that would correspond to $4 \cdot 4 = 16$ unique, equiprobable morphemes (order-dependence notwithstanding), corresponding to $\log_2 16 = 4$ bits of entropy.

[5]See Fig. 7.5 for a sense of the relationship between morpheme bijectivity and empirical inventory quality.

bijectivity/toposim and meaning size which could also be viewed a measure of compositionality; viz. fewer meaning components per morpheme correspond with a more one-to-one relationship between forms and atomic meanings (cf. holistic languages with many meaning components per form). The bijectivity–toposim correlation is also reflected in the progression from reference to set-reference to concept games in the image-based environments: not only does the toposim generally increase moving along the game progression (as presented in J. Mu et al. (2021)) but the morpheme bijectivity as well. This gives even stronger evidence for claims of J. Mu et al. (2021) as morpheme bijectivity is argued to be a better measure of compositionality than that of toposim (Section 7.5.4).

6.6 Discussion

Due to CSAR's strong performance and easy application to a wide variety of emergent language environments, it would be a valuable addition to the standard toolkit of emergent language analyses. In particular, it helps fill a gap of environment-agnostic methods for interpreting the ways that emergent languages convey meaning—a perennial question in the field. Down the road, this opens up research questions concerning the evolution of meaning in emergent language, such as those discussed in Brighton (2003), but with the ability to deal with the larger scale and particular difficulties of *deep learning-based* emergent communication.

Furthermore, morpheme inventories are a foundation for higher-level linguistic analyses of emergent language like inducing their syntactic structure. To skip the morpheme induction step would be comparable to attempting to understand the grammatical role of the letter C in English. Such analyses of the syntax of emergent language and beyond are critical to understanding how emergent and human language are similar and how they are different.

6.7 Conclusion

CSAR presents a strong platform for investigating the morphology of emergent language, demonstrating the ability to find minimal form–meaning pairs in both procedural and human language data. Given the morpheme inventory of an emergent languages we can not only analyze phenomena like synonymy and polysemy but also the typological features of emergent languages, determining which human languages they most closely resemble, if they resemble any. Such a study of morphology forms the foundation for the more general study of the linguistic features of emergent language and unlocks the door to the insights they can provide us about human language.

6.8 Limitations

Greed is not always good While the greediness of CSAR does simplify induction (conceptually and implementation-wise), improve runtime, and provide good partial inventories, it suffers from the same limitation inherent to greedy algorithms: it can get trapped in local optima. For example, it is possible to construct corpora for which a greedy approach is "misled" since certain heuristics require revision based on information encountered later in the induction process (e.g., preferring smaller versus larger forms).

We did consider non-greedy approaches to morpheme induction but ultimately decided not to pursue them in this work because (1) the greedy approach itself demonstrated strong performance and (2) initial attempts at non-greedy approaches (e.g., tree search) yielded intractable runtimes. For example, an error due to greediness might select morpheme B before morpheme A because B had a higher weight while A was ultimately correct. To select A instead of B, the morpheme candidates would have to be reordered which, without an efficient way to propose these order, worsens the time complexity from $O(n^c)$ to $O(n!)$. Related algorithms use iterative approaches (IBM models and Morfessor) or search (Brighton, 2003) to avoid the local minima that trap greedy approaches. Future work could incorporate such methods to improve upon the performance of CSAR for morpheme induction.

Bias towards smaller forms and meanings In light of the greedy approach, a further limitation of CSAR is its bias towards smaller forms and meanings due to the fact that mutual information is intrinsic bias towards frequently appearing forms and meanings (n.b.: smaller forms necessarily appear as or more often than larger, composite forms. While this bias is beneficial where emergent languages are genuinely decomposable into smaller units, in languages which do not decompose effectively, CSAR induces low quality inventories (i.e., having low morpheme bijectivities) with small forms and meanings rather than inducing more bijective inventories with holistic meanings (i.e., larger forms and meanings). Some heuristics balancing mutual information and bijectivity could address this while maintaining greediness, but ultimately non-greedy approaches might be necessary.

Limited emergent language data The other limitation of this paper relates to the type and breadth of emergent language data. In terms of type, since we do not have ground truth morpheme inventories for emergent language, we cannot directly evaluate CSAR's performance on the target domain (hence the validation with procedurally generated and human languages). In terms of breadth, without a larger and more representative sample of more systematically generated data we are unable to make definitive claims about the patterns and trends of morpheme inventories in emergent languages.

Chapter 7

Communicating with Induced Morphological Phrasebooks

Abstract

We build rule-based emergent language (EL) agents using induced morphological phrasebooks and test their communicative performance in the EL environment with its neural network agents. This contributes three things: First, it assesses the quality of the morphemes discovered by the induction algorithm *in situ*, which we find to be effective for communicating in the EL. Second, it allows us to uncover morphosyntactic properties of EL through ablating the morpheme induction and the phrasebook algorithms, showing that the ELs rely on repetition as well as morpheme ordering to convey meaning. Third, we find that the bijectivity of morphemes (measured via normalized pointwise mutual information), serves as a metric of compositionality that is more closely correlated with the ability of the phrasebook-agents to "speak" and "hear" an EL than existing metrics such as topographic similarity or bag-of-symbols disentanglement.[1]

7.1 Introduction

Deep learning-based emergent language (a.k.a. emergent communication) presents a fascinating way to study the origins and development of human language by observing how the interplay of functional pressures and inductive biases produce communication systems. Yet a major challenge in studying emergent languages is interpreting how they convey meaning—neural networks may invent communication systems which lack features of human language. Thus, a primary goal of emergent language research has been to investigate the linguistic structures present in emergent languages from morphology to semantics to syntax and how they compare with those of human language (Wal et al., 2020; Ueda et al., 2022, Chapter 2).

One technique introduced to discover the morphology of emergent language is CSAR (Chapter 6), an algorithm which induces morphemes (minimal form–meaning pairs) from an emergent language corpus of messages and their accompanying semantics. CSAR was initially validated on human and procedurally generated languages, yet its effectiveness on emergent

[1]Based on "Communicating in Emergent Language with an Induced Morphological Phrasebook" under review in the January 2026 cycle of the *Association for Computational Linguistics (ACL) Rolling Review.*

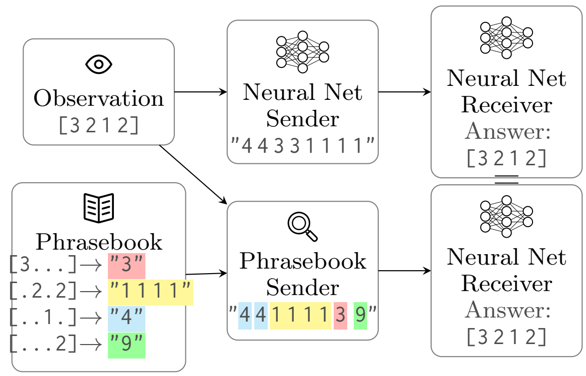


Figure 7.1: Main experimental setting where we compare a phrasebook-based agent with the original neural network-based sender. We also experiment with a phrasebook-based receiver.

languages was not tested due the unavailability of "ground truth" morphemes in emergent languages. To address this shortcoming, we test the morphemes induced by CSAR *in situ* by constructing algorithms to "speak" and "hear" emergent languages using the induced morpheme "phrasebooks" (illustrated in Fig. 7.1). These phrasebook-based agents are built on the assumption that the emergent language is *concatenatively compositional*, that is, that we can compose the meanings of morphemes by concatenating their forms into a message.

Analyzing the effectiveness of these phrasebook-based agents communicating in emergent language provides two insights. First, it reveals whether or not CSAR is uncovering a meaningful mapping between form and meaning for the emergent language, and, as a consequence, can serve as a good foundation for subsequent study of the structure of emergent languages. Second, it serves a probe into the morphosyntactic properties of emergent language (when we vary the morpheme induction and phrasebook algorithms).

Finally, with these results, we propose a new compositionality metric: *morpheme bijectivity*, defined as the weighted average of morphemes' normalized pointwise mutual information as induced by CSAR. This metric correlates better with the ability of the phrasebook-based agents (which presuppose compositionality) to use emergent language than existing metrics of compositionality like topographic similarity or bag-of-symbols disentanglement.

Impact Beyond validating the performance of CSAR and characterizing the morphosyntax of a particular set of emergent languages, this paper provides a more general paradigm for more principled evaluation of the linguistic features of emergent languages. That is, by constructing a procedural method for speaking and hearing the emergent language, we can directly probe for its various morphological and syntactical properties. This is much more akin to the process in linguistics of proposing theoretical models and testing their predictions than prior approaches in emergent language. As a whole, this paper, then, moves emergent language research closer towards its goal of discovering the nature of human language.

Structure Section 7.2 discusses prior work most relevant to this paper. Section 7.3 presents an overview of the environments, agents, and algorithms uses in the experiments. Section 7.4 briefly specifies the experiments and their results with a robust discussion in Section 7.5. The conclusion, discussion of limitations, and ethical considerations are presented in Sections 7.6 to 7.8, respectively.

7.2 Related Work

For a general overview of deep learning-based emergent language see Lazaridou et al. (2020). Generally speaking, this work builds on top of the morphological investigations into the structure of emergent language presented in Chapter 6 and ties in with other approaches, including Ueda et al. (2023), Lipinski et al. (2024), Carmeli et al. (2024), and Gilberti et al. (2025).

Levy et al. (2025) addresses the task machine translation of emergent communication into human language; namely, they employ unsupervised neural machine translation to translate image captions between emergent and human language. This is analogous to how we translate between emergent language messages and semantics, although we focus on white-box algorithms instead of deep learning methods. The morpheme induction and phrasebook algorithms assume compositionality in the emergent languages, and as such, intersect with prior work discussing the link between compositionality and generalization in emergent language (Chaabouni et al., 2020; Kharitonov et al., 2020a; Xu et al., 2022). Galke et al. (2024) discusses the relationship between compositionality in emergent language and its learnability by deep learning techniques analogous to our study of the learnability by statistical and rule-based methods.

The phrasebook-based agents we propose are, in effect, a rule-based machine translation (RBMT) system (e.g., Forcada et al. (2011)) where the language pair is the message and observation spaces of an emergent language, although the comparative simplicity of emergent language and its divergence from human language admit limited applicability of prior approaches from RBMT techniques with human language.

7.3 Methods

The code for this paper available under a free license at `http://example.com/supplemental-materials-for-review`.

7.3.1 Environment

The emergent language environment we use for our investigation is the reconstruction variant of the signaling game (similar to Chaabouni et al. (2019a) and Chaabouni et al. (2020)). The reconstruction game comprises two agents, a sender and a receiver. The sender makes an observation and produces a message which the receiver must use to reproduce the original observation (without any additional input); this can be seen as mimicking an autoencoder architecture where the central bottleneck layer is a sequence of discrete symbols. We select the reconstruction game because it in addition to being well-studied in literature, it provides a clear way of testing generalization via a held-out test set. Our implementation is based on the EGG framework (Kharitonov et al., 2019, MIT license).

In our implementation, the observations are a concatenation of one-hot vectors, which each represent a distinct attribute taking on a particular value. For example, if we have a game where observations comprise 2 attributes which can each take on one of 3 values, we would resent the observation [0 2] as follows:

$$\begin{bmatrix}0 & 2\end{bmatrix} \underset{\text{one-hot}}{\rightarrow} \begin{bmatrix}1 & 0\\0 & 0\\0 & 1\end{bmatrix} \underset{\text{flatten}}{\rightarrow} \begin{bmatrix}1 & 0 & 0 & 0 & 0 & 1\end{bmatrix},$$

with the concatenated one-hot vector being the input to the sender and desired output from the receiver. The messages in our environment are represented similarly: sequences of tokens represented as one-hot vectors (although the one-hot vectors are not concatenated because they are processed sequentially). Observations are sampled uniformly from all of the possible combinations of values for each attribute. Agents only see a subset of these possible observations at training time with the rest being held out for a test set.

7.3.2 Morpheme induction

Morpheme induction is the process of inferring minimal, meaningful form–meaning pairs from a corpus of messages with their accompanying meanings (i.e., an *annotated* corpus). Messages are taken to be sequences of atomic tokens while the accompanying meanings are represented as sets of atomic meaning tokens. Forms, in this case, are subsequences of tokens and meanings are subsets of the observation made by the sender. A given form–meaning pair is *meaningful* if the form and meaning correspond to each other in the language and *minimal* if the pair cannot be further decomposed into simpler pairs while maintaining the same meaning.

We use the CSAR algorithm to induce morphemes from the annotated emergent language corpora produced by the emergent language game (Chapter 6). Its outline is:

C ount co-occurrences of forms and meanings.
S elect pair with highest mutual information.
A blate selected pair from corpus.
R epeat process, starting at *Count*.

The intuition is that form–meaning pairs with high mutual information both correlate well with each other as well as occur frequently within the corpus.

7.3.3 Agents

Neural networks The sender agent comprises two fully-connected embedding layers, separated by a tanh activation, mapping from the observation to the embedding space which is given to an RNN to produce the message. The discrete vocabulary items are sampled using a Gumbel-Softmax layer (Maddison et al., 2017; Jang et al., 2017) before being passed to the RNN of the receiver agent, follows by two full-connected layers, as in the sender. The agents are trained on a mean-squared error objective. When the exact-match training accuracy reaches 99.99%, either the sender or receiver's parameters are reset to encourage better generalization (by default the receiver is reset twice before concluding training after reaching the accuracy threshold) (F. Li et al., 2019).

Algorithm 1 Outline of sender algorithm

```
morpheme :: tuple
  form :: list[int]
  meaning :: set[int]
  weight :: float
# Sorted by weight, descending
inventory :: list[morpheme]
observation :: set[int]
max_length :: int

message :: list[int]
while True:
  meanings_left = observation
  for morph in inventory:
    if morph.meaning ⊆ meanings_left:
      meanings_left -= morph.meaning
      message.extend(morph.form)
      if len(message) ≥ max_length:
        return message
      if meanings_left is empty:
        break # out of for loop
```

We use two different types of neural network-based agents in our empirical evaluation which we term *online* and *offline*. The online agents are simply the neural networks that were trained as part of the original emergent language environment, which means that the sender and receiver were jointly optimized and gradients backpropagated from receiver to sender. The offline agents, on the other hand, are trained directly from the annotated corpora in a fully supervised setting. Thus, the offline agents do not receive any additional information or benefit from joint optimization compared to the phrasebook-based agents which also operate based only on the annotated corpora.

Phrasebook The phrasebook agents map from observations to messages (senders) or messages to observations (receivers) by executing an algorithm given the morpheme inventory ("phrasebook") induced by CSAR. Both algorithms are founded on the assumption that the emergent languages are *concatenatively compositional* over morphemes; that is, the meanings present in morphemes can be combined by concatenating their corresponding forms.

The sender uses a greedy algorithm (Algorithm 1) which loops over the morphemes in the phrasebook from best to worst; when a morpheme's meaning is a subset of the observation, the corresponding form is added to the message under construction. The morpheme's meaning is them ablated from the observation, and the loop continues until all the meanings are matched or the message reaches the maximum length permitted by the environment. If the observation is exhausted before the max length is reached, the observation is reset to the original value and algorithm repeats from the beginning of the morpheme inventory. This results in multiple of the highest-weighed morphemes being employed in the message (this

Algorithm 2 Outline of receiver algorithm

```
# Sorted by weight, descending
inventory :: list[morpheme]
message :: list[int]
attributes :: list[set[int]]

# Cumulative weight of each meaning
meanings :: map[int, float]
for morph in inventory:
  n = count_matches(message, morph.form)
  for m in morph.meaning:
    meanings[m] += n * morph.weight
# Max-weighted value for each attribute
observation :: set[int]
for attr in attributes:
  observation += argmax(meanings ∩ attr)
return observation
```

turns out to be critical to phrasebook sender performance; see Section 7.5.3). When the maximum message length is reached, the morphemes are reordered by ascending positional *affinity* before being concatenated. Positional affinity is defined as the mean start index of that morpheme across the input corpus.

We also implement more sophisticated algorithms for constructing messages from an observation: best-first search and integer programming (used in the ablation experiments and briefly described in Appendix F.2).

The receiver algorithm (Algorithm 2) follows the same basic idea as the sender. The morphemes are looped over, testing to see if a morpheme's form is a subsequence of the message. If it the form is contained in the message, the morpheme's meaning is added to a meaning accumulator proportional to the morpheme's weight and the number of occurrences in the message. Unlike the sender, the form is not subtracted from the message, and the algorithm proceeds through the whole phrasebook only one time; thus, a sort of superposition of all mathcing morphemes is considered. After the phrasebook is exhausted, the final meaning is determined by taking the index of the max accumulator value for each attribute.

7.4 Experiments

In this section, we describe the experiments performed and give their results with a full discussion and analysis of the results in Section 7.5. See Appendix F.3 for details on computing resources used.

7.4.1 Phrasebook vs. neural network agents

We begin our empirical evaluation by comparing the performance of the phrasebook-based agents with the online and offline neural network-based agents across ∼1200 seeds of the

Figure 7.2: Accuracies across ∼1200 random seeds for various agent pairs. *Online* and *Offline* are the neural network-based agents, while *PB* refers to the phrasebook-based agent.

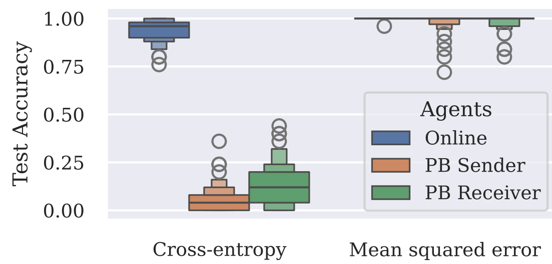


Figure 7.3: Accuracy plot for loss function ablation experiment. Full plots in Appendix F.6.

reconstruction game (Fig. 7.2). We look at exact-match accuracy on train and test sets for the original online–online setting as well as to and from the offline and phrasebook agents. This setting uses 4-attribute, 4-value observations which means that random chance performance is $\frac{1}{4^4} \approx 0.4\%$ (see Appendix F.1 for further hyperparameters). Some qualitative examples of inventories and messages sent by the neural sender are given in Appendix F.4. A quantitative summaries of the morpheme inventories (e.g., inventory size, synonymy) is included in Appendix F.5.

7.4.2 Environment ablation

We follow up our primary evaluation of our phrasebook-based agents with variations of the hyperparameters of the emergent language environment (and the online agents) to determine what environmental factors make emergent languages more or less intelligible to morpheme induction and the accompanying phrasebook agents. An example of the results is shown in Fig. 7.3 with all results given in Appendix F.6. Each setting was run for 100 random seeds (some settings had fewer than 100 runs converge in the initial training).

In particular, we vary the following hyperparameters in the environment and online agents:

***n* attributes, *n* values** (default: $(4, 4)$).
test proportion Proportion of possible observations held out for the test set (default: 10%).
vocabulary size Number of distinct types usable in a message (default: 64).

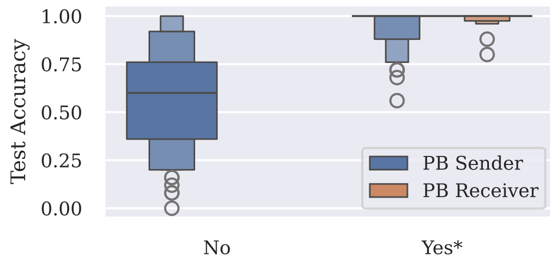


Figure 7.4: Accuracy plots various for *repeat morpheme* phrasebook sender ablation. Full plots in Appendix F.7.

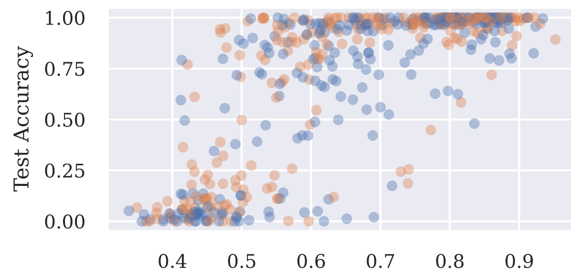
max message length Maximum number of tokens in a message; padded with end-of-sentence token if less than max (default: 8).
GS straight through Use a stochastic one-hot vector during the forward pass through the Gumbel-Softmax layer (default: no).
Agent reset schedule Sequence of sender/receiver resets during training (default: (receiver, receiver)).
Reset optimizer parameters Reset Adam optimizer parameters along with the agent parameters during reset (default: yes).
RNN cell type Which RNN cell type to use (Elman, LSTM, GRU; default: GRU).
loss function Which loss function to train the online agents with (cross-entropy or mean squared error; default: MSE).
***n* embedding layers** Number of fully-connected layers before/after the sender's/receiver's RNN (default: 2).
***n* RNN layers** (default: 1)

7.4.3 Phrasebook agent ablation

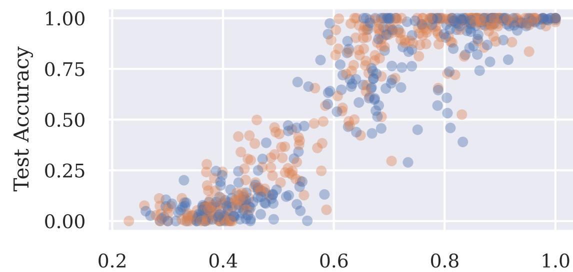
In this section we test ablations and variations of the morpheme induction process as well as the phrasebook sender and receiver algorithms. These variations will elucidate what aspects of the algorithms and morphemes are most salient to using the emergent language. An example of the results is shown in Fig. 7.4 with all results given in Appendix F.7.

In the morpheme induction algorithm we vary:
max inventory size Use the top-k morphemes for the inventory (default: $k = \infty$).
max form length Consider forms up to length l (default: $l = \infty$).
max meaning size Consider meanings up to size s (default: $s = \infty$).
weight method What probabilistic metric to use in selecting weighting and selecting morpheme candidates (default: mutual information); other methods given in Appendix F.7.
strip EoS token Remove the padding/end-of-sentence token from messages before induction (default: yes).
search best Apply a lookahead heuristic in ablating ambiguous morphemes (default: yes).

In the phrasebook sender we vary:
morpheme selection method Whether to use greedy algorithm (default) for selecting morphemes, search, or integer programming.



(a) Corpus topographic similarity ($R^2 = 0.57$)



(b) Morpheme bijectivity ($R^2 = 0.85$)

Figure 7.5: Plot of two different potential predictors of phrasebook sender (blue) and receiver (orange) accuracy: corpus topographic similarity (top) and mean NPMI of morphemes (bottom).

form order How to order the forms in the message (default: *affinity*); other method includes *insertion* order (no reordering) and *shuffled* (at the morpheme level).

repeat morpheme Permit repetitions of the same morpheme (default: yes); if no, exit after all meanings are accounted for.

ablate meaning Track meaning atoms remaining in the observations (default: yes); if no, perform a single pass through the inventory and apply any matching morpheme (increases diversity of morphemes used).

In the phrasebook receiver we vary:

ablate form Mask out form tokens as they are matched with morphemes (default: no); if no, match all morphemes in the inventory that apply.

idempotent form Ignore repetition of the same morpheme in a message (default: no); analogous to *repeat morpheme* for the sender algorithm.

Each variation is applied to the base configuration independently and is tested against the same 100 corpora generated with different random seeds.

7.4.4 Compositionality

In our final experiment, we investigate how the performance of the phrasebook agents correlates with topographic similarity and a compositionality metric we introduce, *morpheme bijectivity* (Fig. 7.5).We also tested bag-of-symbols disentanglement (Chaabouni et al., 2020) and a mutual information-based variant of the morpheme bijectivity metric (Fig. F.7). These metrics are computed reusing the training corpora and phrasebook agent accuracies from the environment ablation experiments (Section 7.4.2) in order to test against a wider variety of languages and phrasebook agent performances.[2] Since the independent variable values (i.e., toposim and intrinsic inventory metrics) are not evenly distributed, we employ stratified sampling across 5 buckets to avoid overweighting the most prevalent buckets.

Topographic similarity (a.k.a., "toposim") (Brighton et al., 2006; Lazaridou et al., 2018) is the most popular metric of compositionality in the emergent language literature and is defined as the Spearman rank correlation coefficient (ρ) between pairwise distances in the observation space and distances in the message space. We use Levenshtein distance as the metric for the message space and Hamming distance for the observation space (since the observations are represented by sets of a fixed size).

We compare toposim with a new metric, *morpheme bijectivity*, derived from CSAR's morpheme inventory. Morpheme bijectivity is defined as the mean normalized pointwise mutual information[3] (NPMI) weighted by the prevalence[4] of morphemes in the inventory given by CSAR. The intuition with mean NPMI is that morphemes with higher NPMI's (i.e., closer to having a one-to-one relationship) result in observation–message mappings that are less ambiguous and easier to process.

7.5 Discussion

7.5.1 Communicating in emergent language

We observe (in Fig. 7.2) that the online–online parings perform near 100% accuracy on training and test data with few outliers. Both the offline sender and offline receiver perform almost as well, though with a marked decrease on the test data; this shows that most—though possibly not all—of in the information necessary for generalizing to the test data is is available in the annotated corpora alone (cf. the online agents "overfitting" to each other).

The phrasebook-based agents, while underperforming the neural network-based agents, perform quite well with median scores of 100% on the test set and seldom falling below 80%. This, in turn, confirms the primary premise in question in this paper: it is possible to communicate in emergent language with phrasebook agents derived from an induced morphological inventory. Nevertheless, it is very easy to distinguish between the messages generated by the neural sender compared to the phrasebook sender (Appendix F.4); the former is more varied in both its morpheme selection and its order compared to the rigid message structure of the phrasebook sender.

7.5.2 Robustness of morpheme induction and phrasebook agents

Varying the hyperparameters of the emergent language environment and the online agents allows us to get a fuller picture of how the morpheme induction, sender, and receiver

[2] We exclude variations on the hyperparameters n values, n attributes, and test proportion because they skew the relative difficulty of the reconstruction game.

[3] Computed for form–meaning pairs before any ablations.

[4] The joint probability of the form–meaning pair after previous ablations are applied.

algorithms perform (Figs. F.3, F.4 and 7.3). We categorize the results by the online and phrasebook agent performances:

High online, high phrasebook Both the online agents and the phrasebook agents perform well (>80%)[5], suggesting that generalization is achieved by the online agents employing a method interpretable to both sender and receiver phrasebook agents. This includes the majority of the settings we tested in our ablations suggesting a fair degree of robustness to variations in the hyperparameters of the environment.

Low online, low phrasebook Both the online agent and the phrasebook agents struggle to generalize well (even after the online agents converge on the training set). These settings include $n_{\text{attribute}} = 2$ (0%) and $n_{\text{value}} = 2$ (0%) and, to a lesser degree, a test proportion of 75% ($[30\%, 45\%]$). The results on these settings are expected insofar as the online agents serve as an upper bound for phrasebook agent performance, that is, the phrasebook agents cannot somehow generalize beyond what possible in the original language.

High online, low phrasebook The online agents maintain a high test performance while the phrasebook agents significantly underperform the online agents; the most notable examples of this are the non-GRU RNN cell type ($[90\%, 100\%]$ vs. $[15\%, 35\%]$), cross-entropy loss (95% vs. $[5\%, 15\%]$), and a vocabulary size of 4 (100% vs. 5%). These setting highlight the fact that the online agents are capable of generalizing to unseen observation in a way that is largely opaque to the morpheme induction algorithm and/or the phrasebook algorithms, that is, in a way that violates the assumption concatenative compositionality. For example, in the case of a small vocabulary, size it would not be possible to create a one-to-one mapping between specific types in the vocabulary and specific values for specific attributes. Two alternatives for conveying meaning include: (1) double articulation where groups of meaningless form tokens comprise meaningful "words" and (2) the semantics of individual types exist in an entangled, non-linear embedding space which neural networks can learn but CSAR cannot. Although further investigation would be required to determine if either of these two cases hold.

7.5.3 Morphosyntax in the ELs

Looking at the performance variations in response to changing parameters of the induction and phrasebook agents, we can infer various properties about the morphology and syntax of the emergent languages in question. With respect to the morpheme induction, we see two subtle patterns: (1) performance goes slightly up when increasing the max meaning size from 1 to 2 (Fig. F.5) and (2) performance goes slightly *down* with max form length increasing above 1 (Fig. F.5). The former pattern suggests that the emergent languages show some small degree of fusionality/cumulative exponence, while the latter indicates that considering multi-token forms is not relevant to the morphology of the emergent languages.

Turning our attention to the variations in the sender algorithm, we see that one of the most important factors in high performance is permitting the repetition of morphemes. When

[5] Parenthesized numbers are median accuracies.

the algorithm employs just a single morpheme per meaning atom, the performance is notably worse (with a median accuracy of 60% instead of 100%; Fig. F.6). Similarly, when the sender algorithm does not ablate meanings and uses subsequent lower-weighed morphemes for a given meaning atom in addition to the higher weighed meanings (i.e., meanings are not ablated), performance drops (Fig. F.6). These observations suggest that repetition of highly-weighted morphemes is a meaningful morphosyntactic feature of the emergent languages studied.

In addition to the importance of repetition, we also found the order of morphemes had a notable impact on the sender agent's performance, with performance decreasing when we perturb morphemes from their preferred place in the message (i.e., the *affinity* method; Fig. F.6). This indicates a syntactic rule (albeit a simple one) that governs how morphemes should be combined. Finally, the variations on the receiver algorithm did not have large impacts on the phrasebook agent performance (Fig. F.6).

7.5.4 Compositionality in emergent language

Compositionality is one of the most talked about topics in emergent language research, and the *in situ* experiments with phrasebook agents provide a uniquely in-depth investigation of this phenomenon. Namely, the morpheme induction algorithm and phrasebook agents are built on the following assumption: the meanings forming the observation correspond to substrings of the message in a disentangled way, such that the substrings can otherwise be recombined independently of each other while retaining their original meaning. This aligns closely with the typical definition of compositionality in emergent language, namely that "[t]he meaning of a compound expression is a function of its parts and of the way they are syntactically combine" (Partee et al., 1984), where syntactic combination is operationalized as concatenation of the forms in emergent language.

Essentially, the phrasebook agents should perform well if and only if (1) the emergent language is concatenatively compositional, (2) the morpheme induction algorithm actually induces meaningful morphemes, and (3) the sender and receiver algorithms sufficiently capture the syntax of the emergent language. Insofar as we assume (2) and (3) to be true, we can use the performance of the phrasebook agents as a proxy for compositionality. This notion of compositionality is deeper than more familiar metrics like topographic similarity because it not only takes into account the externally visible correlations between observations and messages but also tests these correlations with the original emergent language speakers and hearers (i.e., the online agents). We could, in a way, see these experiments being closer to "field linguistics" compared to the "corpus linguistics" approach of toposim. Continuing the analogy: field linguistics is more resource intensive than corpus linguistics, and likewise for testing phrasebook agents *in situ* compared to computing toposim.

Yet, we find that morpheme bijectivity addresses the shortcomings of both measures of compositionality: First, it is more predictive of phrasebook agent performance than toposim (Fig. 7.5; $R^2 = 0.86$ vs. $R^2 = 0.57$).[6] And second, it can be computed with only the annotated corpus, not needing access to the original emergent language environment.

[6]Bag-of-symbols underperforms both of these as a predictor with $R^2 = 0.29$ (Fig. F.7).

Compositionality without generalization An ancillary finding of the experiments across different environments is that they provide further evidence for the notion that neural network agents do not need to use concatenative compositionality to in order to generalize (Kharitonov et al., 2020a). Indeed, in the majority of our settings, we see that the performance of the online and phrasebook agents are relatively matched, suggesting that compositionality corresponds with generalization performance. But in certain settings (e.g., cross-entropy loss, Elman and LSTM RNN cells, and a small vocabulary size of 4; Figs. F.3 and F.4), there is a clear bifurcation: the online agents generalize well while the phrasebook agents do not (and even have trouble fitting the training data). This shows that the neural network agents are capable of compositional as well as non-compositional generalization and may favor one over the other for non-obvious reasons (cf. GRUs vs. LSTMs). Furthermore, the lower correlation of toposim with phrasebook agent performance suggests that it might not be the best metric for detecting such instances of non-compositionality in comparison to the higher predictive power of morpheme bijectivity derived from CSAR.

7.6 Conclusion

The above experiments have demonstrated the effectiveness of CSAR and the phrasebook algorithms as foundations for studying the morphological and syntactic structures of emergent language. Beyond the particular languages studied here, the methods presented demonstrate a more general paradigm of investigating the linguistic features of emergent languages by ablating explicit models of communication and testing them *in situ*. Such an approach is better grounded in the use of the emergent language itself rather than merely hypothesizing about the relationship between surface-level features and their relationship with the language itself. With the tools and techniques presented in this paper, the field of emergent language is better equipped to explore the nature of human language.

7.7 Limitations

We identify the following primary limitations in our work:

1. The phrasebook sender and receiver algorithms were designed more or less heuristically rather than being motivated by some particular feature of how the online agents convey meaning in emergent language. A more principled design approach, grounded in how the neural network agents express meaning, could yield even better performance for the phrase-book agents.
2. We have performed limited qualitative of evaluation of the emergent languages, and in particular the strategies used by the online neural network agents to generalize without simple compositionality. Observing the emergent languages more closely is an important step into figuring out precisely why morpheme induction and/or phrasebook algorithms fail while the neural networks succeed.
3. Although the methods we present are applicable to a wide variety of emergent language environments, the empirical evaluations were performed with a limited variety of

environments, so it is not possible to determine whether or not the experimental results are fully applicable to other environments from the presented experiments alone.

4. The morpheme bijectivity metric is limited to the emergent languages that CSAR can handle, specifically emergent languages with discrete, decomposable observations. While CSAR, and thereby morpheme bijectivity, could theoretically be expanded, it currently cannot be applied to continuous observations.

7.8 Ethical Considerations

We do not identify any ethical considerations or potential risks relevant to this work as it constitutes basic machine learning and linguistics research using synthetic data.

Chapter 8

Conclusion

This thesis has endeavored to sort through chaos of emergent language research at both the micro and macro level. The micro level of emergent language research concerns the analysis of particular emergent language environments, discerning what characteristics may or may not be present in the utterances and behaviors of the agents. The macro level of emergent language research concerns the broader direction of the field and meta-analysis of contributions across many environments and approaches. These levels are interrelated as the difficulty of interpretation at the micro level is reflected in the macro level; thus, it has been necessary to address both at the same time.

Bringing order to the micro level

XferBench and CSAR present novel analytical algorithms for investigating emergent languages as they are both applicable to a wide variety of environments and grounded in the ultimate goals of emergent language techniques. Both algorithms have been designed with large-scale quantitative analysis in mind where many emergent languages can be searched over for interesting structures with minimal human input. Such scale is necessary given the vast design of space of emergent languages and the difficulty of finding structure within it. Thus, these methods can serve as a first line of analysis to situate the otherwise difficult to interpret output of emergent language simulations.

XferBench and engineering applications XferBench is specifically tailored to advancing emergent languages towards applications in engineering, particularly those in which emergent languages act as synthetic human language data. For any given emergent language, it immediately gives a sense of how well it would function as synthetic human language data. For simpler emergent languages, this correlates with factors such as the distribution of tokens but could eventually indicate higher levels of structure like syntax for more sophisticated emergent languages. The results of XferBench, for example, can then be used to select hyperparameters which encourage realism (as in Chapter 5) or filter out emergent languages which fail to even produce realistic distributional patterns.

Looking forward, XferBench has much room to improve through scaling up in terms of model size, data sizer, and downstream tasks. Such scaling would essentially give the metric a "higher resolution" insofar as it would be able to distinguish more nuances in the input data.

Improvements could also be made to the interpretability of XferBench by adding probes for features such as syntax or morphology which would broaden its applicability to linguistics analyses as well.

CSAR and linguistic applications CSAR, on the other hand, is oriented more directly toward the linguistic applications of emergent language research. Naturally, taking stock of the morphology of an emergent language factors into the analysis of further linguistic structures like lexical distributions and strategies for conveying meaning. But even apart from explicitly linguistic applications, CSAR's qualitative and quantitative outputs can serve as a sort of preview and statistical summary for an emergent language, far more interpretable compared to the labor-intensive and often ineffectual process of trying to decode an emergent language by hand.

CSAR itself could be improved through more sophisticated algorithms which are better able to distinguish between compositional and holistic languages as well as work with a broader range of semantics (e.g., embeddings, quantized continuous spaces). It could also be extended to function on a syntactic level, where a corpus translated into morphemes can be given to a *structure* induction algorithm modeled after CSAR. These induced patterns can then be tested *in situ* in a similar fashion to the approach of Chapter 7; these tests, subsequently, yield an even finer grained account of the validity of the induced structural patterns.

Bringing order to the macro level

On the larger scale, this thesis enables the direct comparison of emergent languages from disparate environments, the same way we would expect any two neural architectures to be comparable on a given task. Again, due to the unique challenges of emergent language, methodologies that can be taken for granted in other areas of machine learning require particular attention as has been done in this work. What the result is, though, is a field of research that is more familiar at the meta level: cumulative, incremental findings which gradually reveal the most effective approaches and occasionally yielding major breakthroughs.

Going forward, my intuition is that a more prudent research program is to start with more classical approaches and gradually replacing components with deep neural networks. Even with these new tools for cutting through chaos of emergent languages, there is an element of luck required when starting with end-to-end neural networks and hoping to find linguistically relevant patterns (and deep emergent language research does not seem to have been "lucky" thus far). Furthermore, such intermediate results are harder to ground in the relevant linguistic theories of the evolution of human language whereas starting in the better grounded classical models allows for readier insights along the way.

XferBench and engineering applications Beyond the analysis of individual environments, XferBench demonstrates the potential to compare statistical realism across many different varieties of environments. This would direct the field towards environmental innovations which genuinely increase realism while pruning design choices which add complexity without a corresponding benefit. XferBench, then, provides an opportunity for hill-climbing towards more useful emergent languages for downstream tasks. Furthermore, the fact that XferBench is automated and quantitative encourages larger-scale analysis, that is, looking across hundreds or even thousands of emergent languages in a given study, decreasing statistical noise as well as results that are peculiar to specific hyperparameters. These improvements to methodology of emergent language have at least some potential to harness some of the success that has characterized the broader field of deep learning.

CSAR and linguistic applications CSAR makes emergent languages more amenable to the kinds of analysis that we see in classical emergent language which often presuppose access to meaningful units (as opposed to mere meaningful unit *components* in a doubly articulated language). This, in turn, can replace the somewhat shaky assumptions like "one token equals one word" that linguistic analyses on emergent languages were previously forced to make. Establishing the presence of such basic structures in emergent languages is a critical step in better aligning the paradigms of classical and deep learning-based emergent language. A better alignment between these approaches would likely grant the next decade of deep learning-based emergent language research a broader impact than the last.

Epilogue

The year is 2026. Large language models have even taken over the academic and popular consciousness alike. Emergent language has maintained its niche *status quo* since the time described in the Prologue, but it cannot match the optimism of "GenAI". While Sutton's "Bitter Lesson" seems to hold more now than ever, deep reinforcement learning is often now just a method for augmenting LLMs. This thesis hardly mentions LLMs let alone contributes to their advancement—how are its pages still relevant, then?

These pages are still relevant because the foundational questions of this thesis are still relevant. LLMs, while immensely successful, still have their problems. The question of how grounded their "knowledge" is in a rich, multi-modal world still hangs in the air. The concerns of surveillance, privacy, and copyright have increases, not decreased. Low resource languages are still difficult to work with. In fact, LLMs sometimes even create their own emergent languages which need their own interpretation.

But secondly, the questions surrounding the evolution of language are fundamentally *scientific* questions. LLMs can certainly assist in answering some of these questions, but a model that hardly mimics how humans acquire language is limited in what it can tell us about the internal and external pressures begetting human language. No degree of real-world utility or benchmark topping will provide the understanding that characterizes scientific questions. The project of building high-fidelity simulations of *de novo* language evolution, on the other hand, have potential to address the deeper questions of the origins of human language. While emergent language certainly lacks the glamour of deep RL in 2018 or LLMs in 2026, so long as there are linguists inquiring into language (and the Lindy effect is on my side, here), the contributions of this thesis will have a home.

Bibliography

Akhtar, Mubashara, Omar Benjelloun, Costanza Conforti, Pieter Gijsbers, Joan Giner-Miguelez, Nitisha Jain, Michael Kuchnik, Quentin Lhoest, Pierre Marcenac, Manil Maskey, Peter Mattson, Luis Oala, Pierre Ruyssen, Rajat Shinde, Elena Simperl, Goeffry Thomas, Slava Tykhonov, Joaquin Vanschoren, Jos van der Velde, Steffen Vogler, and Carole-Jean Wu (2024). "Croissant: A Metadata Format for ML-Ready Datasets". In: DEEM '24. Santiago, Chile: Association for Computing Machinery, pp. 1–6. ISBN: 9798400706110. DOI: 10.1145/3650203.3663326. URL: https://doi.org/10.1145/3650203.3663326.

Akiba, Takuya, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama (2019). "Optuna: A Next-Generation Hyperparameter Optimization Framework". In: *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631.

Andreas, Jacob (Apr. 2019). "Measuring Compositionality in Representation Learning". In: arXiv: 1902.07181. URL: http://arxiv.org/abs/1902.07181 (visited on 12/23/2021).

Artetxe, Mikel, Sebastian Ruder, and Dani Yogatama (July 2020). "On the Cross-lingual Transferability of Monolingual Representations". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, pp. 4623–4637. DOI: 10.18653/v1/2020.acl-main.421. URL: https://aclanthology.org/2020.acl-main.421.

Auersperger, Michal and Pavel Pecina (2022). *Defending Compositionality in Emergent Languages*. eprint: arXiv:2206.04751. URL: https://arxiv.org/abs/2206.04751.

Baker, Bowen, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch (2020). "Emergent Tool Use From Multi-Agent Autocurricula". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=SkxpxJBKwS.

Baker, Mark (1985). "The Mirror Principle and Morphosyntactic Explanation". In: *Linguistic Inquiry* 16.3, pp. 373–415. ISSN: 00243892, 15309150. URL: http://www.jstor.org/stable/4178442 (visited on 08/21/2023).

Bar-Yam, Yaneer (2002). "General features of complex systems". In: *Encyclopedia of Life Support Systems (EOLSS), UNESCO, EOLSS Publishers, Oxford, UK* 1.

Baroni, Marco (Feb. 2020). "Linguistic generalization and compositionality in modern artificial neural networks". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 375.1791. arXiv: 1904.00157, p. 20190307. ISSN: 0962-8436, 1471-2970. DOI: 10.1098/rstb.2019.0307.

Batali, John (1998). "Computational simulations of the emergence of grammar". In: *Approaches to the evolution of language: Social and Cognitive bases*. Cambridge: Cambridge University Press, pp. 405–426.

Batsuren, Khuyagbaatar, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova (July 2022). "The SIGMORPHON 2022 Shared Task on Morpheme Segmentation". In: *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Ed. by Garrett Nicolai and Eleanor Chodroff. Seattle, Washington: Association for Computational Linguistics, pp. 103–116. DOI: 10.18653/v1/2022.sigmorphon-1.11. URL: https://aclanthology.org/2022.sigmorphon-1.11/.

BBC (2016). *Artificial intelligence: Go master Lee Se-dol wins against AlphaGo program*. https://www.bbc.com/news/technology-35797102. Accessed: 2026-02-07.

Bergstra, James, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl (2011). "Algorithms for Hyper-Parameter Optimization". In: *Advances in Neural Information Processing Systems*. Ed. by J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger. Vol. 24. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.

Bisk, Yonatan, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian (Nov. 2020). "Experience Grounds Language". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 8718–8735. DOI: 10.18653/v1/2020.emnlp-main.703. URL: https://www.aclweb.org/anthology/2020.emnlp-main.703.

Blevins, Juliette (2004). *Evolutionary phonology: The emergence of sound patterns*. Cambridge University Press.

Blum, Frederic, Carlos Barrientos, Adriano Ingunza, Damián E Blasi, and Roberto Zariquiey (2023). "Grammars Across Time Analyzed (GATA): a dataset of 52 languages". In: *Scientific Data* 10.1, p. 835.

Blumenkamp, Jan and Amanda Prorok (2020). *The Emergence of Adversarial Communication in Multi-Agent Reinforcement Learning*. eprint: arXiv:2008.02616. URL: https://arxiv.org/abs/2008.02616.

Bogin, Ben, Mor Geva, and Jonathan Berant (2018). *Emergence of Communication in an Interactive World with Consistent Speakers*. eprint: arXiv:1809.00549. URL: https://arxiv.org/abs/1809.00549.

Bojar, Ond rej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri (Aug. 2016). "Findings of the 2016 Conference on Machine Translation". In: *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, pp. 131–198. URL: http://www.aclweb.org/anthology/W/W16/W16-2301.

Bojar, Ondrej, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna (June 2014). "Findings of the 2014 Workshop on Statistical Machine Translation". In: *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, pp. 12–58. URL: http://www.aclweb.org/anthology/W/W14/W14-3302.

Boldt, Brendon and David R. Mortensen (2022a). *Mathematically Modeling the Lexicon Entropy of Emergent Language*. arXiv: 2211.15783 [cs.CL]. URL: https://arxiv.org/abs/2211.15783.

— (2022b). *Modeling Emergent Lexicon Formation with a Self-Reinforcing Stochastic Process*. eprint: arXiv:2206.11146. URL: https://arxiv.org/abs/2206.11146.

— (2022c). "Recommendations for systematic research on emergent language". In: arXiv: 2206.11302.

— (2024a). "A Review of the Applications of Deep Learning-Based Emergent Communication". In: *Transactions on Machine Learning Research*. ISSN: 2835-8856. URL: https://openreview.net/forum?id=jesKcQxQ7j.

— (2024b). *ELCC: the Emergent Language Corpus Collection*. arXiv: 2407.04158 [cs.CL]. URL: https://arxiv.org/abs/2407.04158.

— (June 2024c). "XferBench: a Data-Driven Benchmark for Emergent Language". In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Ed. by Kevin Duh, Helena Gomez, and Steven Bethard. Mexico City, Mexico: Association for Computational Linguistics, pp. 1475–1489. DOI: 10.18653/v1/2024.naacl-long.82. URL: https://aclanthology.org/2024.naacl-long.82.

— (Nov. 2025a). "Morpheme Induction for Emergent Language". In: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Ed. by Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng. Suzhou, China: Association for Computational Linguistics, pp. 25275–25290. ISBN: 979-8-89176-332-6. URL: https://aclanthology.org/2025.emnlp-main.1284/.

— (Nov. 2025b). "Searching for the Most Human-like Emergent Language". In: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Ed. by Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng. Suzhou, China: Association for Computational Linguistics, pp. 23289–23307. ISBN: 979-8-89176-332-6. DOI: 10.18653/v1/2025.emnlp-main.1188. URL: https://aclanthology.org/2025.emnlp-main.1188/.

Bosc, Tom (2022). "Varying meaning complexity to explain and measure compositionality". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=BnGzfmZ07bq.

Bosc, Tom and Pascal Vincent (2022). "The Emergence of Argument Structure in Artificial Languages". In: *Transactions of the Association for Computational Linguistics* 10, pp. 1375–1391.

Bouchacourt, Diane and Marco Baroni (2018). *How agents see things: On visual representations in an emergent language game*. eprint: arXiv:1808.10696. URL: https://arxiv.org/abs/1808.10696.

— (2019). *Miss Tools and Mr Fruit: Emergent communication in agents learning about object affordances*. eprint: arXiv:1905.11871. URL: https://arxiv.org/abs/1905.11871.

Bracci, S., J. Mraz, A. Zeman, G. Leys, and H. Op de Beeck (Apr. 2023). "The representational hierarchy in human and artificial visual systems in the presence of object-scene regularities". In: *PLoS Comput Biol* 19.4, e1011086.

Brandizzi, Nicolo', Davide Grossi, and Luca Iocchi (2022a). "RLupus: Cooperation through the emergent communication in The Werewolf social deduction game". In: *Intelligenza Artificiale* 15.2, pp. 55–70. URL: https://content.iospress.com/articles/intelligenza-artificiale/ia210081.

Brandizzi, Nicolo' and Luca Iocchi (2022b). "Emergent communication in human-machine games". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=rqLgeQWCXZ9.

Brighton, Henry (2003). "Simplicity as a driving force in linguistic evolution". PhD thesis. Edinburgh, UK: University of Edinburgh. URL: https://era.ed.ac.uk/handle/1842/23810.

Brighton, Henry and Simon Kirby (2006). "Understanding Linguistic Evolution by Visualizing the Emergence of Topographic Mappings". In: *Artificial Life* 12.2, pp. 229–242. DOI: 10.1162/artl.2006.12.2.229.

Brockman, Greg, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba (2016). *OpenAI Gym*. eprint: arXiv:1606.01540. URL: https://arxiv.org/abs/1606.01540.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer (1993). "The Mathematics of Statistical Machine Translation: Parameter Estimation". In: *Computational Linguistics* 19.2. Ed. by Julia Hirschberg, pp. 263–311. URL: https://aclanthology.org/J93-2003/.

Bullard, Kalesha, Douwe Kiela, Franziska Meier, Joelle Pineau, and Jakob Foerster (2021). *Quasi-Equivalence Discovery for Zero-Shot Emergent Communication*. arXiv: 2103.08067 [cs.MA].

Bullard, Kalesha, Franziska Meier, Douwe Kiela, Joelle Pineau, and Jakob Foerster (2020). *Exploring Zero-Shot Emergent Communication in Embodied Multi-Agent Populations*. eprint: arXiv:2010.15896. URL: https://arxiv.org/abs/2010.15896.

Bybee, Joan L. (1985). *Morphology: A study of the relation between meaning and form*. John Benjamins.

Cao, Kris, Angeliki Lazaridou, Marc Lanctot, Joel Z Leibo, Karl Tuyls, and Stephen Clark (2018). *Emergent Communication through Negotiation*. eprint: arXiv:1804.03980. URL: https://arxiv.org/abs/1804.03980.

Carlini, Nicholas, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel (2021). *Extracting Training Data from Large Language Models*. arXiv: 2012.07805 [cs.CR]. URL: https://arxiv.org/abs/2012.07805.

Carmeli, Boaz, Yonatan Belinkov, and Ron Meir (Aug. 2024). "Concept-Best-Matching: Evaluating Compositionality In Emergent Communication". In: *Findings of the Association for Computational Linguistics: ACL 2024*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, pp. 3186–3194. DOI: 10.18653/v1/2024.findings-acl.189. URL: https://aclanthology.org/2024.findings-acl.189/.

Carmeli, Boaz, Ron Meir, and Yonatan Belinkov (2022). *Emergent Quantized Communication*. eprint: arXiv:2211.02412. URL: https://arxiv.org/abs/2211.02412.

Chaabouni, Rahma, Eugene Kharitonov, Diane Bouchacourt, Emmanuel Dupoux, and Marco Baroni (July 2020). "Compositionality and Generalization In Emergent Languages". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, pp. 4427–4442. DOI: 10.18653/v1/2020.acl-main.407. URL: https://aclanthology.org/2020.acl-main.407/.

Chaabouni, Rahma, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni (2019a). "Anti-efficient encoding in emergent communication". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc.

— (2021). "Communicating artificial neural networks develop efficient color-naming systems". In: *Proceedings of the National Academy of Sciences* 118.12, e2016569118. DOI: 10.1073/pnas.2016569118. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2016569118. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2016569118.

Chaabouni, Rahma, Eugene Kharitonov, Alessandro Lazaric, Emmanuel Dupoux, and Marco Baroni (2019b). *Word-order biases in deep-agent emergent communication*. eprint: arXiv:1905.12330. URL: https://arxiv.org/abs/1905.12330.

Chaabouni, Rahma, Florian Strub, Florent Altché, Eugene Tarassov, Corentin Tallec, Elnaz Davoodi, Kory Wallace Mathewson, Olivier Tieleman, Angeliki Lazaridou, and Bilal Piot (2022). "Emergent Communication at Scale". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=AUGBfDIV9rL.

Chang, Tyler A. and Benjamin K. Bergen (Jan. 2022). "Word Acquisition in Neural Language Models". In: *Transactions of the Association for Computational Linguistics* 10, pp. 1–16. ISSN: 2307-387X. DOI: 10.1162/tacl_a_00444. eprint: https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl_a_00444/1986589/tacl_a_00444.pdf. URL: https://doi.org/10.1162/tacl%5C_a%5C_00444.

Chen, Siqi, Yang Yang, and Ran Su (2022). "Deep reinforcement learning with emergent communication for coalitional negotiation games." In: *Mathematical biosciences and engineering : MBE* 19 5, pp. 4592–4609.

Chen, Yang, Liangxuan Guo, and Shan Yu (2023). *Emergence of Symbols in Neural Networks for Semantic Understanding and Communication*. eprint: arXiv:2304.06377. URL: https://arxiv.org/abs/2304.06377.

Chevalier-Boisvert, Maxime, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio (2018). "Babyai: A platform to study the sample efficiency of grounded language learning". In: arXiv: 1810.08272.

Chevalier-Boisvert, Maxime, Bolun Dai, Mark Towers, Rodrigo de Lazcano, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry (2023). "Minigrid & Miniworld: Modular & Customizable Reinforcement Learning Environments for Goal-Oriented Tasks". In: *CoRR* abs/2306.13831.

Choi, Edward, Angeliki Lazaridou, and Nando de Freitas (2018). "Multi-Agent Compositional Communication Learning from Raw Visual Input". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=rknt2Be0-.

Chowdhury, Aritra, James R. Kubricht, Anup Sood, Peter Tu, and Alberto Santamaria-Pang (2020a). "ESCELL: Emergent Symbolic Cellular Language". In: DOI: 10.1109/ISBI45749.2020.9098343. eprint: arXiv:2007.09469. URL: https://arxiv.org/abs/2007.09469.

Chowdhury, Aritra, Alberto Santamaria-Pang, James R. Kubricht, Jianwei Qiu, and Peter Tu (2020b). *Symbolic Semantic Segmentation and Interpretation of COVID-19 Lung Infections in Chest CT volumes based on Emergent Languages*. eprint: arXiv:2008.09866. URL: https://arxiv.org/abs/2008.09866.

Chowdhury, Aritra, Alberto Santamaría-Pang, James R. Kubricht, and Peter H. Tu (2021). "Emergent Symbolic Language Based Deep Medical Image Classification". In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 689–692.

Cogswell, Michael, Jiasen Lu, Rishabh Jain, Stefan Lee, Devi Parikh, and Dhruv Batra (2020). "Dialog without Dialog Data: Learning Visual Dialog Agents from VQA Data". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 19988–19999. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/e7023ba77a45f7e84c5ee8a28dd63585-Paper.pdf.

Cogswell, Michael, Jiasen Lu, Stefan Lee, Devi Parikh, and Dhruv Batra (2019). *Emergence of Compositional Language with Deep Generational Transmission*. eprint: arXiv:1904.09067. URL: https://arxiv.org/abs/1904.09067.

Cope, Dylan and Peter McBurney (2022). "Joining the Conversation: Towards Language Acquisition for Ad Hoc Team Play". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=SLqgf7ZCQbq.

Cope, Dylan and Nandi Schoots (2021). "Learning to Communicate with Strangers via Channel Randomisation Methods". In: eprint: arXiv:2104.09557. URL: https://arxiv.org/abs/2104.09557.

Cowen-Rivers, Alexander I. and Jason Naradowsky (2020). *Emergent Communication with World Models*. eprint: arXiv:2002.09604. URL: https://arxiv.org/abs/2002.09604.

Creutz, Mathias and Krista Lagus (July 2002). "Unsupervised Discovery of Morphemes". In: *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*. Association for Computational Linguistics, pp. 21–30. DOI: 10.3115/1118647.1118650. URL: https://aclanthology.org/W02-0603/.

Dagan, Gautier, Dieuwke Hupkes, and Elia Bruni (2020). *Co-evolution of language and agents in referential games*. eprint: arXiv:2001.03361. URL: https://arxiv.org/abs/2001.03361.

Dekker, Peter and Bart De Boer (Dec. 2020). "Neural Agent-based Models To Study Language Contact Using Linguistic Data". English. In: 4th NeurIPS Workshop on Emergent Communication : Talking to Strangers: Zero-Shot Emergent Communication ; Conference date: 12-12-2020 Through 12-12-2020, pp. 1–6. URL: https://sites.google.com/view/emecom2020/home.

Denamganaï, Kevin, Sondess Missaoui, and James Alfred Walker (2023). *Visual Referential Games Further the Emergence of Disentangled Representations*. eprint: arXiv:2304.14511. URL: https://arxiv.org/abs/2304.14511.

Denamganaï, Kevin and James Alfred Walker (2020a). *On (Emergent) Systematic Generalisation and Compositionality in Visual Referential Games with Straight-Through*

Gumbel-Softmax Estimator. eprint: arXiv:2012.10776. URL: https://arxiv.org/abs/2012.10776.

Denamganaï, Kevin and James Alfred Walker (2020b). *ReferentialGym: A Nomenclature and Framework for Language Emergence & Grounding in (Visual) Referential Games*. eprint: arXiv:2012.09486. URL: https://arxiv.org/abs/2012.09486.

Dessì, Roberto, Diane Bouchacourt, Davide Crepaldi, and Marco Baroni (2019). *Focus on What's Informative and Ignore What's not: Communication Strategies in a Referential Game*. eprint: arXiv:1911.01892. URL: https://arxiv.org/abs/1911.01892.

Dessì, Roberto, Eugene Kharitonov, and Marco Baroni (2021). *Interpretable agent communication from scratch (with a generic visual processor emerging on the side)*. eprint: arXiv:2106.04258. URL: https://arxiv.org/abs/2106.04258.

Downey, C.m., Xuhui Zhou, Zeyu Liu, and Shane Steinert-Threlkeld (Dec. 2023). "Learning to translate by learning to communicate". In: *Proceedings of the 3rd Workshop on Multilingual Representation Learning (MRL)*. Ed. by Duygu Ataman. Singapore: Association for Computational Linguistics, pp. 218–238. URL: https://aclanthology.org/2023.mrl-1.17.

Dryer, Matthew S. and Martin Haspelmath, eds. (2013). *WALS Online*. Leipzig: Max Planck Institute for Evolutionary Anthropology. URL: https://wals.info/.

Eccles, Tom, Yoram Bachrach, Guy Lever, Angeliki Lazaridou, and Thore Graepel (2019). "Biases for Emergent Communication in Multi-agent Reinforcement Learning". In: *NeurIPS*.

Eecke, Paul Van, Katrien Beuls, Jérôme Botoko Ekila, and Roxana Rădulescu (2023). "Language games meet multi-agent reinforcement learning: A case study for the naming game". In: *Journal of Language Evolution*.

Eloff, Kevin, Arnu Pretorius, Okko Räsänen, Herman A. Engelbrecht, and Herman Kamper (2021). *Towards Learning to Speak and Hear Through Multi-Agent Communication over a Continuous Acoustic Channel*. eprint: arXiv:2111.02827. URL: https://arxiv.org/abs/2111.02827.

Evtimova, Katrina, Andrew Drozdov, Douwe Kiela, and Kyunghyun Cho (2017). *Emergent Communication in a Multi-Modal, Multi-Step Referential Game*. eprint: arXiv:1705.10369. URL: https://arxiv.org/abs/1705.10369.

Feng, Yicheng, Boshi An, and Zongqing Lu (2023). *Learning Multi-Object Positional Relationships via Emergent Communication*. eprint: arXiv:2302.08084. URL: https://arxiv.org/abs/2302.08084.

Fitzgerald, Nicole (2019). "To Populate Is To Regulate". In: *Emergent Communication Workshop at NeurIPS 2019*. URL: https://drive.google.com/file/d/1mINOlJvyWxGtqW_qNadMEhWNNJ4ZPIIa/view.

Flemming, Edward S (2013). *Auditory representations in phonology*. Routledge.

Foerster, Jakob N., Yannis M. Assael, Nando de Freitas, and Shimon Whiteson (2016). "Learning to communicate with Deep multi-agent reinforcement learning". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., pp. 2145–2153. ISBN: 9781510838819.

Foguelman, Daniel, Esteban Lanzarotti, E. Ferreyra, and Rodrigo Castro (2021). *Simulation of emergence in artificial societies: a practical model-based approach with the EB-DEVS formalism*. arXiv: 2110.08170.

Forcada, Mikel L, Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez, and Francis M Tyers (2011). "Apertium: a free/open-source platform for rule-based machine translation". In: *Machine translation* 25.2, pp. 127–144.

Fulker, Zachary, Patrick Forber, Rory Smead, and Christoph Riedl (2022). *Spontaneous emergence of groups and signaling diversity in dynamic networks*. eprint: arXiv:2210.17309. URL: https://arxiv.org/abs/2210.17309.

Gage, Philip (Feb. 1994). "A new algorithm for data compression". In: *C Users J.* 12.2, pp. 23–38. ISSN: 0898-9788.

Galke, Lukas, Yoav Ram, and Limor Raviv (2022). *Emergent Communication for Understanding Human Language Evolution: What's Missing?* eprint: arXiv:2204.10590. URL: https://arxiv.org/abs/2204.10590.

— (Dec. 2024). "Deep neural networks and humans both benefit from compositional language structure". In: *Nature Communications* 15.1. ISSN: 2041-1723. DOI: 10.1038/s41467-024-55158-1. URL: http://dx.doi.org/10.1038/s41467-024-55158-1.

Garcia, Washington, H S Clouse, and Kevin Butler (2022). "Disentangling Categorization in Multi-agent Emergent Communication". In: *NAACL*.

Gaya, Miquel Cornudella (2017). "Autotelic Principle: the role of intrinsic motivation in the emergence and development of artificial language." PhD thesis. Université de recherche Paris Sciences et Lettres. URL: https://hal.science/tel-01765234v1.

Gibson, Edward, Richard Futrell, Julian Jara-Ettinger, Kyle Mahowald, Leon Bergen, Sivalogeswaran Ratnasingam, Mitchell Gibson, Steven T. Piantadosi, and Bevil R. Conway (2017). "Color naming across languages reflects color use". In: *Proceedings of the National Academy of Sciences* 114.40, pp. 10785–10790. DOI: 10.1073/pnas.1619666114. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1619666114. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1619666114.

Gilberti, Miles, Shane Storks, and Huteng Dai (2025). *Discovering Properties of Inflectional Morphology in Neural Emergent Communication*. arXiv: 2508.05843 [cs.CL]. URL: https://arxiv.org/abs/2508.05843.

Graesser, Laura, Kyunghyun Cho, and Douwe Kiela (2019). *Emergent Linguistic Phenomena in Multi-Agent Communication Games*. eprint: arXiv:1901.08706. URL: https://arxiv.org/abs/1901.08706.

Grossi, Gina and Brian J. Ross (2017). "Evolved communication strategies and emergent behaviour of multi-agents in pursuit domains". In: *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, pp. 110–117.

Grupen, Niko A., Daniel D. Lee, and Bart Selman (2020). *Low-Bandwidth Communication Emerges Naturally in Multi-Agent Learning Systems*. eprint: arXiv:2011.14890. URL: https://arxiv.org/abs/2011.14890.

— (2021). *Curriculum-Driven Multi-Agent Learning and the Role of Implicit Communication in Teamwork*. arXiv: 2106.11156.

— (2022). "Multi-Agent Curricula and Emergent Implicit Signaling". In: *AAMAS*.

Guo, Biyang, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu (2023). *How Close is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection*. arXiv: 2301.07597 [cs.CL]. URL: https://arxiv.org/abs/2301.07597.

Guo, Shangmin (2019). *Emergence of Numeric Concepts in Multi-Agent Autonomous Communication*. eprint: arXiv:1911.01098. URL: https://arxiv.org/abs/1911.01098.

Guo, Shangmin, Yi Ren, Serhii Havrylov, Stella Frank, Ivan Titov, and Kenny Smith (2019). *The Emergence of Compositional Languages for Numeric Concepts Through Iterated Learning in Neural Agents*. eprint: arXiv:1910.05291. URL: https://arxiv.org/abs/1910.05291.

Guo, Shangmin, Yi Ren, Kory Mathewson, Simon Kirby, Stefano V. Albrecht, and Kenny Smith (2021). "Expressivity of Emergent Language is a Trade-off between Contextual Complexity and Unpredictability". In: eprint: arXiv:2106.03982. URL: https://arxiv.org/abs/2106.03982.

Guo, Shangmin, Yi Ren, Agnieszka Słowik, and Kory Mathewson (2020). *Inductive Bias and Language Expressivity in Emergent Communication*. eprint: arXiv:2012.02875. URL: https://arxiv.org/abs/2012.02875.

Guo, Yuxuan, Yifan Hao, Rui Zhang, Enshuai Zhou, Zidong Du, Xishan Zhang, Xinkai Song, Yuanbo Wen, Yongwei Zhao, Xuehai Zhou, Jiaming Guo, Qi Yi, Shaohui Peng, Di Huang, Ruizhi Chen, Qi Guo, and Yunji Chen (2023). "Emergent Communication for Rules Reasoning". In: *Thirty-seventh Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=gx20B4ItIw.

Gupta, Abhinav, Cinjon Resnick, Jakob Foerster, Andrew Dai, and Kyunghyun Cho (July 2020). "Compositionality and Capacity in Emergent Languages". In: *Proceedings of the 5th Workshop on Representation Learning for NLP*. Ed. by Spandana Gella, Johannes Welbl, Marek Rei, Fabio Petroni, Patrick Lewis, Emma Strubell, Minjoon Seo, and Hannaneh Hajishirzi. Online: Association for Computational Linguistics, pp. 34–38. DOI: 10.18653/v1/2020.repl4nlp-1.5. URL: https://aclanthology.org/2020.repl4nlp-1.5.

Gupta, Shubham and Ambedkar Dukkipati (2019). *On Voting Strategies and Emergent Communication*. arXiv: 1902.06897.

Gupta, Shubham, Rishi Hazra, and Ambedkar Dukkipati (2020). "Networked Multi-Agent Reinforcement Learning with Emergent Communication". In: *AAMAS*.

Ha, David and Jürgen Schmidhuber (2018). "World Models". In: DOI: 10.5281/ZENODO.1207631. URL: https://zenodo.org/record/1207631.

Hagiwara, Yoshinobu, Kazuma Furukawa, Akira Taniguchi, and Tadahiro Taniguchi (2021). *Multiagent Multimodal Categorization for Symbol Emergence: Emergent Communication via Interpersonal Cross-modal Inference*. eprint: arXiv:2109.07194. URL: https://arxiv.org/abs/2109.07194.

Hagiwara, Yoshinobu, Hiroyoshi Kobayashi, Akira Taniguchi, and Tadahiro Taniguchi (2019). "Symbol Emergence as an Interpersonal Multimodal Categorization". In: *Frontiers in Robotics and AI* 6.

Harris, Zellig S. (1955). "From Phoneme to Morpheme". In: *Language* 31.2, pp. 190–222. ISSN: 00978507, 15350665. URL: http://www.jstor.org/stable/411036 (visited on 11/29/2024).

Havrylov, Serhii and Ivan Titov (2017). "Emergence of Language with Multi-agent Games: Learning to Communicate with Sequences of Symbols". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., pp. 2149–2159. URL:

https://proceedings.neurips.cc/paper/2017/file/70222949cc0db89ab32c9969754d4758-Paper.pdf.

Hazra, Rishi, Sonu Dixit, and Sayambhu Sen (2020). *Infinite use of finite means: Zero-Shot Generalization using Compositional Emergent Protocols*. eprint: arXiv:2012.05011. URL: https://arxiv.org/abs/2012.05011.

— (2021). *Zero-Shot Generalization using Intrinsically Motivated Compositional Emergent Protocols*. eprint: arXiv:2105.05069. URL: https://arxiv.org/abs/2105.05069.

Herrmann, Daniel Alexander and Jacob VanDrunen (2022). "Sifting the Signal from the Noise". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=rcUeWQbRQb5.

Heusel, Martin, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter (2017). "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf.

IBM (n.d.). *Deep Blue*. https://www.ibm.com/history/deep-blue. Accessed: 2026-02-07.

Ikram, K., E. Mondragon, E. Alonso, and M. Garcia-Ortiz (June 2021). "HexaJungle: a MARL Simulator to Study the Emergence of Language". URL: https://openaccess.city.ac.uk/id/eprint/26284/.

Jang, Eric, Shixiang Gu, and Ben Poole (2017). "Categorical Reparameterization with Gumbel-Softmax". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=rkE3y85ee.

Kågebäck, Mikael (2018). "Word Representations for Emergent Communication and Natural Language Processing". PhD thesis. Chalmers University of Technology. URL: https://research.chalmers.se/en/publication/506086.

Kågebäck, Mikael, Devdatt P. Dubhashi, and Asad B. Sayeed (2018). "DeepColor: Reinforcement Learning optimizes information efficiency and well-formedness in color name partitioning". In: *Cognitive Science*.

Kajić, Ivana, Eser Aygün, and Doina Precup (2020). *Learning to cooperate: Emergent communication in multi-agent navigation*. eprint: arXiv:2004.01097. URL: https://arxiv.org/abs/2004.01097.

Kalinowska, Aleksandra, Elnaz Davoodi, Florian Strub, Kory Mathewson, Todd Murphey, and Patrick Pilarski (2022a). "Situated Communication: A Solution to Over-communication between Artificial Agents". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=HLqzzQWA7Z9.

Kalinowska, Aleksandra, Elnaz Davoodi, Florian Strub, Kory W Mathewson, Ivana Kajic, Michael Bowling, Todd D Murphey, and Patrick M Pilarski (2022b). *Over-communicate no more: Situated RL agents learn concise communication protocols*. eprint: arXiv:2211.01480. URL: https://arxiv.org/abs/2211.01480.

Kang, Yipeng, Tonghan Wang, and Gerard de Melo (2020). *Incorporating Pragmatic Reasoning Communication into Emergent Language*. eprint: arXiv:2006.04109. URL: https://arxiv.org/abs/2006.04109.

Karch, Tristan, Yoann Lemesle, Romain Laroche, Clément Moulin-Frier, and Pierre-Yves Oudeyer (2022). *Contrastive Multimodal Learning for Emergence of Graphical Sensory-*

Motor Communication. eprint: arXiv:2210.06468. URL: https://arxiv.org/abs/2210.06468.

Karten, Seth, Siddharth Agrawal, Mycal Tucker, Dana Hughes, Michael Lewis, Julie Shah, and Katia Sycara (2022a). *The Enforcers: Consistent Sparse-Discrete Methods for Constraining Informative Emergent Communication*. eprint: arXiv:2201.07452. URL: https://arxiv.org/abs/2201.07452.

Karten, Seth, Siva Kailas, Huao Li, and Katia Sycara (2023). *On the Role of Emergent Communication for Social Learning in Multi-Agent Reinforcement Learning*. eprint: arXiv:2302.14276. URL: https://arxiv.org/abs/2302.14276.

Karten, Seth and K. Sycara (2022b). *Intent-Grounded Compositional Communication through Mutual Information in Multi-Agent Teams*. in Decision Making in Multi-Agent Systems Workshop at IROS 2022. URL: https://dcslgatech.github.io/iros22-multi-agent-workshop/contributed_papers/IROS22-DMMAS_paper_5730.pdf.

Karten, Seth, Mycal Tucker, Huao Li, Siva Kailas, Michael Lewis, and K. Sycara (2022c). "Interpretable Learned Emergent Communication for Human-Agent Teams". In: *IEEE Transactions on Cognitive and Developmental Systems*.

Kegl, Judy, Ann Senghas, et al. (1999). "Creation through contact: Sign language emergence and sign language change in Nicaragua". In: *Language creation and language change: Creolization, diachrony, and development*, pp. 179–237.

Keresztury, Bence and Elia Bruni (2020). *Compositional properties of emergent languages in deep learning*. eprint: arXiv:2001.08618. URL: https://arxiv.org/abs/2001.08618.

Kharitonov, Eugene and Marco Baroni (Nov. 2020a). "Emergent Language Generalization and Acquisition Speed are not tied to Compositionality". In: *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. Ed. by Afra Alishahi, Yonatan Belinkov, Grzegorz Chrupała, Dieuwke Hupkes, Yuval Pinter, and Hassan Sajjad. Online: Association for Computational Linguistics, pp. 11–15. DOI: 10.18653/v1/2020.blackboxnlp-1.2. URL: https://aclanthology.org/2020.blackboxnlp-1.2/.

Kharitonov, Eugene, Rahma Chaabouni, Diane Bouchacourt, and Marco Baroni (Nov. 2019). "EGG: a toolkit for research on Emergence of lanGuage in Games". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*. Ed. by Sebastian Padó and Ruihong Huang. Hong Kong, China: Association for Computational Linguistics, pp. 55–60. DOI: 10.18653/v1/D19-3010. URL: https://aclanthology.org/D19-3010.

— (13–18 Jul 2020b). "Entropy Minimization In Emergent Languages". In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 5220–5230. URL: https://proceedings.mlr.press/v119/kharitonov20a.html.

Khazar Khorrami, Okko Räsänen (2019). *Can phones, syllables, and words emerge as side-products of cross-situational audiovisual learning? – A computational investigation*. eprint: arXiv:2109.14200. URL: https://arxiv.org/abs/2109.14200.

Khomtchouk, Bohdan and Shyam Sudhakaran (2018). *Modeling natural language emergence with integral transform theory and reinforcement learning*. eprint: arXiv:1812.01431. URL: https://arxiv.org/abs/1812.01431.

Kim, Jooyeon and Alice Oh (2021). "Emergent Communication under Varying Sizes and Connectivities". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., pp. 17579–17591. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/92dfa194391a59dc65b88b704599dbd6-Paper.pdf.

Kingma, Diederik P. and Jimmy Ba (2015). "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: http://arxiv.org/abs/1412.6980.

Kirby, Simon (2000). "Syntax Without Natural Selection: How Compositionality Emerges from Vocabulary in a Population of Learners". In: *The Evolutionary Emergence of Language: Social Function and the Origins of Linguistic Form*. Ed. by Chris Knight, Michael Studdert-Kennedy, and JamesEditors Hurford. Cambridge University Press, pp. 303–323. DOI: 10.1017/CBO9780511606441.019.

— (Apr. 2002). "Natural Language From Artificial Life". In: *Artificial Life* 8.2, pp. 185–215. ISSN: 1064-5462. DOI: 10.1162/106454602320184248. eprint: https://direct.mit.edu/artl/article-pdf/8/2/185/1661892/106454602320184248.pdf. URL: https://doi.org/10.1162/106454602320184248.

Kirby, Simon, Hannah Cornish, and Kenny Smith (2008). "Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language". In: *Proceedings of the National Academy of Sciences* 105, pp. 10681–10686.

Kirby, Simon, Mike Dowman, and Thomas L. Griffiths (2007). "Innateness and culture in the evolution of language". In: *Proceedings of the National Academy of Sciences* 104.12, pp. 5241–5245. DOI: 10.1073/pnas.0608222104. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.0608222104. URL: https://www.pnas.org/doi/abs/10.1073/pnas.0608222104.

Kirby, Simon, Tom Griffiths, and Kenny Smith (2014). "Iterated learning and the evolution of language". In: *Current Opinion in Neurobiology* 28. SI: Communication and language, pp. 108–114. ISSN: 0959-4388. DOI: https://doi.org/10.1016/j.conb.2014.07.014. URL: https://www.sciencedirect.com/science/article/pii/S0959438814001421.

Kirby, Simon and James Hurford (1997). "Learning, culture and evolution in the origin of linguistic constraints". In: *Fourth European conference on artificial life*. Cambridge (MS): MIT Press, pp. 493–502.

Kirby, Simon and James R. Hurford (2002). "The Emergence of Linguistic Structure: An Overview of the Iterated Learning Model". In: *Simulating the Evolution of Language*. Ed. by Angelo Cangelosi and Domenico Parisi. London: Springer London, pp. 121–147. ISBN: 978-1-4471-0663-0. DOI: 10.1007/978-1-4471-0663-0_6. URL: https://doi.org/10.1007/978-1-4471-0663-0_6.

Kolb, Benjamin, Leon Lang, Henning Bartsch, Arwin Gansekoele, Raymond Koopmanschap, Leonardo Romor, David Speck, Mathijs Mul, and Elia Bruni (2019). *Learning to Request Guidance in Emergent Communication*. eprint: arXiv:1912.05525. URL: https://arxiv.org/abs/1912.05525.

Korbak, Tomasz, Julian Zubek, Łukasz Kuciński, Piotr Milos, and Joanna Rączaszek-Leonardi (2021). "Interaction history as a source of compositionality in emergent communication". In: *Interaction Studies*.

Korbak, Tomasz, Julian Zubek, Łukasz Kuciński, Piotr Miłoś, and Joanna Rączaszek-Leonardi (2019). *Developmentally motivated emergence of compositional communication via template transfer*. eprint: arXiv:1910.06079. URL: https://arxiv.org/abs/1910.06079.

Korbak, Tomasz, Julian Zubek, and Joanna Rączaszek-Leonardi (2020). *Measuring non-trivial compositionality in emergent communication*. arXiv: 2010.15058 [cs.NE]. URL: https://arxiv.org/abs/2010.15058.

Kottur, Satwik, José Moura, Stefan Lee, and Dhruv Batra (Sept. 2017). "Natural Language Does Not Emerge 'Naturally' in Multi-Agent Dialog". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Copenhagen, Denmark: Association for Computational Linguistics, pp. 2962–2967. DOI: 10.18653/v1/D17-1321. URL: https://aclanthology.org/D17-1321/.

Kriegeskorte, Nikolaus, Marieke Mur, and Peter Bandettini (2008). "Representational similarity analysis - connecting the branches of systems neuroscience". In: *Frontiers in Systems Neuroscience* 2. ISSN: 1662-5137. DOI: 10.3389/neuro.06.004.2008. URL: https://www.frontiersin.org/articles/10.3389/neuro.06.004.2008.

Kuciński, Łukasz, Paweł Kołodziej, and Piotr Miłoś (2020). "Emergence of compositional language in communication through noisy channel". In: *Language in Reinforcement Learning Workshop at ICML 2020*. URL: https://openreview.net/forum?id=ZbXlSL_xwtA.

Kuciński, Łukasz, Tomasz Korbak, Paweł Kołodziej, and Piotr Miłoś (2021). *Catalytic Role Of Noise And Necessity Of Inductive Biases In The Emergence Of Compositional Communication*. eprint: arXiv:2111.06464. URL: https://arxiv.org/abs/2111.06464.

Kudo, Taku (July 2018). "Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Iryna Gurevych and Yusuke Miyao. Melbourne, Australia: Association for Computational Linguistics, pp. 66–75. DOI: 10.18653/v1/P18-1007. URL: https://aclanthology.org/P18-1007/.

Kuhn, Thomas S. (1962). *The Structure of Scientific Revolutions*. Chicago, IL: University of Chicago Press.

Kuhnle, Alexander and Ann Copestake (2017). *ShapeWorld - A new test methodology for multimodal language understanding*. arXiv: 1704.04517 [cs.CL]. URL: https://arxiv.org/abs/1704.04517.

Kurimo, Mikko, Sami Virpioja, Ville Turunen, and Krista Lagus (July 2010). "Morpho Challenge 2005-2010: Evaluations and Results". In: *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*. Ed. by Jeffrey Heinz, Lynne Cahill, and Richard Wicentowski. Uppsala, Sweden: Association for Computational Linguistics, pp. 87–95. URL: https://aclanthology.org/W10-2211/.

LaCroix, Travis (2019). "Biology and Compositionality: Empirical Considerations for Emergent-Communication Protocols". In: *CoRR* abs/1911.11668. arXiv: 1911.11668. URL: http://arxiv.org/abs/1911.11668.

Lake, Brenden and Marco Baroni (Oct. 2018). "Generalization without Systematicity: On the Compositional Skills of Sequence-to-Sequence Recurrent Networks". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas

Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 2873–2882. URL: https://proceedings.mlr.press/v80/lake18a.html.

Lam, Siu Kwan, Antoine Pitrou, and Stanley Seibert (2015). "Numba: a LLVM-based Python JIT compiler". In: *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*. LLVM '15. Austin, Texas: Association for Computing Machinery. ISBN: 9781450340052. DOI: 10.1145/2833157.2833162. URL: https://doi.org/10.1145/2833157.2833162.

Lan, Nur Geffen, Emmanuel Chemla, and Shane Steinert-Threlkeld (2020). *On the Spontaneous Emergence of Discrete and Compositional Signals*. eprint: arXiv:2005.00110. URL: https://arxiv.org/abs/2005.00110.

Lannelongue, K., M. De Milly, Roberto Marcucci, S. Selevarangame, A. Supizet, and A. Grincourt (2019). "Compositional Grounded Language for Agent Communication in Reinforcement Learning Environment". In: *Journal of Autonomous Intelligence*.

Lazaridou, Angeliki and Marco Baroni (2020). *Emergent Multi-Agent Communication in the Deep Learning Era*. arXiv: 2006.02419 [cs.CL]. URL: https://arxiv.org/abs/2006.02419.

Lazaridou, Angeliki, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark (2018). *Emergence of Linguistic Communication from Referential Games with Symbolic and Pixel Input*. arXiv: 1804.03984 [cs.AI]. URL: https://arxiv.org/abs/1804.03984.

Lazaridou, Angeliki, Alexander Peysakhovich, and Marco Baroni (2016). *Multi-Agent Cooperation and the Emergence of (Natural) Language*. eprint: arXiv:1612.07182. URL: https://arxiv.org/abs/1612.07182.

Lee, Jason, Kyunghyun Cho, and Douwe Kiela (Nov. 2019). "Countering Language Drift via Visual Grounding". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, pp. 4385–4395. DOI: 10.18653/v1/D19-1447. URL: https://aclanthology.org/D19-1447.

Lee, Jason, Kyunghyun Cho, Jason Weston, and Douwe Kiela (2018). *Emergent Translation in Multi-Agent Communication*. arXiv: 1710.06922.

Leni, Juan, John Levine, and John Quigley (2018). *Seq2Seq Mimic Games: A Signaling Perspective*. arXiv: 1811.06564 [cs.AI]. URL: https://arxiv.org/abs/1811.06564.

Levy, Ido, Orr Paradise, Boaz Carmeli, Ron Meir, Shafi Goldwasser, and Yonatan Belinkov (2025). *Unsupervised Translation of Emergent Communication*. arXiv: 2502.07552 [cs.CL]. URL: https://arxiv.org/abs/2502.07552.

Lewis, David Kellogg (1969). *Convention: A Philosophical Study*. Cambridge, MA, USA: Wiley-Blackwell.

Li, Fushan and Michael Bowling (2019). "Ease-of-Teaching and Language Structure from Emergent Communication". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc.

Li, Quanyi, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou (2023). "MetaDrive: Composing Diverse Driving Scenarios for Generalizable Reinforcement Learning". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.3, pp. 3461–3475. DOI: 10.1109/TPAMI.2022.3190471.

Li, Sheng, Yutai Zhou, R. Allen, and Mykel J. Kochenderfer (2022). "Learning Emergent Discrete Message Communication for Cooperative Reinforcement Learning". In: *ICRA*.

Li, Yaoyiran, Edoardo Maria Ponti, Ivan Vulić, and Anna Korhonen (Dec. 2020). "Emergent Communication Pretraining for Few-Shot Machine Translation". In: *Proceedings of the 28th International Conference on Computational Linguistics*. Ed. by Donia Scott, Nuria Bel, and Chengqing Zong. Barcelona, Spain (Online): International Committee on Computational Linguistics, pp. 4716–4731. DOI: 10.18653/v1/2020.coling-main.416. URL: https://aclanthology.org/2020.coling-main.416.

Li, Yuxi (2018). *Deep Reinforcement Learning: An Overview*. arXiv: 1701.07274 [cs.LG]. URL: https://arxiv.org/abs/1701.07274.

Liang, Paul Pu, Jeffrey Chen, Ruslan Salakhutdinov, Louis-Philippe Morency, and Satwik Kottur (2020). "On Emergent Communication in Competitive Multi-Agent Teams". In: *AAMAS*.

Lin, Toru, Minyoung Huh, Chris Stauffer, Ser-Nam Lim, and Phillip Isola (2021). *Learning to Ground Multi-Agent Communication with Autoencoders*. eprint: arXiv:2110.15349. URL: https://arxiv.org/abs/2110.15349.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár (2015). *Microsoft COCO: Common Objects in Context*. aaXiv: 1405.0312 (cs.CV). URL: https://arxiv.org/abs/1405.0312.

Lindblom, B. (1990). "Explaining Phonetic Variation: A Sketch of the H&H Theory". In: *Speech Production and Speech Modelling*. Ed. by William J. Hardcastle and Alain Marchal. Dordrecht: Springer Netherlands, pp. 403–439. ISBN: 978-94-009-2037-8. DOI: 10.1007/978-94-009-2037-8_16. URL: https://doi.org/10.1007/978-94-009-2037-8_16.

Lipinski, Olaf, Adam Sobey, Federico Cerutti, and Timothy J Norman (2022). "Emergent Password Signalling in the Game of Werewolf". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=B4xM-Qb0mbq.

— (2024). "Speaking Your Language: Spatial Relationships in Interpretable Emergent Communication". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=vIP8IWmZlN.

Lipowska, Dorota and Adam Lipowski (2018). "Emergence of linguistic conventions in multi-agent reinforcement learning". In: *PLoS ONE* 13.

Lo, Yat Long and Biswa Sengupta (2022). "Learning To Ground Decentralized Multi-Agent Communication with Contrastive Learning". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=rLceWXWCmZc.

Lowe, Ryan, Jakob N. Foerster, Y-Lan Boureau, Joelle Pineau, and Yann Dauphin (2019). "On the Pitfalls of Measuring Emergent Communication". In: *AAMAS*.

Lowe, Ryan, Abhinav Gupta, Jakob Foerster, Douwe Kiela, and Joelle Pineau (2020). *On the interaction between supervision and self-play in emergent communication*. eprint: arXiv:2002.01093. URL: https://arxiv.org/abs/2002.01093.

Lu, Yuchen, Soumye Singhal, Florian Strub, Olivier Pietquin, and Aaron Courville (2020). "Countering Language Drift with Seeded Iterated Learning". In: *Proceedings of the 37th International Conference on Machine Learning*. ICML'20. JMLR.org.

MacLennan, Bruce J. and Gordon M. Burghardt (1993). "Synthetic Ethology and the Evolution of Cooperative Communication". In: *Adaptive Behavior* 2.2, pp. 161–188. DOI:

10.1177/105971239300200203. eprint: https://doi.org/10.1177/105971239300200203. URL: https://doi.org/10.1177/105971239300200203.

Maddison, Chris J., Andriy Mnih, and Yee Whye Teh (2017). "The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables". In: *Proceedings of the 2017 International Conference on Learning Representations (ICLR)*. URL: https://openreview.net/forum?id=S1jE5L5gl.

Mahaut, Matéo, Francesca Franzon, Roberto Dessì, and Marco Baroni (2023). *Referential communication in heterogeneous communities of pre-trained visual deep networks*. eprint: arXiv:2302.08913. URL: https://arxiv.org/abs/2302.08913.

Mahowald, Kyle, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko (2023). "Dissociating language and thought in large language models: a cognitive perspective". In: arXiv: 2301.06627.

Mandelbrot, Benoit et al. (1953). "An informational theory of the statistical structure of language". In: *Communication theory* 84, pp. 486–502.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz (1993). "Building a Large Annotated Corpus of English: The Penn Treebank". In: *Computational Linguistics* 19.2. Ed. by Julia Hirschberg, pp. 313–330. URL: https://aclanthology.org/J93-2004/.

Masquil, Elías, Gautier Hamon, Eleni Nisioti, and Clément Moulin-Frier (2022). *Intrinsically-Motivated Goal-Conditioned Reinforcement Learning in Multi-Agent Environments*. eprint: arXiv:2211.06082. URL: https://arxiv.org/abs/2211.06082.

Mihai, Daniela and Jonathon Hare (2019). *Avoiding hashing and encouraging visual semantics in referential emergent language games*. eprint: arXiv:1911.05546. URL: https://arxiv.org/abs/1911.05546.

— (2021a). *Learning to Draw: Emergent Communication through Sketching*. arXiv: 2106.02067 [cs.CV].

— (2021b). *The emergence of visual semantics through communication games*. eprint: arXiv:2101.10253. URL: https://arxiv.org/abs/2101.10253.

Mirzaee, Roshanak and Parisa Kordjamshidi (Dec. 2022). "Transfer Learning with Synthetic Corpora for Spatial Role Labeling and Reasoning". In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 6148–6165. DOI: 10.18653/v1/2022.emnlp-main.413. URL: https://aclanthology.org/2022.emnlp-main.413.

Misra, Kanishka, Allyson Ettinger, and Julia Taylor Rayz (2021). *Do language models learn typicality judgments from text?* arXiv: 2105.02987.

Mordatch, Igor and Pieter Abbeel (2018). "Emergence of grounded compositional language in multi-agent populations". In: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*. AAAI'18/IAAI'18/EAAI'18. New Orleans, Louisiana, USA: AAAI Press. ISBN: 978-1-57735-800-8.

Moulin-Frier, Clément and Pierre-Yves Oudeyer (2020). "Multi-Agent Reinforcement Learning as a Computational Tool for Language Evolution Research: Historical Context and Future Challenges". In: eprint: arXiv:2002.08878. URL: https://arxiv.org/abs/2002.08878.

Mu, Jesse and Noah Goodman (2021). "Emergent Communication of Generalizations". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., pp. 17994–18007. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/9597353e41e6957b5e7aa79214fcb256-Paper.pdf.

Mu, Yao, Shunyu Yao, Mingyu Ding, Ping Luo, and Chuang Gan (2023). "EC2: Emergent Communication for Embodied Control". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6704–6714. DOI: 10.1109/CVPR52729.2023.00648.

Mul, Mathijs, Diane Bouchacourt, and Elia Bruni (2019). *Mastering emergent language: learning to guide in simulated navigation*. eprint: arXiv:1908.05135. URL: https://arxiv.org/abs/1908.05135.

Naik, Aakanksha (2022). "Adapting to the Long Tail in Language Understanding". PhD thesis. Carnegie Mellon University. URL: https://www.lti.cs.cmu.edu/sites/default/files/naik%2C%20aakanksha%20-%20Thesis_1.pdf.

Narasimhan, Karthik, Regina Barzilay, and Tommi Jaakkola (2015). "An Unsupervised Method for Uncovering Morphological Chains". In: *Transactions of the Association for Computational Linguistics* 3. Ed. by Michael Collins and Lillian Lee, pp. 157–167. DOI: 10.1162/tacl_a_00130. URL: https://aclanthology.org/Q15-1012/.

Narayan-Chen, Anjali, Prashant Jayannavar, and Julia Hockenmaier (July 2019). "Collaborative Dialogue in Minecraft". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 5405–5415. DOI: 10.18653/v1/P19-1537. URL: https://aclanthology.org/P19-1537.

Nevens, Jens, Paul Van Eecke, and Katrien Beuls (2020). *A Practical Guide to Studying Emergent Communication through Grounded Language Games*. arXiv: 2004.09218.

Noukhovitch, Michael, Travis LaCroix, Angeliki Lazaridou, and Aaron C. Courville (2021). "Emergent Communication under Competition". In: *AAMAS*.

Ohmer, Xenia, Marko Duda, and Elia Bruni (2022). *Emergence of hierarchical reference systems in multi-agent communication*. eprint: arXiv:2203.13176. URL: https://arxiv.org/abs/2203.13176.

Ohmer, Xenia, Michael Marino, Michael Franke, and Peter König (2021a). *Mutual influence between language and perception in multi-agent communication games*. eprint: arXiv:2112.14518. URL: https://arxiv.org/abs/2112.14518.

Ohmer, Xenia, Michael Marino, Peter Koenig, and Michael Franke (2021b). "Why and how to study the impact of perception on language emergence in artificial agents". In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. Vol. 43. URL: https://escholarship.org/uc/item/6p82v6st.

Olah, Chris, Alexander Mordvintsev, and Ludwig Schubert (2017). "Feature Visualization". In: *Distill*. https://distill.pub/2017/feature-visualization. DOI: 10.23915/distill.00007.

OpenAI (2023). *GPT-4 Technical Report*. arXiv: 2303.08774 [cs.CL]. URL: https://arxiv.org/abs/2303.08774.

OpenAI, : Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub Pachocki, Michael Petrov, Henrique

P. d. O. Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang (2019). *Dota 2 with Large Scale Deep Reinforcement Learning*. arXiv: 1912.06680 [cs.LG]. URL: https://arxiv.org/abs/1912.06680.

Ossenkopf, Marie (2019). "Enhancing Communication Learning through Empathic Prediction". In: *Emergent Communication Workshop at NeurIPS 2019*. URL: https://drive.google.com/file/d/19VWGANQUBWBBqEgXi1zbRgQ9uv0zh8jJ/view.

Ossenkopf, Marie, Kevin Sebastian Luck, and Kory Wallace Mathewson (2022). "Which Language Evolves Between Heterogeneous Agents? - Communicating Movement Instructions With Widely Different Time Scopes". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=BnfgM7-0mW5.

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe (2022). *Training language models to follow instructions with human feedback*. arXiv: 2203.02155 [cs.CL]. URL: https://arxiv.org/abs/2203.02155.

Papadimitriou, Isabel and Dan Jurafsky (Nov. 2020). "Learning Music Helps You Read: Using Transfer to Study Linguistic Structure in Language Models". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu. Online: Association for Computational Linguistics, pp. 6829–6839. DOI: 10.18653/v1/2020.emnlp-main.554. URL: https://aclanthology.org/2020.emnlp-main.554/.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu (July 2002). "Bleu: a Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Ed. by Pierre Isabelle, Eugene Charniak, and Dekang Lin. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040.

Partee, Barbara et al. (1984). "Compositionality". In: *Varieties of formal semantics* 3, pp. 281–311.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Patel, Shivansh, Saim Wani, Unnat Jain, Alexander G. Schwing, Svetlana Lazebnik, Manolis Savva, and Angel Xuan Chang (2021). "Interpretation of Emergent Communication in Heterogeneous Collaborative Embodied Agents". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15993–15943.

Perkins, Hugh (2021a). *Neural networks can understand compositional functions that humans do not, in the context of emergent communication*. eprint: arXiv:2103.04180. URL: https://arxiv.org/abs/2103.04180.

— (2021b). "TexRel: a Green Family of Datasets for Emergent Communications on Relations". In: arXiv: 2105.12804.

— (2022). *Icy: A benchmark for measuring compositional inductive bias of emergent communication models*. URL: https://openreview.net/forum?id=S352vriz3G.

Piantadosi, S.T. (2014). "Zipf's word frequency law in natural language: A critical review and future directions". In: *Psychon Bull Rev* 21, pp. 1112–1130. URL: https://doi.org/10.3758/s13423-014-0585-6.

Piazza, Nancirose and Vahid Behzadan (2023). *A Theory of Mind Approach as Test-Time Mitigation Against Emergent Adversarial Communication*. eprint: arXiv:2302.07176. URL: https://arxiv.org/abs/2302.07176.

Pollard, Carl and Ivan A Sag (1994). *Head-driven phrase structure grammar*. University of Chicago Press.

Popović, Maja (Sept. 2015). "chrF: character n-gram F-score for automatic MT evaluation". In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal: Association for Computational Linguistics, pp. 392–395. DOI: 10.18653/v1/W15-3049. URL: https://aclanthology.org/W15-3049.

Portelance, Eva, Michael C. Frank, Dan Jurafsky, Alessandro Sordoni, and Romain Laroche (Nov. 2021). "The Emergence of the Shape Bias Results from Communicative Efficiency". In: *Proceedings of the 25th Conference on Computational Natural Language Learning*. Ed. by Arianna Bisazza and Omri Abend. Online: Association for Computational Linguistics, pp. 607–623. DOI: 10.18653/v1/2021.conll-1.48. URL: https://aclanthology.org/2021.conll-1.48/.

Post, Matt (Oct. 2018). "A Call for Clarity in Reporting BLEU Scores". In: *Proceedings of the Third Conference on Machine Translation: Research Papers*. Belgium, Brussels: Association for Computational Linguistics, pp. 186–191. URL: https://www.aclweb.org/anthology/W18-6319.

Qiu, Shuwen, Sirui Xie, Lifeng Fan, Tao Gao, Song-Chun Zhu, and Yixin Zhu (2021). *Emergent Graphical Conventions in a Visual Communication Game*. eprint: arXiv:2111.14210. URL: https://arxiv.org/abs/2111.14210.

Qiu, Zhuang, Xufeng Duan, and Zhenguang G Cai (May 2023). *Pragmatic Implicature Processing in ChatGPT*. DOI: 10.31234/osf.io/qtbh9. URL: osf.io/preprints/psyarxiv/qtbh9.

Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever (2019). "Language Models are Unsupervised Multitask Learners". In.

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang (Nov. 2016). "SQuAD: 100,000+ Questions for Machine Comprehension of Text". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Ed. by Jian Su, Kevin Duh, and Xavier Carreras. Austin, Texas: Association for Computational Linguistics, pp. 2383–2392. DOI: 10.18653/v1/D16-1264. URL: https://aclanthology.org/D16-1264.

Raviv, Limor, Marianne de Heer Kloots, and Antje Meyer (2022). "What makes a language easy to learn? A preregistered study on how systematic structure and community size affect language learnability". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=BdbexQ-0XW9.

Ren, Yi, Shangmin Guo, Matthieu Labeau, Shay B. Cohen, and Simon Kirby (2020). "Compositional Languages Emerge in a Neural Iterated Learning Model". In: eprint: arXiv:2002.01365. URL: https://arxiv.org/abs/2002.01365.

Resnick, Cinjon, Abhinav Gupta, Jakob N. Foerster, Andrew M. Dai, and Kyunghyun Cho (2020). "Capacity, Bandwidth, and Compositionality in Emergent Language Learning". In: *AAMAS*.

Rita, Mathieu, Rahma Chaabouni, and Emmanuel Dupoux (Nov. 2020). ""LazImpa": Lazy and Impatient neural agents learn to communicate efficiently". In: *Proceedings of the 24th Conference on Computational Natural Language Learning*. Ed. by Raquel Fernández and Tal Linzen. Online: Association for Computational Linguistics, pp. 335–343. DOI: 10.18653/v1/2020.conll-1.26. URL: https://aclanthology.org/2020.conll-1.26/.

Rita, Mathieu, Florian Strub, Jean-Bastien Grill, Olivier Pietquin, and Emmanuel Dupoux (2022a). "On the role of population heterogeneity in emergent communication". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=5Qkd7-bZfI.

Rita, Mathieu, Corentin Tallec, Paul Michel, Jean-Bastien Grill, Olivier Pietquin, Emmanuel Dupoux, and Florian Strub (2022b). "Emergent Communication: Generalization and Overfitting in Lewis Games". In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. URL: https://openreview.net/forum?id=qqHMvHbfu6.

Roberts, Gareth (2017). "The linguist's Drosophila: Experiments in language change". In: *Linguistics Vanguard* 2017, pp. 1–13.

Rodríguez Luna, Diana, Edoardo Maria Ponti, Dieuwke Hupkes, and Elia Bruni (Nov. 2020). "Internal and external pressures on language emergence: least effort, object constancy and frequency". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Ed. by Trevor Cohn, Yulan He, and Yang Liu. Online: Association for Computational Linguistics, pp. 4428–4437. DOI: 10.18653/v1/2020.findings-emnlp.397. URL: https://aclanthology.org/2020.findings-emnlp.397/.

Sabathiel, Silvester, Trygve Solstad, Alberto Testolin, and Flavio Petruzzellis (2022). "Self-Communicating Deep Reinforcement Learning Agents Develop External Number Representations". In: *Proceedings of the Northern Lights Deep Learning Workshop*.

Santamaria-Pang, Alberto, James Kubricht, Aritra Chowdhury, Chitresh Bhushan, and Peter Tu (2020). *Towards Emergent Language Symbolic Semantic Segmentation and Model Interpretability*. eprint: arXiv:2007.09448. URL: https://arxiv.org/abs/2007.09448.

Santamaría-Pang, Alberto, James R. Kubricht, Chinmaya Devaraj, Aritra Chowdhury, and Peter H. Tu (2019). "Towards Semantic Action Analysis via Emergent Language". In: *2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, pp. 224–2244.

Scholz, Barbara C., Francis Jeffry Pelletier, Geoffrey K. Pullum, and Ryan Nefdt (2024). "Philosophy of Linguistics". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta and Uri Nodelman. Spring 2024. Metaphysics Research Lab, Stanford University.

Schrimpf, Martin, Idan Blank, Greta Tuckute, Carina Kauf, Eghbal A. Hosseini, Nancy Kanwisher, Joshua Tenenbaum, and Evelina Fedorenko (2020). "Artificial Neural Networks

Accurately Predict Language Processing in the Brain". In: *bioRxiv*. DOI: 10.1101/2020.06.26.174482. eprint: https://www.biorxiv.org/content/early/2020/06/27/2020.06.26.174482.full.pdf. URL: https://www.biorxiv.org/content/early/2020/06/27/2020.06.26.174482.

Schützenberger, M.P. (1963). "On context-free languages and push-down automata". In: *Information and Control* 6.3, pp. 246–264. ISSN: 0019-9958. DOI: https://doi.org/10.1016/S0019-9958(63)90306-1. URL: https://www.sciencedirect.com/science/article/pii/S0019995863903061.

Sennrich, Rico, Barry Haddow, and Alexandra Birch (Aug. 2016). "Neural Machine Translation of Rare Words with Subword Units". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Katrin Erk and Noah A. Smith. Berlin, Germany: Association for Computational Linguistics, pp. 1715–1725. DOI: 10.18653/v1/P16-1162. URL: https://aclanthology.org/P16-1162/.

Sharma, Piyush, Nan Ding, Sebastian Goodman, and Radu Soricut (July 2018). "Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Iryna Gurevych and Yusuke Miyao. Melbourne, Australia: Association for Computational Linguistics, pp. 2556–2565. DOI: 10.18653/v1/P18-1238. URL: https://aclanthology.org/P18-1238.

Silver, David, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, L. Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis (2017). "Mastering the game of Go without human knowledge". In: *Nature* 550, pp. 354–359.

Sirota, Joshua, Vadim Bulitko, Matthew R. G. Brown, and Sergio Poo Hernandez (2019). "Evolving recurrent neural networks for emergent communication". In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*.

Słowik, Agnieszka, Abhinav Gupta, William L. Hamilton, Mateja Jamnik, and Sean B. Holden (2020a). *Towards Graph Representation Learning in Emergent Communication*. eprint: arXiv:2001.09063. URL: https://arxiv.org/abs/2001.09063.

Słowik, Agnieszka, Abhinav Gupta, William L. Hamilton, Mateja Jamnik, Sean B. Holden, and Christopher Pal (2020b). *Structural Inductive Biases in Emergent Communication*. eprint: arXiv:2002.01335. URL: https://arxiv.org/abs/2002.01335.

Smith, Kenny (2002). "The cultural evolution of communication in a population of neural networks". In: *Connection Science* 14.1, pp. 65–84. DOI: 10.1080/09540090210164306. eprint: https://doi.org/10.1080/09540090210164306. URL: https://doi.org/10.1080/09540090210164306.

Smith, Kenny, Simon Kirby, and Henry Brighton (Oct. 2003). "Iterated Learning: A Framework for the Emergence of Language". In: *Artificial Life* 9.4, pp. 371–386. ISSN: 1064-5462. DOI: 10.1162/106454603322694825.

Steels, Luc (1997). "The Synthetic Modeling of Language Origins". In: *Evolution of Communication* 1.1, pp. 1–34. ISSN: 1387-5337. DOI: https://doi.org/10.1075/eoc.1.1.02ste. URL: https://www.jbe-platform.com/content/journals/10.1075/eoc.1.1.02ste.

— (1999). "The talking heads experiment". In: Words and meanings Volume 1. DOI: 10.17169/FUDOCS_document_000000022455.

Steinert-Threlkeld, Shane (2019). *Paying Attention to Function Words*. eprint: arXiv:1909.11060. URL: https://arxiv.org/abs/1909.11060.

— (2020). "Toward the Emergence of Nontrivial Compositionality". In: *Philosophy of Science* 87, pp. 897–909.

Steinert-Threlkeld, Shane, Xuhui Zhou, Zeyu Liu, and C. M. Downey (2022). "Emergent Communication Fine-tuning (EC-FT) for Pretrained Language Models". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=SUqrM7WR7W5.

Suzgun, Mirac, Yonatan Belinkov, Stuart Shieber, and Sebastian Gehrmann (Aug. 2019). "LSTM Networks Can Perform Dynamic Counting". In: *Proceedings of the Workshop on Deep Learning and Formal Languages: Building Bridges*. Ed. by Jason Eisner, Matthias Gallé, Jeffrey Heinz, Ariadna Quattoni, and Guillaume Rabusseau. Florence: Association for Computational Linguistics, pp. 44–54. DOI: 10.18653/v1/W19-3905. URL: https://aclanthology.org/W19-3905.

Tanaka-Ishii, Kumiko (2021). "Articulation of Elements". In: *Statistical Universals of Language: Mathematical Chance vs. Human Choice*. Cham: Springer International Publishing, pp. 115–124. ISBN: 978-3-030-59377-3. DOI: 10.1007/978-3-030-59377-3_11. URL: https://doi.org/10.1007/978-3-030-59377-3_11.

Taniguchi, Tadahiro, Takayuki Nagai, Tomoaki Nakamura, Naoto Iwahashi, Tetsuya Ogata, and Hideki Asoh (2015). *Symbol Emergence in Robotics: A Survey*. eprint: arXiv:1509.08973. URL: https://arxiv.org/abs/1509.08973.

Taniguchi, Tadahiro, Yuto Yoshida, Akira Taniguchi, and Yoshinobu Hagiwara (2022). *Emergent Communication through Metropolis-Hastings Naming Game with Deep Generative Models*. eprint: arXiv:2205.12392. URL: https://arxiv.org/abs/2205.12392.

Thomas, Christo Kurisummoottil and Walid Saad (2022). *Neuro-Symbolic Causal Reasoning Meets Signaling Game for Emergent Semantic Communications*. eprint: arXiv:2210.12040. URL: https://arxiv.org/abs/2210.12040.

Todo, Kense and Masayuki Yamamura (2020). "Contribution of reflection in language emergence with an under-restricted situation". In: *2020 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, pp. 147–154.

Tucker, Mycal, Roger P. Levy, Julie Shah, and Noga Zaslavsky (2022a). "Generalization and Translatability in Emergent Communication via Informational Constraints". In: *NeurIPS 2022 Workshop on Information-Theoretic Principles in Cognitive Systems*. URL: https://openreview.net/forum?id=yf8suFtNZ5v.

Tucker, Mycal, Huao Li, Siddharth Agrawal, Dana Hughes, Katia Sycara, Michael Lewis, and Julie A Shah (2021). "Emergent Discrete Communication in Semantic Spaces". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., pp. 10574–10586. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/5812f92450ccaf17275500841c70924a-Paper.pdf.

Tucker, Mycal, Julie A. Shah, Roger Philip Levy, and Noga Zaslavsky (2022b). *Towards Human-Agent Communication via the Information Bottleneck Principle*. arXiv: 2207.00088.

Ueda, Ryo, Taiga Ishii, and Yusuke Miyao (2023). “On the Word Boundaries of Emergent Languages Based on Harris’s Articulation Scheme”. In: *The Eleventh International Conference on Learning Representations*. URL: https://openreview.net/forum?id=b4t9_XASt6G.

Ueda, Ryo, Taiga Ishii, Koki Washio, and Yusuke Miyao (2022). “Categorial Grammar Induction as a Compositionality Measure for Emergent Languages in Signaling Games”. In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=Sbgb7b0Q-5.

Ullman, Michael T. (2001a). “A neurocognitive perspective on language: The declarative/procedural model”. In: *Nature Reviews Neuroscience* 2, pp. 717–726.

— (2001b). “The neural basis of lexicon and grammar in first and second language: the declarative/procedural model”. In: *Bilingualism: Language and Cognition* 4.2, pp. 105–122. DOI: 10.1017/S1366728901000220.

Unger, Thomas A. and Elia Bruni (2020). *Generalizing Emergent Communication*. eprint: arXiv:2001.01772. URL: https://arxiv.org/abs/2001.01772.

Verma, Shresth and Joydip Dhar (2019). “Emergence of Writing Systems Through Multi-Agent Cooperation”. In: DOI: 10.1609/aaai.v34i10.7243. eprint: arXiv:1910.00741. URL: https://arxiv.org/abs/1910.00741.

Virpioja, Sami, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo (2013). *Morfessor 2.0: Python Implementation and Extensions for Morfessor Baseline*. Tech. rep. ISBN 978-952-60-5501-5. Helsinki, Finland: Aalto University.

Virtanen, Pauli, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors (2020). “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17, pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

Wah, C., S. Branson, P. Welinder, P. Perona, and S. Belongie (2011). *The Caltech-UCSD Birds-200-2011 Dataset*. Tech. rep. CNS-TR-2011-001. California Institute of Technology.

Wal, Oskar van der, Silvan de Boer, Elia Bruni, and Dieuwke Hupkes (Nov. 2020). “The Grammar of Emergent Languages”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu. Online: Association for Computational Linguistics, pp. 3339–3359. DOI: 10.18653/v1/2020.emnlp-main.270. URL: https://aclanthology.org/2020.emnlp-main.270.

Wang, Alex, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman (Nov. 2018). “GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding”. In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Ed. by Tal Linzen, Grzegorz Chrupała, and Afra Alishahi. Brussels, Belgium: Association for Computational Linguistics, pp. 353–355. DOI: 10.18653/v1/W18-5446. URL: https://aclanthology.org/W18-5446/.

Wang, Woodrow Z., Andy Shih, Annie Xie, and Dorsa Sadigh (2021). “Influencing Towards Stable Multi-Agent Interactions”. In: *CoRL*.

Wang, Yuqi, Xu-Yao Zhang, Cheng-Lin Liu, and Zhaoxiang Zhang (2022). *Emergence of Machine Language: Towards Symbolic Intelligence with Neural Networks*. eprint: arXiv:2201.05489. URL: https://arxiv.org/abs/2201.05489.

Warstadt, Alex and Samuel R Bowman (2022). "What artificial neural networks can tell us about human language acquisition". In: *Algebraic Structures in Natural Language*, pp. 17–60.

— (2020). *Can neural networks acquire a structural bias from raw linguistic data?* arXiv: 2007.06761 [cs.CL]. URL: https://arxiv.org/abs/2007.06761.

Watanabe, Shuhei (2023). *Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance*. arXiv: 2304.11127 [cs.LG]. URL: https://arxiv.org/abs/2304.11127.

Weidinger, Laura, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel (2021). *Ethical and social risks of harm from Language Models*. arXiv: 2112.04359 [cs.CL]. URL: https://arxiv.org/abs/2112.04359.

Weinreich, Uriel, William Labov, and Marvin Herzog (1968). "Empirical foundations for a theory of language change". In: *Directions for Historical Linguistics*. Ed. by Winfred Lehman. Austin: University of Texas Press, pp. 95–88.

Werner, Gregory M. and Michael G. Dyer (1991). "Evolution of Communication in Artificial Organisms". In: *Artificial Life II*. Ed. by C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, pp. 659–687.

Wieczorek, Tobias J., T. Tchumatchenko, Carlos Wert Carvajal, and M. Eggl (2023). *A framework for the emergence and analysis of language in social learning agents*.

Wikimedia Foundation (n.d.). *Wikimedia Downloads*. URL: https://dumps.wikimedia.org.

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush (Oct. 2020). "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Ed. by Qun Liu and David Schlangen. Online: Association for Computational Linguistics, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. URL: https://aclanthology.org/2020.emnlp-demos.6/.

Xiang, Ping, Hangguan Shan, Zhou Su, Zhaoyang Zhang, Chen Chen, and Er-Ping Li (2023). "Multi-Agent Reinforcement Learning-Based Decentralized Spectrum Access in Vehicular Networks With Emergent Communication". In: *IEEE Communications Letters* 27, pp. 195–199.

Xu, Zhenlin, Marc Niethammer, and Colin A Raffel (2022). "Compositional generalization in unsupervised compositional representation learning: A study on disentanglement and emergent language". In: *Advances in Neural Information Processing Systems* 35, pp. 25074–25087.

Yao, Shunyu, Mo Yu, Yang Zhang, Karthik R Narasimhan, Joshua B. Tenenbaum, and Chuang Gan (2022). "Linking Emergent and Natural Languages via Corpus Transfer". In:

International Conference on Learning Representations. URL: https://openreview.net/forum?id=49A1Y6tRhaq.

Yat Long Lo, Biswa Sengupta (2022). *Learning to Ground Decentralized Multi-Agent Communication with Contrastive Learning*. eprint: arXiv:2203.03344. URL: https://arxiv.org/abs/2203.03344.

Yu, Dhara, Jesse Mu, and Noah Goodman (2022). "Emergent Covert Signaling in Adversarial Reference Games". In: *Emergent Communication Workshop at ICLR 2022*. URL: https://openreview.net/forum?id=H-eMQbR7Z5.

Yuan, Luyao, Zipeng Fu, Jingyue Shen, Lu Xu, Junhong Shen, and Song-Chun Zhu (2020). "Emergence of Pragmatics from Referential Game between Theory of Mind Agents". In: eprint: arXiv:2001.07752. URL: https://arxiv.org/abs/2001.07752.

Yuan, Luyao, Zipeng Fu, Linqi Zhou, Kexin Yang, and Song-Chun Zhu (2021). *Emergence of Theory of Mind Collaboration in Multiagent Systems*. eprint: arXiv:2110.00121. URL: https://arxiv.org/abs/2110.00121.

Zaslavsky, Noga, Charles Kemp, Naftali Tishby, and Terry Regier (2019). "Color Naming Reflects Both Perceptual Structure and Communicative Need". In: *Topics in Cognitive Science* 11.1, pp. 207–219. DOI: https://doi.org/10.1111/tops.12395. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/tops.12395. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/tops.12395.

Zhang, Qi, Richard L. Lewis, Satinder Singh, and Edmund H. Durfee (2019). "Learning to Communicate and Solve Visual Blocks-World Tasks". In: *AAAI*.

Zhang, Yi, Arturs Backurs, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, and Tal Wagner (2022). *Unveiling Transformers with LEGO: a synthetic reasoning task*. arXiv: 2206.04301 [cs.LG]. URL: https://arxiv.org/abs/2206.04301.

Zheng, Lianmin, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. (2024). "Judging llm-as-a-judge with mt-bench and chatbot arena". In: *Advances in Neural Information Processing Systems* 36.

Zipf, George K. (1950). "Human behavior and the principle of least effort". In: *Journal of Clinical Psychology* 6.3, pp. 306–306. DOI: https://doi.org/10.1002/1097-4679(195007)6:3<306::AID-JCLP2270060331>3.0.CO;2-7. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/1097-4679%28195007%296%3A3%3C306%3A%3AAID-JCLP2270060331%3E3.0.CO%3B2-7. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/1097-4679%28195007%296%3A3%3C306%3A%3AAID-JCLP2270060331%3E3.0.CO%3B2-7.

Zoph, Barret, Deniz Yuret, Jonathan May, and Kevin Knight (Nov. 2016). "Transfer Learning for Low-Resource Neural Machine Translation". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Ed. by Jian Su, Kevin Duh, and Xavier Carreras. Austin, Texas: Association for Computational Linguistics, pp. 1568–1575. DOI: 10.18653/v1/D16-1163. URL: https://aclanthology.org/D16-1163.

Zubek, Julian, Tomasz Korbak, and Joanna Rączaszek-Leonardi (2023). *Models of symbol emergence in communication: a conceptual review and a guide for avoiding local minima*. eprint: arXiv:2303.04544. URL: https://arxiv.org/abs/2303.04544.

Appendix A

Review of the Applications of Emergent Language

A.1 Review Methods

In this section, we give a brief account of the methods used for obtaining the papers referenced in this review. As a considerable amount of the content in this review will draw from the authors' background knowledge, describing the methods does not imply that this paper is "fully reproducible". Nevertheless, presenting the process used for producing this paper can aid in understanding its context and origin.

A.1.1 Collecting papers

To collect papers we searched arXiv (`https://arxiv.org/`) and Semantic Scholar (`https://www.semanticscholar.org/`) for: "emergent language", "emergent communication", "language emergence", and "communication emergence". Any paper that had a title plausibly related to emergent communication was passed along to annotation stage. Occasionally, the abstract would be skimmed at this stage, but here we aired on the side of recall and not precision.

We selected arXiv because (1) a majority of emergent communication papers are posted on arXiv and (2) the *Computer Science* archive provides a good signal to noise ratio due to the type of research that tends to be posted on arXiv. We supplemented arXiv with Semantic Scholar primarily to collect emergent communication papers that come from sources outside typical computer science discipline (as well as any CS papers which simply were not posted to arXiv). Additionally, we collected papers from all years of EmeCom[1], a series of workshops on (primarily deep learning-based) emergent communication. With very few (<5) exceptions, we gathered all of the emergent communication papers through this method. This was done primarily because it provided a good balance between overall coverage, principled methodology, and labor intensity.

[1]EmeCom URLs `https://sites.google.com/site/emecom2017/accepted-papers`, `https://sites.google.com/site/emecom2018/accepted-papers`, `https://sites.google.com/view/emecom2019/accepted-papers`, `https://sites.google.com/view/emecom2020/accepted-papers`, and `https://openreview.net/group?id=ICLR.cc/2022/Workshop/EmeCom#all-submissions`.

arXiv We searched arXiv with a disjunction of the aforementioned queries starting with the year 2015 up until present. This search was originally performed around July 1, 2022 and then again around May 4, 2023.[2] The result is approximately 4 500 entries of which 157 were selected for the next stage.

Semantic Scholar The search process of Semantic Scholar was a bit more complicated because the results could not be reviewed exhaustively. This was in part because the results were sorted by relevance and also because a wider range of topics were searched. Thus, the first 100−400 results were inspected, until further results seemed largely irrelevant to emergent communication. Similarly to arXiv, the searches were performed in two batches with the first one spanning 2015 to July 28, 2022:

- "emergent language": all fields; 100 titles reviewed: `https://www.semanticscholar.org/search?year%5B0%5D=2015&year%5B1%5D=2022&fos%5B0%5D=computer-science&fos%5B1%5D=engineering&fos%5B2%5D=linguistics&fos%5B3%5D=philosophy&fos%5B4%5D=psychology&fos%5B5%5D=sociology&fos%5B6%5D=mathematics&fos%5B7%5D=biology&fos%5B8%5D=economics&q=emergent%20language&sort=relevance`
- "emergent language": computer science; 220 titles reviewed: `https://www.semanticscholar.org/search?year[0]=2015&year[1]=2022&fos[0]=computer-science&fos[1]=engineering&fos[2]=mathematics&q=emergent%20language&sort=relevance&page=1`
- "emergent communication": all fields; 370 titles reviewed: `https://www.semanticscholar.org/search?year[0]=2015&year[1]=2022&fos[0]=computer-science&fos[1]=engineering&fos[2]=mathematics&q=emergent%20communication&sort=relevance&page=1`
- "emergent communication": computer science: 260 titles reviewed: `https://www.semanticscholar.org/search?year%5B0%5D=2015&year%5B1%5D=2022&fos%5B0%5D=computer-science&fos%5B1%5D=engineering&fos%5B2%5D=mathematics&fos%5B3%5D=biology&fos%5B4%5D=economics&fos%5B5%5D=linguistics&fos%5B6%5D=philosophy&fos%5B7%5D=psychology&fos%5B8%5D=sociology&q=emergent%20communication&sort=relevance&page=26`
- "language emergence": computer science; 400 pages reviewed: `https://www.semanticscholar.org/search?year[0]=2015&year[1]=2022&fos[0]=computer-science&fos[1]=engineering&fos[2]=mathematics&q=language%20emergence&sort=relevance`
- "language emergence": biology, economics, linguistics, philosophy, psychology, sociology; 110 titles reviewed: `https://www.semanticscholar.org/search?year%5B0%5D=2015&year%5B1%5D=2022&fos%5B0%5D=biology&fos%5B1%5D=economics&fos%5B2%5D=linguistics&fos%5B3%5D=philosophy&fos%5B4%5D=psychology&fos%5B5%5D=sociology&q=language%20emergence&sort=relevance&page=1`

[2] Search URL for arXiv: `https://arxiv.org/search/advanced?terms-0-operator=AND&terms-0-term=emergent+language&terms-0-field=all&terms-1-operator=OR&terms-1-term=language+emergence&terms-1-field=all&terms-2-operator=OR&terms-2-term=emergent+communication&terms-2-field=all&terms-3-operator=OR&terms-3-term=communication+emergence&terms-3-field=all&classification-computer_science=y&classification-physics_archives=all&classification-include_cross_list=include&date-year=&date-filter_by=date_range&date-from_date=2015-01-01&date-to_date=2023-05-04&date-date_type=submitted_date_first&abstracts=hide&size=100&order=-announced_date_first`.

Category	Number of Papers
Initial Search	443
Duplicate	84
Out-of-scope	106
Not a research paper	5
No access	16
Included	232

Table A.1: Number of papers excluded from initial search for various reasons. Duplicate papers were either overlaps between different sources or papers that were substantially similar and by the same authors.

The second pass was performed on May 8, 2023, spanning 2022 and 2023:

- "emergent language": all fields; 300 titles reviewed: `https://www.semanticscholar.org/search?year%5B0%5D=2022&year%5B1%5D=2023&fos%5B0%5D=computer-science&fos%5B1%5D=engineering&fos%5B2%5D=linguistics&fos%5B3%5D=philosophy&fos%5B4%5D=psychology&fos%5B5%5D=sociology&fos%5B6%5D=mathematics&fos%5B7%5D=biology&fos%5B8%5D=economics&q=emergent%20language&sort=relevance&page=2`
- "emergent communication": computer science, engineering; 250 titles reviewed: `https://www.semanticscholar.org/search?year[0]=2022&year[1]=2023&fos[0]=computer-science&fos[1]=engineering&fos[2]=mathematics&q=emergent%20communication&sort=relevance&page=1`
- "emergent communication": computer science, egineering, linguistics, philosophy, psychology, mathematics, biology, economics; 100 titles reviewed: `https://www.semanticscholar.org/search?year%5B0%5D=2015&year%5B1%5D=2022&fos%5B0%5D=computer-science&fos%5B1%5D=engineering&fos%5B2%5D=linguistics&fos%5B3%5D=philosophy&fos%5B4%5D=psychology&fos%5B5%5D=sociology&fos%5B6%5D=mathematics&fos%5B7%5D=biology&fos%5B8%5D=economics&q=emergent%20language&sort=relevance`
- "language emergence": all fields; 300 titles reviewed: `https://www.semanticscholar.org/search?year[0]=2022&year[1]=2023&fos[0]=computer-science&fos[1]=engineering&fos[2]=linguistics&fos[3]=philosophy&fos[4]=psychology&fos[5]=sociology&fos[6]=mathematics&fos[7]=biology&fos[8]=economics&q=language%20emergence&sort=relevance`

These searches yielded 214 papers which were selected for the next stage.

A.1.2 Goal categorization

Given these papers from our initial search, we reviewed the papers first to determine if they are in-scope (as described by Section 2.1.1) and second to categorize them according to the goals they pursued. The number of papers included and excluded is given in Table A.1.

The following categories were used for the annotation of the included papers. They do not precisely line up with the section ultimately used for the paper largely because the annotation categories were determined largely *a priori* while the paper sections were determined *a posteriori*.

- Internal
  - measure properties of emergent communication
  - produce some property in emergent communication
  - other emergent communication improvement (e.g., efficiency, robustness)
  - tooling
  - theoretical frameworks
- Task-driven
  - artificial general intelligence, better NLP
  - replication of natural language
  - alternative data source/paradigm
  - robust multiagent communication
  - explainable models
  - synthetic data for evaluation
  - communicating with humans
- Knowledge-driven
  - increase understanding of language in general
  - evolution of language:
  - fundamentals of language (e.g., phonology, lexicon, syntax)
    - phonology
    - syntax
    - semantics
    - compositionality
    - morphology
    - pragmatics
    - sociolinguistics
  - language acquisition
  - cognitive science and language
    - perception

Some of these categories were eventually discarded since they either did not receive much attention in the literature or the category itself was too vague to productively discussed. A quantitative summary of the categorization after remapping them to section in the paper is for each paper is presented in Section 2.5.1.

For the majority of papers, we would read the abstract, introduction, and conclusion in order to assign the proper categories; this would take, on average, 6 minutes to complete per paper. These sections are the most common places for describing the broader applications and contributions of the papers. Paper were reviewed more thoroughly as needed to determine the proper categories. Determining which papers to highlight in the body of this paper depended on the application. For applications with a small number of papers, we were able to exhaustively discuss the applicable papers. For applications with many papers, we highlighted a representative sample of the papers which best illustrated that application.

A.2 Complete List of Reviewed Papers

Rederiving human language *No papers.*

Metrics for emergent communication Bogin et al. (2018), Bosc et al. (2022), Bosc (2022), Chaabouni et al. (2020), Chaabouni et al. (2022), Denamganaï et al. (2023), S. Guo et al. (2021), S. Guo et al. (2020), Korbak et al. (2020), Korbak et al. (2021), Kuciński et al. (2020), Lowe et al. (2019), J. Mu et al. (2021), Perkins (2021a), Resnick et al. (2020), Thomas et al. (2022), Tucker et al. (2022b), Verma et al. (2019), Wal et al. (2020), and Yao et al. (2022)

Theoretical models Boldt et al. (2022b), Boldt et al. (2022a), Eecke et al. (2023), Khomtchouk et al. (2018), Ren et al. (2020), Resnick et al. (2020), Rita et al. (2022b), and Tucker et al. (2022b)

Tooling Denamganaï et al. (2020b), Ikram et al. (2021), Kharitonov et al. (2019), and Perkins (2021b)

Synthetic language data Dessì et al. (2021), Downey et al. (2023), Yaoyiran Li et al. (2020), Y. Mu et al. (2023), Santamaría-Pang et al. (2019), Steinert-Threlkeld et al. (2022), and Yao et al. (2022)

Multi-agent communication Bullard et al. (2021), Bullard et al. (2020), S. Chen et al. (2022), Cope et al. (2021), Karten et al. (2023), S. Li et al. (2022), Mahaut et al. (2023), Masquil et al. (2022), Mul et al. (2019), Piazza et al. (2023), Thomas et al. (2022), Tucker et al. (2021), Y. Wang et al. (2022), and Xiang et al. (2023)

Interacting with humans Hagiwara et al. (2021), Karten et al. (2022c), S. Li et al. (2022), Mihai et al. (2021a), Tucker et al. (2021), and Tucker et al. (2022a)

Explainable machine learning models Chowdhury et al. (2020b), Chowdhury et al. (2021), Chowdhury et al. (2020a), and Santamaria-Pang et al. (2020)

Language, cognition, and perception Bouchacourt et al. (2018), Chaabouni et al. (2021), Choi et al. (2018), Cowen-Rivers et al. (2020), Dekker et al. (2020), Denamganaï et al. (2023), Dessì et al. (2021), Feng et al. (2023), Grupen et al. (2020), Hagiwara et al. (2021), Herrmann et al. (2022), Kågebäck et al. (2018), Lazaridou et al. (2018), Mahaut et al. (2023), Ohmer et al. (2021b), Masquil et al. (2022), Mihai et al. (2021b), Ohmer et al. (2021a), Ossenkopf et al. (2022), Patel et al. (2021), Piazza et al. (2023), Portelance et al. (2021), Sabathiel et al. (2022), Todo et al. (2020), Yuan et al. (2021), Yuan et al. (2020), and Zubek et al. (2023)

Origin of language Chaabouni et al. (2020), Cogswell et al. (2019), Dagan et al. (2020), Dekker et al. (2020), Galke et al. (2022), Grossi et al. (2017), Grupen et al. (2021), LaCroix (2019), Moulin-Frier et al. (2020), Ohmer et al. (2021a), and Ren et al. (2020)

Language change Dekker et al. (2020) and Graesser et al. (2019)

Language acquisition Cope et al. (2022), Kharitonov et al. (2020a), Korbak et al. (2019), Korbak et al. (2021), F. Li et al. (2019), and Portelance et al. (2021)

Linguistic variables, phonology (focused) Eloff et al. (2021) and Lan et al. (2020)

Linguistic variables, phonology (related) Eloff et al. (2021), Lan et al. (2020), and Verma et al. (2019)

Linguistic variables, morphology (focused) *No papers.*

Linguistic variables, morphology (related) Mihai et al. (2021a)

Linguistic variables, syntax (focused) Chaabouni et al. (2019b), Ueda et al. (2022), and Wal et al. (2020)

Linguistic variables, syntax (related) Bosc et al. (2022), Chaabouni et al. (2019b), Resnick et al. (2020), Słowik et al. (2020b), Ueda et al. (2022), and Wal et al. (2020)

Linguistic variables, semantics (focused) Chaabouni et al. (2019a), Chaabouni et al. (2021), Kågebäck et al. (2018), Rodríguez Luna et al. (2020), Rita et al. (2020), and Steinert-Threlkeld (2019)

Linguistic variables, semantics (related) Bosc et al. (2022), Bouchacourt et al. (2019), Chaabouni et al. (2020), Chaabouni et al. (2019a), Chaabouni et al. (2021), Cope et al. (2021), Dessì et al. (2019), Garcia et al. (2022), Grupen et al. (2020), S. Guo et al. (2021), S. Guo et al. (2020), S. Guo (2019), S. Guo et al. (2019), Herrmann et al. (2022), Kågebäck et al. (2018), Kharitonov et al. (2020a), Kharitonov et al. (2020b), Khomtchouk et al. (2018), Korbak et al. (2019), Kågebäck (2018), Lazaridou et al. (2016), T. Lin et al. (2021), Rodríguez Luna et al. (2020), Ohmer et al. (2021b), Mihai et al. (2021b), Mihai et al. (2019), J. Mu et al. (2021), Ohmer et al. (2021a), Portelance et al. (2021), S. Qiu et al. (2021), Rita et al. (2020), Sabathiel et al. (2022), Steinert-Threlkeld (2019), Słowik et al. (2020a), Tucker et al. (2021), Tucker et al. (2022b), Tucker et al. (2022b), Unger et al. (2020), Yu et al. (2022), Q. Zhang et al. (2019), and Zubek et al. (2023)

Linguistic variables, compositionality (focused) *No papers.*

Linguistic variables, compositionality (related) Auersperger et al. (2022), Bogin et al. (2018), Bosc et al. (2022), Bosc (2022), Chaabouni et al. (2020), Chaabouni et al. (2022), Y. Chen et al. (2023), Choi et al. (2018), Cogswell et al. (2019), Denamganaï et al. (2020a), Denamganaï et al. (2023), Galke et al. (2022), Garcia et al. (2022), S. Guo et al. (2020), S. Guo (2019), S. Guo et al. (2019), Havrylov et al. (2017), Hazra et al. (2021), Hazra et al. (2020), Karten et al. (2022b), Keresztury et al. (2020), Kharitonov et al. (2020a), Korbak et al. (2021), Kottur et al. (2017), Kuciński et al. (2021), Kuciński et al. (2020), LaCroix (2019), Lan et al. (2020), Lazaridou et al. (2018), F. Li et al. (2019), Liang et al. (2020), Rodríguez Luna et al. (2020), Mordatch et al. (2018), Ohmer et al. (2022), Ohmer et al. (2021a), Perkins (2021a), Ren et al. (2020), Resnick et al. (2020), Rita et al. (2022a), Rita et al. (2022b), Steinert-Threlkeld (2020), Słowik et al. (2020b), Thomas et al. (2022), Ueda et al. (2022), Xu et al. (2022), and Todo et al. (2020)

Linguistic variables, pragmatics (focused) Kang et al. (2020)

Linguistic variables, pragmatics (related) Blumenkamp et al. (2020), Bouchacourt et al. (2019), Bullard et al. (2021), Bullard et al. (2020), Cao et al. (2018), Eccles et al. (2019), Evtimova et al. (2017), Kalinowska et al. (2022b), Kalinowska et al. (2022a), Kang et al. (2020), Karten et al. (2023), Kolb et al. (2019), Leni et al. (2018), Lipinski et al. (2022), Lowe et al. (2019), Masquil et al. (2022), Mordatch et al. (2018), Noukhovitch et al. (2021), Ossenkopf et al. (2022), Ossenkopf (2019), Piazza et al. (2023), Yu et al. (2022), and Yuan et al. (2020)

Linguistic variables, sociolinguistics (focused) Dekker et al. (2020), Fulker et al. (2022), Graesser et al. (2019), and Kim et al. (2021)

Linguistic variables, sociolinguistics (related) Dekker et al. (2020), Fitzgerald (2019), Fulker et al. (2022), Galke et al. (2022), Graesser et al. (2019), Grupen et al. (2022), S. Gupta et al. (2019), Kim et al. (2021), F. Li et al. (2019), Liang et al. (2020), and Rita et al. (2022a)

No applications Boldt et al. (2022c), Brandizzi et al. (2022a), Carmeli et al. (2022), Foguelman et al. (2021), Gaya (2017), S. Guo (2019), S. Gupta et al. (2020), Hagiwara et al. (2019), Kajić et al. (2020), Karch et al. (2022), Karten et al. (2022a), Karten et al. (2022a), Lannelongue et al. (2019), Lazaridou et al. (2020), Lee et al. (2018), Lipowska et al. (2018), Yat Long Lo (2022), Lo et al. (2022), Lowe et al. (2020), Nevens et al. (2020), Brandizzi et al. (2022b), Raviv et al. (2022), Sirota et al. (2019), Taniguchi et al. (2022), W. Z. Wang et al. (2021), and Wieczorek et al. (2023)

Appendix B

Building a Library of Emergent Languages

B.1 ECS-Level Metadata Specification

source The URL for the repository implementing the ECS.

upstream_source The URL of the original repo if **source** is a fork.

paper The URL of the paper documenting the ECS (if any).

game_type The high level category of the game implemented in the ECS; currently one of *signaling*, *conversation*, or *navigation*.

game_subtype A finer-grained categorization of the game, if applicable.

observation_type The type of observation that the agents make; currently either *vector* or *image* (i.e., an image embedding).

observation_continuous Whether or not the observation is continuous as opposed to discrete (e.g., image embeddings versus concatenated one-hot vectors).

data_source Whether the data being communicated about is from a natural source (e.g., pictures), is synthetic, or comes from another source (e.g., in a social deduction game).

variants A dictionary where each entry corresponds to one of the variants of the particular ECS. Each entry in the dictionary contains any relevant hyperparameters that distinguish it from the other variants.

seeding_available Whether or not the ECS implements seeding the random elements of the system.

multi_step Whether or not the ECS has multiple steps per episode.

symmetric_agents Whether or not agents both send and receive messages.

multi_utterance Whether or not multiple utterances are included per line in the dataset.

```yaml
origin:
  upstream_source:
    https://github.com/google-deepmind/emergent_communication...
  paper: https://openreview.net/forum?id=AUGBfDIV9rL
system:
  game_type: signaling
  data_source: natural
  game_subtype: discrimination
  observation_type: image
  observation_continuous: true
  seeding_available: true
  multi_step: false
  more_than_2_agents: true
  multi_utterance: false
  symmetric_agents: false
  variants:
    imagenet-1x10:
      n_receivers: 10
      n_senders: 1
    imagenet-10x10:
      n_receivers: 10
      n_senders: 10
    imagenet-5x5:
      n_receivers: 5
      n_senders: 5
    imagenet-1x1:
      n_receivers: 1
      n_senders: 1
    imagenet-10x1:
      n_receivers: 1
      n_senders: 10
```


Figure B.1: Example of an ECS metadata file in the YAML format.

more_than_2_agents Whether or not the ECS has a population of >2 agents.

B.2 ECS-Level Metadata Example

See Figure B.1.

B.3 Papers based on the signaling game

J. Mu et al. (2021), Ohmer et al. (2022), Yao et al. (2022), Rita et al. (2022a), Ohmer et al. (2021a), Kuciński et al. (2021), Portelance et al. (2021), Tucker et al. (2021), Dessì et al. (2021), Bullard et al. (2021), Perkins (2021a), Mihai et al. (2021b), Denamganaï et al. (2020a), S. Guo et al. (2020), Yaoyiran Li et al. (2020), Rita et al. (2020), Chowdhury et al. (2020b), Chowdhury et al. (2021), Lan et al. (2020), Chaabouni et al. (2020), Rodríguez Luna et al. (2020), Kharitonov et al. (2020a), Ren et al. (2020), Słowik et al. (2020b), Lowe et al.

(2020), Keresztury et al. (2020), Dagan et al. (2020), Mihai et al. (2019), Dessì et al. (2019), S. Guo et al. (2019), Steinert-Threlkeld (2019), F. Li et al. (2019), Kharitonov et al. (2020b), Chaabouni et al. (2019a), Khomtchouk et al. (2018), Bouchacourt et al. (2018), Lazaridou et al. (2018), Havrylov et al. (2017), Lazaridou et al. (2016), Mahaut et al. (2023), Carmeli et al. (2022), Rita et al. (2022b), and Downey et al. (2023)

B.4 Per system analysis

See Tables B.1 to B.4.

name	Token Count	Line Count	Tokens per Line	Tokens per Line SD
babyai-sr/GoToObj	130648	6116	21.361674	12.470737
babyai-sr/GoToObjLocked	272712	5629	48.447682	15.939260
babyai-sr/GoToObjLocked_ambiguous	229504	5507	41.674959	17.414703
babyai-sr/GoToObjLocked_ambiguous-freq_1	2605112	5179	503.014482	45.179870
babyai-sr/GoToObjLocked_ambiguous-freq_2	1061520	5396	196.723499	70.458489
babyai-sr/GoToObjLocked_ambiguous-freq_32	67248	5496	12.235808	3.993043
babyai-sr/GoToObjLocked_ambiguous-freq_4	402248	5728	70.224860	30.849604
babyai-sr/GoToObjLocked_ambiguous-msg_16	511840	5514	92.825535	33.765546
babyai-sr/GoToObjLocked_ambiguous-msg_32	855744	5508	155.363834	58.659355
babyai-sr/GoToObjLocked_ambiguous-msg_4	103228	5730	18.015358	6.603910
babyai-sr/GoToObjUnlocked	118752	6077	19.541221	7.060342
babyai-sr/GoToObjUnlocked-freq_1	1666456	6006	277.465201	205.399768
babyai-sr/GoToObjUnlocked-freq_2	333552	5777	57.737926	28.293252
babyai-sr/GoToObjUnlocked-freq_32	48616	6001	8.101316	0.894576
babyai-sr/GoToObjUnlocked-freq_4	193176	5762	33.525859	13.813609
babyai-sr/GoToObjUnlocked-msg_16	273008	6038	45.214972	18.173718
babyai-sr/GoToObjUnlocked-msg_32	469440	5765	81.429315	33.131090
babyai-sr/GoToObjUnlocked-msg_4	58588	5759	10.173294	4.351285
corpus-transfer-yao-et-al/cc	42977805	2865187	15.000000	0.000000
corpus-transfer-yao-et-al/coco_2014	1241745	82783	15.000000	0.000000
ec-at-scale/imagenet-10x1	2500000	250000	10.000000	0.000000
ec-at-scale/imagenet-10x10	2500000	250000	10.000000	0.000000
ec-at-scale/imagenet-1x1	2500000	250000	10.000000	0.000000
ec-at-scale/imagenet-1x10	2500000	250000	10.000000	0.000000
ec-at-scale/imagenet-5x5	2500000	250000	10.000000	0.000000
egg-discrimination/4-attr_4-val_3-dist_0-seed	110000	10000	11.000000	0.000000
egg-discrimination/4-attr_4-val_3-dist_1-seed	110000	10000	11.000000	0.000000
egg-discrimination/4-attr_4-val_3-dist_2-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_3-dist_0-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_3-dist_1-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_3-dist_2-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_9-dist_0-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_9-dist_1-seed	110000	10000	11.000000	0.000000
egg-discrimination/6-attr_6-val_9-dist_2-seed	110000	10000	11.000000	0.000000
egg-discrimination/8-attr_8-val_3-dist_0-seed	110000	10000	11.000000	0.000000
egg-discrimination/8-attr_8-val_3-dist_1-seed	110000	10000	11.000000	0.000000
egg-discrimination/8-attr_8-val_3-dist_2-seed	110000	10000	11.000000	0.000000
egg-reconstruction/4-attr_4-val_10-vocab_10-len	110000	10000	11.000000	0.000000
egg-reconstruction/6-attr_6-val_10-vocab_10-len	110000	10000	11.000000	0.000000
egg-reconstruction/8-attr_8-val_10-vocab_10-len	110000	10000	11.000000	0.000000
generalizations-mu-goodman/cub-concept	1333330	133333	10.000000	0.000000
generalizations-mu-goodman/cub-reference	1333330	133333	10.000000	0.000000
generalizations-mu-goodman/cub-set_reference	1333330	133333	10.000000	0.000000
generalizations-mu-goodman/shapeworld-concept	1164800	166400	7.000000	0.000000
generalizations-mu-goodman/shapeworld-reference	1164800	166400	7.000000	0.000000
generalizations-mu-goodman/shapeworld-set_reference	1164800	166400	7.000000	0.000000
nav-to-center/lexicon_size_11	65528	10000	6.552800	2.521193
nav-to-center/lexicon_size_118	58664	10000	5.866400	2.167199
nav-to-center/lexicon_size_17	59936	10000	5.993600	2.282533
nav-to-center/lexicon_size_174	63129	10000	6.312900	2.434747
nav-to-center/lexicon_size_25	61659	10000	6.165900	2.323010
nav-to-center/lexicon_size_255	60054	10000	6.005400	2.263000
nav-to-center/lexicon_size_37	62753	10000	6.275300	2.396604
nav-to-center/lexicon_size_54	58778	10000	5.877800	2.197195
nav-to-center/lexicon_size_7	61295	10000	6.129500	2.315368
nav-to-center/lexicon_size_80	60250	10000	6.025000	2.256097
nav-to-center/temperature_0.1	74939	10000	7.493900	3.180623
nav-to-center/temperature_0.167	72255	10000	7.225500	2.995171
nav-to-center/temperature_0.278	75732	10000	7.573200	3.106580
nav-to-center/temperature_0.464	79810	10000	7.981000	3.564778
nav-to-center/temperature_0.774	65665	10000	6.566500	2.527326
nav-to-center/temperature_1.29	62566	10000	6.256600	2.364647
nav-to-center/temperature_10	63105	10000	6.310500	2.397059
nav-to-center/temperature_2.15	62019	10000	6.201900	2.314160
nav-to-center/temperature_3.59	58786	10000	5.878600	2.187661
nav-to-center/temperature_5.99	61106	10000	6.110600	2.289491
rlupus/21-player.run-0	7131411	1001	7124.286713	445.837385
rlupus/21-player.run-1	7196469	999	7203.672673	396.806665
rlupus/21-player.run-2	7212723	1000	7212.723000	404.660045
rlupus/9-player.run-0	565164	1003	563.473579	14.708875
rlupus/9-player.run-1	417924	1010	413.786139	124.676603
rlupus/9-player.run-2	414612	1000	414.612000	124.794461
rlupus/9-player.run-3	416538	1003	415.292124	124.887337

Table B.1

name	Unique Tokens	Unique Lines	EoS Token Present	EoS Padding
babyai-sr/GoToObj	5	653	False	False
babyai-sr/GoToObjLocked	6	788	False	False
babyai-sr/GoToObjLocked_ambiguous	6	1253	False	False
babyai-sr/GoToObjLocked_ambiguous-freq_1	5	5171	False	False
babyai-sr/GoToObjLocked_ambiguous-freq_2	4	3078	False	False
babyai-sr/GoToObjLocked_ambiguous-freq_32	3	18	False	False
babyai-sr/GoToObjLocked_ambiguous-freq_4	6	3241	False	False
babyai-sr/GoToObjLocked_ambiguous-msg_16	9	4428	False	False
babyai-sr/GoToObjLocked_ambiguous-msg_32	3	1887	False	False
babyai-sr/GoToObjLocked_ambiguous-msg_4	2	1362	False	False
babyai-sr/GoToObjUnlocked	7	521	False	False
babyai-sr/GoToObjUnlocked-freq_1	7	4614	False	False
babyai-sr/GoToObjUnlocked-freq_2	8	3820	False	False
babyai-sr/GoToObjUnlocked-freq_32	4	41	False	False
babyai-sr/GoToObjUnlocked-freq_4	7	2766	False	False
babyai-sr/GoToObjUnlocked-msg_16	13	1740	False	False
babyai-sr/GoToObjUnlocked-msg_32	15	1430	False	False
babyai-sr/GoToObjUnlocked-msg_4	3	400	False	False
corpus-transfer-yao-et-al/cc	391	309405	True	True
corpus-transfer-yao-et-al/coco_2014	902	82783	True	False
ec-at-scale/imagenet-10x1	20	161235	False	False
ec-at-scale/imagenet-10x10	20	126775	False	False
ec-at-scale/imagenet-1x1	20	145834	False	False
ec-at-scale/imagenet-1x10	20	120182	False	False
ec-at-scale/imagenet-5x5	20	169505	False	False
egg-discrimination/4-attr_4-val_3-dist_0-seed	10	240	True	True
egg-discrimination/4-attr_4-val_3-dist_1-seed	10	220	True	True
egg-discrimination/4-attr_4-val_3-dist_2-seed	9	187	True	True
egg-discrimination/6-attr_6-val_3-dist_0-seed	8	2326	True	True
egg-discrimination/6-attr_6-val_3-dist_1-seed	10	3279	True	True
egg-discrimination/6-attr_6-val_3-dist_2-seed	9	1976	True	True
egg-discrimination/6-attr_6-val_9-dist_0-seed	9	2883	True	True
egg-discrimination/6-attr_6-val_9-dist_1-seed	9	1015	False	False
egg-discrimination/6-attr_6-val_9-dist_2-seed	10	2499	True	True
egg-discrimination/8-attr_8-val_3-dist_0-seed	10	2610	True	True
egg-discrimination/8-attr_8-val_3-dist_1-seed	10	2789	True	True
egg-discrimination/8-attr_8-val_3-dist_2-seed	9	2656	True	True
egg-reconstruction/4-attr_4-val_10-vocab_10-len	7	228	True	True
egg-reconstruction/6-attr_6-val_10-vocab_10-len	8	1373	True	True
egg-reconstruction/8-attr_8-val_10-vocab_10-len	8	1464	False	False
generalizations-mu-goodman/cub-concept	23	27163	False	False
generalizations-mu-goodman/cub-reference	23	39457	False	False
generalizations-mu-goodman/cub-set_reference	23	35042	False	False
generalizations-mu-goodman/shapeworld-concept	17	12481	False	False
generalizations-mu-goodman/shapeworld-reference	17	7683	False	False
generalizations-mu-goodman/shapeworld-set_reference	17	28061	True	False
nav-to-center/lexicon_size_11	8	2317	False	False
nav-to-center/lexicon_size_118	61	4392	False	False
nav-to-center/lexicon_size_17	15	3124	False	False
nav-to-center/lexicon_size_174	40	3226	False	False
nav-to-center/lexicon_size_25	12	1961	False	False
nav-to-center/lexicon_size_255	37	3706	False	False
nav-to-center/lexicon_size_37	22	2440	False	False
nav-to-center/lexicon_size_54	43	4911	False	False
nav-to-center/lexicon_size_7	7	1937	False	False
nav-to-center/lexicon_size_80	35	3486	False	False
nav-to-center/temperature_0.1	4	1437	False	False
nav-to-center/temperature_0.167	4	1313	False	False
nav-to-center/temperature_0.278	10	1308	False	False
nav-to-center/temperature_0.464	4	1498	False	False
nav-to-center/temperature_0.774	7	1639	False	False
nav-to-center/temperature_1.29	9	2100	False	False
nav-to-center/temperature_10	64	8793	False	False
nav-to-center/temperature_2.15	64	8643	False	False
nav-to-center/temperature_3.59	64	9044	False	False
nav-to-center/temperature_5.99	64	9263	False	False
rlupus/21-player.run-0	21	1001	False	False
rlupus/21-player.run-1	21	999	False	False
rlupus/21-player.run-2	21	1000	False	False
rlupus/9-player.run-0	9	1003	False	False
rlupus/9-player.run-1	9	1010	False	False
rlupus/9-player.run-2	9	1000	False	False
rlupus/9-player.run-3	9	1003	False	False

Table B.2

name	1-gram Entropy	1-gram Normalized Entropy	Entropy per Line
babyai-sr/GoToObj	1.237631	0.533019	26.437867
babyai-sr/GoToObjLocked	0.986990	0.381820	47.817369
babyai-sr/GoToObjLocked_ambiguous	1.724020	0.666942	71.848479
babyai-sr/GoToObjLocked_ambiguous-freq_1	1.463654	0.630362	736.239281
babyai-sr/GoToObjLocked_ambiguous-freq_2	1.385921	0.692961	272.643237
babyai-sr/GoToObjLocked_ambiguous-freq_32	0.358125	0.225952	4.381954
babyai-sr/GoToObjLocked_ambiguous-freq_4	1.996955	0.772528	140.235868
babyai-sr/GoToObjLocked_ambiguous-msg_16	2.555153	0.806061	237.183454
babyai-sr/GoToObjLocked_ambiguous-msg_32	1.350560	0.852108	209.828108
babyai-sr/GoToObjLocked_ambiguous-msg_4	0.922138	0.922138	16.612643
babyai-sr/GoToObjUnlocked	1.993155	0.709976	38.948688
babyai-sr/GoToObjUnlocked-freq_1	0.896426	0.319313	248.726924
babyai-sr/GoToObjUnlocked-freq_2	2.116083	0.705361	122.178237
babyai-sr/GoToObjUnlocked-freq_32	1.643569	0.821785	13.315074
babyai-sr/GoToObjUnlocked-freq_4	2.165184	0.771254	72.589650
babyai-sr/GoToObjUnlocked-msg_16	2.608207	0.704837	117.930014
babyai-sr/GoToObjUnlocked-msg_32	2.940985	0.752769	239.482412
babyai-sr/GoToObjUnlocked-msg_4	1.453225	0.916883	14.784087
corpus-transfer-yao-et-al/cc	1.398306	0.162386	20.974592
corpus-transfer-yao-et-al/coco_2014	6.599321	0.672235	98.989817
ec-at-scale/imagenet-10x1	3.980879	0.921089	39.808790
ec-at-scale/imagenet-10x10	3.908713	0.904391	39.087127
ec-at-scale/imagenet-1x1	4.121796	0.953694	41.217964
ec-at-scale/imagenet-1x10	3.975498	0.919844	39.754975
ec-at-scale/imagenet-5x5	4.213196	0.974842	42.131963
egg-discrimination/4-attr_4-val_3-dist_0-seed	2.996740	0.902109	32.964144
egg-discrimination/4-attr_4-val_3-dist_1-seed	2.494699	0.750979	27.441685
egg-discrimination/4-attr_4-val_3-dist_2-seed	2.564778	0.809097	28.212561
egg-discrimination/6-attr_6-val_3-dist_0-seed	2.581470	0.860490	28.396171
egg-discrimination/6-attr_6-val_3-dist_1-seed	2.887394	0.869192	31.761330
egg-discrimination/6-attr_6-val_3-dist_2-seed	2.573849	0.811959	28.312341
egg-discrimination/6-attr_6-val_9-dist_0-seed	2.861929	0.902838	31.481224
egg-discrimination/6-attr_6-val_9-dist_1-seed	2.462500	0.776832	27.087504
egg-discrimination/6-attr_6-val_9-dist_2-seed	2.750845	0.828087	30.259294
egg-discrimination/8-attr_8-val_3-dist_0-seed	2.426752	0.730525	26.694277
egg-discrimination/8-attr_8-val_3-dist_1-seed	2.556315	0.769528	28.119469
egg-discrimination/8-attr_8-val_3-dist_2-seed	2.802140	0.883977	30.823535
egg-reconstruction/4-attr_4-val_10-vocab_10-len	2.296329	0.817969	25.259614
egg-reconstruction/6-attr_6-val_10-vocab_10-len	2.573243	0.857748	28.305674
egg-reconstruction/8-attr_8-val_10-vocab_10-len	2.295767	0.765256	25.253441
generalizations-mu-goodman/cub-concept	3.752944	0.829644	37.529443
generalizations-mu-goodman/cub-reference	3.103881	0.686159	31.038812
generalizations-mu-goodman/cub-set_reference	3.213538	0.710400	32.135376
generalizations-mu-goodman/shapeworld-concept	3.226724	0.789420	22.587066
generalizations-mu-goodman/shapeworld-reference	3.224439	0.788861	22.571074
generalizations-mu-goodman/shapeworld-set_reference	3.365556	0.823385	23.558893
nav-to-center/lexicon_size_11	2.805418	0.935139	18.383341
nav-to-center/lexicon_size_118	3.767532	0.635255	22.101847
nav-to-center/lexicon_size_17	3.186153	0.815521	19.096524
nav-to-center/lexicon_size_174	3.245330	0.609803	20.487443
nav-to-center/lexicon_size_25	2.804201	0.782212	17.290421
nav-to-center/lexicon_size_255	3.534679	0.678513	21.227163
nav-to-center/lexicon_size_37	3.028477	0.679117	19.004602
nav-to-center/lexicon_size_54	3.754792	0.691966	22.069917
nav-to-center/lexicon_size_7	2.758577	0.982625	16.908697
nav-to-center/lexicon_size_80	3.457586	0.674088	20.831957
nav-to-center/temperature_0.1	1.994309	0.997155	14.945156
nav-to-center/temperature_0.167	1.981753	0.990876	14.319155
nav-to-center/temperature_0.278	1.986637	0.598037	15.045198
nav-to-center/temperature_0.464	1.982692	0.991346	15.823868
nav-to-center/temperature_0.774	2.311150	0.823248	15.176164
nav-to-center/temperature_1.29	2.754878	0.869067	17.236170
nav-to-center/temperature_10	4.905167	0.817528	30.954056
nav-to-center/temperature_2.15	4.966695	0.827782	30.802945
nav-to-center/temperature_3.59	5.340638	0.890106	31.395475
nav-to-center/temperature_5.99	5.266347	0.877724	32.180539
rlupus/21-player.run-0	4.062520	0.924915	28942.558214
rlupus/21-player.run-1	4.196960	0.955523	30233.522552
rlupus/21-player.run-2	3.997152	0.910033	28830.352466
rlupus/9-player.run-0	3.079577	0.971498	1735.260160
rlupus/9-player.run-1	3.119583	0.984119	1290.840311
rlupus/9-player.run-2	3.090164	0.974838	1281.218984
rlupus/9-player.run-3	3.111235	0.981485	1292.071343

Table B.3

name	2-gram Entropy	2-gram Conditional Entropy
babyai-sr/GoToObj	1.544519	0.306888
babyai-sr/GoToObjLocked	1.147285	0.160295
babyai-sr/GoToObjLocked_ambiguous	1.978413	0.254393
babyai-sr/GoToObjLocked_ambiguous-freq_1	2.071162	0.607508
babyai-sr/GoToObjLocked_ambiguous-freq_2	1.538991	0.153070
babyai-sr/GoToObjLocked_ambiguous-freq_32	0.420175	0.062050
babyai-sr/GoToObjLocked_ambiguous-freq_4	2.406197	0.409242
babyai-sr/GoToObjLocked_ambiguous-msg_16	3.097571	0.542418
babyai-sr/GoToObjLocked_ambiguous-msg_32	1.463220	0.112660
babyai-sr/GoToObjLocked_ambiguous-msg_4	1.717505	0.795367
babyai-sr/GoToObjUnlocked	2.497606	0.504450
babyai-sr/GoToObjUnlocked-freq_1	1.092966	0.196541
babyai-sr/GoToObjUnlocked-freq_2	2.877898	0.761815
babyai-sr/GoToObjUnlocked-freq_32	1.731359	0.087790
babyai-sr/GoToObjUnlocked-freq_4	2.979210	0.814026
babyai-sr/GoToObjUnlocked-msg_16	3.043978	0.435771
babyai-sr/GoToObjUnlocked-msg_32	3.157215	0.216230
babyai-sr/GoToObjUnlocked-msg_4	2.307255	0.854029
corpus-transfer-yao-et-al/cc	2.059689	0.661383
corpus-transfer-yao-et-al/coco_2014	12.884451	6.285130
ec-at-scale/imagenet-10x1	6.811992	2.831113
ec-at-scale/imagenet-10x10	6.328754	2.420041
ec-at-scale/imagenet-1x1	6.882813	2.761016
ec-at-scale/imagenet-1x10	6.375876	2.400379
ec-at-scale/imagenet-5x5	7.137788	2.924591
egg-discrimination/4-attr_4-val_3-dist_0-seed	4.434835	1.438094
egg-discrimination/4-attr_4-val_3-dist_1-seed	3.550278	1.055580
egg-discrimination/4-attr_4-val_3-dist_2-seed	3.544613	0.979835
egg-discrimination/6-attr_6-val_3-dist_0-seed	3.917021	1.335551
egg-discrimination/6-attr_6-val_3-dist_1-seed	4.308021	1.420628
egg-discrimination/6-attr_6-val_3-dist_2-seed	3.738390	1.164541
egg-discrimination/6-attr_6-val_9-dist_0-seed	4.371053	1.509123
egg-discrimination/6-attr_6-val_9-dist_1-seed	3.578326	1.115826
egg-discrimination/6-attr_6-val_9-dist_2-seed	4.070906	1.320061
egg-discrimination/8-attr_8-val_3-dist_0-seed	3.504384	1.077631
egg-discrimination/8-attr_8-val_3-dist_1-seed	3.712531	1.156216
egg-discrimination/8-attr_8-val_3-dist_2-seed	4.006086	1.203946
egg-reconstruction/4-attr_4-val_10-vocab_10-len	3.212115	0.915787
egg-reconstruction/6-attr_6-val_10-vocab_10-len	3.750294	1.177051
egg-reconstruction/8-attr_8-val_10-vocab_10-len	3.515011	1.219244
generalizations-mu-goodman/cub-concept	5.686797	1.933852
generalizations-mu-goodman/cub-reference	5.641346	2.537465
generalizations-mu-goodman/cub-set_reference	5.509904	2.296366
generalizations-mu-goodman/shapeworld-concept	6.040857	2.814134
generalizations-mu-goodman/shapeworld-reference	5.908455	2.684016
generalizations-mu-goodman/shapeworld-set_reference	6.409305	3.043749
nav-to-center/lexicon_size_11	4.240224	1.434806
nav-to-center/lexicon_size_118	5.389004	1.621472
nav-to-center/lexicon_size_17	4.655472	1.469319
nav-to-center/lexicon_size_174	4.717891	1.472561
nav-to-center/lexicon_size_25	4.106729	1.302528
nav-to-center/lexicon_size_255	5.098629	1.563950
nav-to-center/lexicon_size_37	4.335838	1.307361
nav-to-center/lexicon_size_54	5.463441	1.708649
nav-to-center/lexicon_size_7	4.123176	1.364599
nav-to-center/lexicon_size_80	5.001934	1.544348
nav-to-center/temperature_0.1	3.405157	1.410848
nav-to-center/temperature_0.167	3.469046	1.487293
nav-to-center/temperature_0.278	3.396763	1.410126
nav-to-center/temperature_0.464	3.377160	1.394467
nav-to-center/temperature_0.774	3.777791	1.466642
nav-to-center/temperature_1.29	4.202502	1.447624
nav-to-center/temperature_10	8.121348	3.216181
nav-to-center/temperature_2.15	7.739814	2.773120
nav-to-center/temperature_3.59	8.433494	3.092856
nav-to-center/temperature_5.99	8.660965	3.394618
rlupus/21-player.run-0	6.956412	2.893892
rlupus/21-player.run-1	7.403071	3.206111
rlupus/21-player.run-2	7.039882	3.042730
rlupus/9-player.run-0	5.883233	2.803656
rlupus/9-player.run-1	5.925070	2.805487
rlupus/9-player.run-2	5.979073	2.888910
rlupus/9-player.run-3	5.865222	2.753987

Table B.4

Appendix C

Evaluation with Deep Transfer Learning

C.1 Hyperparameters

C.1.1 Causal language modeling

For values not listed, see Hugging Face Transformers' defaults at `https://huggingface.co/docs/transformers/v4.36.1/en//model_doc/gpt2#transformers.GPT2Config`.

- Model: GPT-2
- Tokenizer: Byte pair encoding
- Hidden size: 768 (default)
- Vocabulary size: 30 000
- Context length: 256
- Number of layers: 6
- Number of attention heads: 6
- Learning rate: $1 \cdot 10^{-4}$
- Optimizer: AdamW
- Weight decay: 0.01
- Learning rate schedule: linear (to 0)
- Batch size: 32
- Train dataset size: $15 \cdot 10^6$ tokens
- Train epochs: 5
- Tune dataset size: $2 \cdot 10^6$ tokens
- Train epochs: 10

C.1.2 Machine translation

For values not listed, see Hugging Face Transformers' defaults at `https://huggingface.co/docs/transformers/v4.36.1/en/model_doc/bart#transformers.BartConfig`. The following is for the *Full* setting.

- Model: BART
- Training objective: text infilling only (see note below)
- Tokenizer: Byte pair encoding
- Hidden size: 512

- Vocabulary size: 30 000
- Context length: 512
- Number of encoder layers: 6
- Number of decoder layers: 6
- Number of encoder attention heads: 8
- Number of decoder attention heads: 8
- Encoder feedforward dimension: 2048
- Decoder feedforward dimension: 2048
- Train learning rate: $1 \cdot 10^{-4}$
- Tune learning rate: $2 \cdot 10^{-4}$
- Optimizer: AdamW
- Weight decay: 0.01
- Learning rate schedule: linear (to 0)
- Batch size: 32
- Train dataset size: $100 \cdot 10^{6}$ tokens
- Train epochs: 5
- Tune dataset size: $50 \cdot 10^{6}$ tokens
- Train epochs: 3
- Test beam size: $1, 3, 5$ (final metric averaged across each size)
- Test context size: 128

The objective used to pretrain BART was text infilling *only*; we cannot use the sentence permutation objective because we do not know *a priori* what constitutes a sentence in an emergent language, hence we do not use it for any settings. For the *Frozen* setting, all is as above, but all non-embedding layers are frozen for the duration of tuning. For the *Reduced* setting, all is as above except for the following:

- Tune learning rate: $1 \cdot 10^{-5}$
- Tune dataset size: $10 \cdot 10^{6}$

C.1.3 Generic signaling game

We use the following hyperparameters for the *Disc, small* emergent language.

- Game (from EGG):
  `egg.zoo.basic_games.play`
- Message optimization: Gumbel-softmax (as opposed to REINFORCE)
- Game type: discrimination
- Number of attributes: 4
- Number of values: 4
- Number of distractors: 5
- Vocabulary size: 6
- Max message length: 10
- Number of examples: 32 768
- Batch size; 1024
- Number of epochs: 10
- Sender hidden size: 256
- Receiver hidden size: 512
- Sender embedding size: 32

- Receiver embedding size: 32
- Sender network type: GRU
- Receiver network type: GRU
- Learning rate: 0.001

The *Disc, large* setting uses the same hyperparameters as above with the exception of the following.

- Number of attributes: 12
- Number of values: 8
- Number of distractors: 5
- Number of examples: $3.5 \cdot 10^6$
- Max message length: 30
- Vocabulary size: 100
- Number of epochs: 100

The *Recon, large* setting is as in *Disc, large* with the following changes.

- Game type: reconstruction
- Number of attributes: 8
- Number of distractors: N/A
- Number of examples: $1 \cdot 10^6$
- Number of epochs: 10

C.2 Example of benchmark input format

The input format for the benchmark is simple: integer arrays in a JSON format separated by newlines (i.e., JSON Lines, JSONL, `*.jsonl`). The following is an example of file contents in this format:

```
[3, 1, 4, 1, 5, 9, 2]
[6, 5, 3, 5, 8, 9, 7, 9, 3]
[2, 3, 8, 4]
[6, 2, 6, 4, 3, 3]
[8, 3, 2, 7, 9, 5, 0, 2, 8, 8, 4]
```

C.3 Computing resources used

See Table C.1 for rough estimates of the compute used in writing this paper. Most experiments were run on a shared cluster comprising approximately 150 NVIDIA A6000 (or comparable) GPUs.

C.4 Additional results

C.4.1 BLEU scores for machine translation

See Table C.2.

Item	Base GH	n items	Total
XferBench	6	45	270
MT	8	50	400
Other experiments	2	50	100
Total			770

Table C.1: Estimate of compute used for this paper in GPU-hours (specifically NVIDIA RTX 2080 Ti–hours).

Source	Full	Frozen	Reduced
French	12.93	5.33	6.61
Spanish	13.32	4.52	6.35
Russian	12.93	4.37	7.02
Chinese	12.71	3.04	6.03
Korean	12.83	2.95	6.36
Arabic	13.12	4.16	6.74
Hindi	12.72	3.20	5.24
Paren, real	12.60	0.65	6.26
Paren, synth	13.19	0.82	6.15
Disc, large	12.93	2.08	4.44
Disc, small	0.17	0.19	0.38
Rec, large	1.92	0.86	2.50
Yao+	0.01	1.04	2.57
Mu+, SW	0.00	1.05	1.86
Mu+, CUB	12.71	1.45	2.35
Random	0.00	0.00	1.02
No pretrain	0.10	0.06	3.43

Table C.2: BLEU scores for machine translation experiment. Colors normalized by column.

C.4.2 Raw cross-entropies on XferBench

See Table C.3.

C.4.3 Writing system matrix for normalized XferBench scores

See Tables C.5 and C.6. Scores for reach writing system are aggregated by taking the mean. Table C.4 gives the writing system classification for the languages used in the experiments. Although the class imbalance makes it impossible to draw any definitive claims, the preliminary results do not suggest any correlation in XferBench between the writing systems of the source and target languages.

Source	Danish	Basque	Persian	Finnish	Hebrew	Indonesian	Japanese	Kazakh	Romanian	Urdu	*Mean*
French	4.93	6.03	5.04	5.62	5.48	4.87	5.23	5.46	5.15	4.43	5.22
Spanish	4.92	6.06	5.03	5.61	5.47	4.82	5.25	5.46	5.12	4.42	5.22
Russian	4.94	6.04	5.04	5.65	5.48	4.88	5.27	5.48	5.14	4.45	5.24
Chinese	4.89	6.02	5.01	5.58	5.43	4.76	5.18	5.44	5.12	4.39	5.18
Korean	4.89	6.01	5.02	5.57	5.44	4.78	5.20	5.45	5.12	4.38	5.19
Arabic	4.90	6.02	5.02	5.59	5.45	4.81	5.22	5.44	5.13	4.40	5.20
Hindi	4.94	6.06	5.08	5.65	5.47	4.83	5.29	5.52	5.20	4.46	5.25
Paren, real	5.07	6.11	5.11	5.75	5.59	5.06	5.38	5.57	5.22	4.56	5.34
Paren, synth	5.08	6.13	5.14	5.74	5.58	5.09	5.43	5.58	5.26	4.57	5.36
Disc, large	5.00	6.06	5.11	5.71	5.52	4.92	5.34	5.56	5.25	4.49	5.30
Disc, small	5.09	6.06	5.17	5.80	5.59	5.05	5.41	5.65	5.31	4.56	5.37
Rec, large	5.09	6.06	5.16	5.79	5.57	5.04	5.41	5.64	5.30	4.55	5.36
Yao+	5.07	6.03	5.17	5.79	5.56	5.03	5.41	5.65	5.31	4.56	5.36
Mu+, SW	5.09	6.10	5.18	5.80	5.58	5.05	5.42	5.65	5.33	4.58	5.38
Mu+, CUB	5.08	6.06	5.18	5.79	5.58	5.05	5.42	5.65	5.32	4.56	5.37
Random	5.23	6.17	5.31	5.92	5.71	5.22	5.55	5.76	5.45	4.72	5.50
No pretrain	5.17	6.10	5.23	5.85	5.66	5.14	5.47	5.68	5.38	4.65	5.43
Mean	5.02	6.07	5.12	5.72	5.54	4.96	5.35	5.57	5.24	4.51	5.31

Table C.3: Cross-entropies across all source and target languages. Colors normalized by column.

Type	Writing System	Language
Abjad	Arabic	ar fa ur
	Hebrew	he
Abugida	Devanagari	hi
Alphabet	Cyrillic	kk ru
	Hangul	ko
	Latin	da es eu fi fr id ro
Logographic	Chinese	zh
Mixed	Japanese	ja

Table C.4: Coarse and fine classifications of writing systems of human languages (source and target) used in the experiments.

Source	Arabic	Cyrillic	Hebrew	Japanese	Latin
Arabic	−0.65	−0.81	−0.42	−0.35	−0.56
Chinese	−1.05	−0.93	−1.60	−1.46	−1.00
Cyrillic	0.63	0.68	1.07	0.90	0.78
Devanagari	1.62	1.93	0.39	1.47	1.23
Hangul	−0.98	−0.41	−0.89	−0.80	−1.04
Latin	0.22	−0.22	0.72	0.12	0.29

Table C.5: Normalized XferBench scores by writing system (lower is better). Color is normalized across all values.

Source	Abjad	Alphabet	Mixed
Abjad	−0.57	−0.60	−0.35
Abugida	1.21	1.35	1.47
Alphabet	0.15	0.06	0.09
Logographic	−1.23	−0.99	−1.46

Table C.6: Normalized XferBench scores by writing system type (lower is better). Color is normalized across all values.

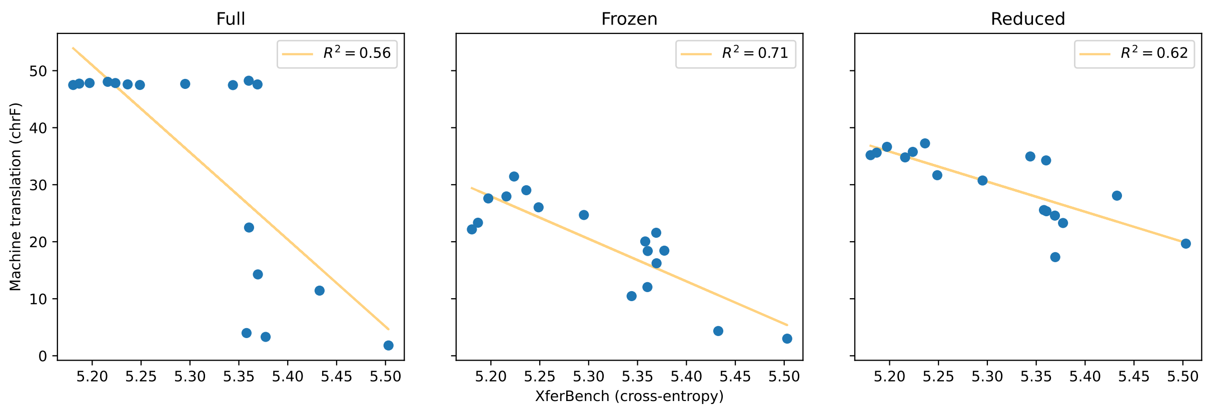


Figure C.1: Scatter plots showing XferBench score versus machine translation score.

C.4.4 Scatter plots for XferBench and MT

See Figure C.1.

C.5 Cross-entropy confidence interval computation

Let $s \in S$ and $t \in T$ represent source and target languages, respectively. $h_{s,t}$ represents the test cross-entropy of a model pretrained on s and evaluated on t. As sated in Equation (4.1), the score on XferBench is the mean cross-entropy across all target languages:

$$h_s = \operatorname*{mean}_{t' \in T} \left(h_{s,t'} \right). \tag{C.1}$$

We would like to calculate a confidence interval (i.e., h_s^- and h_s^+) for a source language's mean cross-entropy using the different cross-entropies on the target languages (i.e., $h_{s,t}$ for $t \in T$), yet these samples are not i.i.d., since the mean of cross-entropy each *target* language can vary. Thus, if we would like to use bootstrapping to calculate confidence intervals, we must first normalize the cross-entropies. Let $\hat{h}_{s,t}$ be the normalized score:

$$\hat{h}_{s,t} = \frac{h_{s,t} - \operatorname{mean}_{s' \in S} \left(h_{s',t} \right)}{\operatorname{stdev}_{s' \in S} \left(h_{s',t} \right)}. \tag{C.2}$$

Given the normalized scores, we can now bootstrap in order to compute confidence intervals for $\hat{h}_s$ (i.e., in the normalized space).[1] Let $\hat{h}_s^+$ and $\hat{h}_s^-$ be the upper and lower bounds of the confidence interval computed using bootstrapping in the normalized space. We can now translate these back into the raw cross-entropy space using the means and standard deviations from before:

$$h_s^+ = \hat{h}_s^+ \cdot \operatorname*{stdev}_{s' \in S} \left(h_{s',t} \right) + \operatorname*{mean}_{s' \in S} \left(h_{s',t} \right) \tag{C.3}$$

$$h_s^- = \hat{h}_s^- \cdot \operatorname*{stdev}_{s' \in S} \left(h_{s',t} \right) + \operatorname*{mean}_{s' \in S} \left(h_{s',t} \right). \tag{C.4}$$

C.6 Error analysis

In the *Full* setting of the machine translation task, the *Yao+* and *Mu+, SW* settings perform worse than expected (*a priori* and compared to the other results in the setting). Validation loss converged while chrF and BLEU scores remained near zero. We provide a couple examples (taken from the predefined test set of WMT 2014) of model output to provide some insight into the reason for this. No post processing used, generation is capped at 50 tokens, and "\u0000" represent single non-printable characters.

Example 1 *Input*: "And while Congress can't agree on whether to proceed, several states are not waiting."

Reference: "Et tandis que les membres du Congrès n'arrivent pas à se mettre d'accord pour savoir s'il faut continuer, plusieurs États n'ont pas attendu."

[Model pretrained on] *French*: "#Et alors que le Congrès ne peut pas convenir de poursuivre, plusieurs États ne sont pas en attente. » (traduction libre) Le Parlement européen. Le Parlement européen est d'avis que le Parlement européen doit être en mesure de faire preuve#"

Disc, large: "#Et bien que le Congrès ne puisse pas convenir de la marche à suivre, plusieurs États ne sont pas en attente.\u2028\u2028[Traduit par la Rédaction]\u2028(Traduit par la Rédaction)\u2028(Tra#"

[1] This is not intended to be statistically rigorous. Our cross-entropies are unlikely to be normally distributed, but this still be helpful for generally gauging uncertainty.

Yao+: "#Annexe II, par.\xa02.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.#"
Mu+, SW: "#Annexe II.\xa0\xa0: Appendice I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I#"

Example 2 *Input*: "This really is a must for our nation."
Reference: "Cela est vraiment indispensable pour notre nation."
French: "#C'est vraiment une nécessité pour notre nation. Nous devons y parvenir. Nous devons y parvenir. Nous devons y parvenir. Nous devons y parvenir. Nous devons y parvenir. Nous devons y parvenir. Nous devons y parvenir. Nous devons y#"
Disc, large: "#C'est vraiment un devoir pour notre nation. C'est un devoir.\u2028 \u2028(...)\u2028\u2028(...)\u2028(...)\u2028(...)\u2028(...)\u2028(...)\u2028(...)\u2028(#"
Yao+: "#Annexe II, par.\xa02.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.#"
Mu+, SW: "#Annexe II.\xa0\xa0: Appendice I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I.\xa0I#"

Discussion Although all of the models have trouble terminating properly, the *French* and *Disc, large* models (which have high chrF scores) clearly condition their generation on the text, whereas *Yao+* and *Mu+, SW* give the same output regardless of the input. Although this is unexpected, we can see in the *Full* setting in Figure C.1 that there is sharp drop off between high-performing and low-performing languages. We suspect that the higher learning rate during tuning caused this bimodal distribution of results and is at least in part responsible for the poor performance *Yao+* and *Mu+, SW* models on the MT experiment's *Full* setting.

Appendix D

Optimizing for Statistical Similarity to Human Language

D.1 Correlation of Evaluation Languages

One of XferBench's chief weaknesses is its long runtime, taking 2 to 6 hours depending on the GPU used. Approximately 30% of that time is spent on the initial pretraining with the emergent language corpus, with the other 70% spent on finetuning and testing on the 10 downstream languages. We observe from the XferBench scores on the emergent languages of ELCC and the human language baselines of Chapter 4 that 9 out of the 10 evaluation languages are highly correlated with each other, that is, the XferBench score on one language is highly predictive of the overall XferBench score. In particular, test cross-entropy on Danish (da) alone can predict >95% of the variation of the overall XferBench score (i.e., the linear regression has an $R^2 > 0.95$). For this reason, in the hyperparameter optimization trials, we compute XferBench-da (XferBench evaluated on Danish only) which is around 3× faster than the full XferBench; the final evaluation nevertheless uses the full set of evaluation language

	All	Human	Emergent
Basque	0.340	0.685	0.318
Danish	0.992	0.966	0.987
Finnish	0.971	0.968	0.969
Hebrew	0.967	0.967	0.977
Indonesian	0.988	0.952	0.983
Japanese	0.973	0.930	0.974
Kazakh	0.983	0.936	0.977
Persian	0.972	0.951	0.971
Romanian	0.985	0.945	0.982
Urdu	0.951	0.849	0.929

Table D.1: R^2 values for individual target XferBench languages predicting the full XferBench score. *Human* and *Emergent* refer to the R^2 value considering only the human or emergent languages, respectively.

for XferBench.

In Table D.1, we show the R^2 values derived from training a linear model on just one of the target language's XferBench scores to predict the overall XferBench score. The emergent languages are all of the corpora from ELCC (Chapter 3), and the human language corpora are the baselines from the original XferBench paper (Chapter 4). R^2 value corresponds to the percent of the variance in the full XferBench score explained by just the score (i.e., cross-entropy) on that particular target language. We find, strikingly enough, that all of the target languages, with the exception of Basque, are highly correlated, having R^2 values above 0.95 all languages, and greater than 0.80 even when considering human languages alone. Danish, of all of the languages, has the highest R^2 value (>0.99), which is the reason we select it as the sole target for a more time-efficient variant of XferBench (which we term XferBench-da).

D.2 Hyperparameters Not Discussed

In this section we briefly discuss hyperparameters that were tried but not not documented in the paper or that were not investigated at all. We selected a batch size of 32 based on comparing the compute efficiency of different sizes. Larger batch sizes could process more data faster but would not update the parameters often enough. On the other hand, smaller batch sizes would not process enough data to maximize the utility of each update. Mixed precision training was tested but not found to improve runtime. For learning rate scheduling, we found cosine annealing to be slightly more effective than no learning, but further schedules were not investigated. Weight decay was investigated in earlier experiment but found not to have a noticeable effect.

The implementation of the signaling game we used could also be optimized using REINFORCE to handle the discrete message, but we only tested with a Gumbel-Softmax layer as it is faster and more stable to optimize with. We did not vary the neural architecture beyond altering the number of units in the hidden and embedding layers; for example, we did not add additional layers, try different RNN cells (e.g., LSTM), or use transformers.

D.3 Full Table of Hyperparameters

In Table D.2, we show all of the hyperparameters selected for the searches and trials referenced in the paper.

D.4 Computing Resources Used

Experiments were performed across about 20–30 NVIDIA A6000 (or equivalent) GPUs (one trial per GPU) on an institutional cluster. We estimate approximately 5500 GPU-hours were used for all experiments directly related to this paper, including those not documented or directly referenced. The primary searches for the best-performing emergent languages on XferBench (Searches 1–4) took about 1300 GPU-hours.

#	|Trials|	|Attrs.|	|Vals.|	|Distrs.|	Temp.	|Embed.|	|Hidden|	LR	|Vocab|	Length	|Epochs|
1	578	$[3, 7]$	$[3, 7]$	$[1, 127]$	$[0.1, 10]$	$[8, 128]$	$[8, 128]$	[500µ, 50m]	[10, 20k]	$[1, 40]$	500
2	171	$[5, 10]$	$[5, 10]$	—	$[0.5, 4]$	$[64, 512]$	$[64, 512]$	[500µ, 5m]	[300, 30k]	—	—
3	140	—	—	—	—	—	—	—	—	—	[500, 5k]
4	282	$[6, 20]$	6	23	2	128	256	[1m, 3m]	[500, 30k]	—	—
4.1	1	11	6	—	—	—	—	1.79m	9721	16	1715
4.2	1	12	6	—	—	—	—	1.86m	12496	22	1593
4.3	1	13	6	—	—	—	—	1.74m	8096	18	1511
5r	411	$[4, 20]$	$[3, 10]$	$[1, 127]$	$[0.1, 10]$	$[8, 512]$	$[8, 512]$	[500µ, 10m]	[2, 30k]	$[1, 40]$	[10, 3k]
6e	109	10	10	$[63, 511]$	2	32	32	2.7m	25k	15	5k
6e.1	1	—	—	228	—	—	—	—	—	—	—
6e.2	1	—	—	372	—	—	—	—	—	—	—
6e.2	1	—	—	165	—	—	—	—	—	—	—

Table D.2: All hyperparameters were treated as log-scale hyperparameters. $|\cdot|$ refers to cardinality. "—" means unchanged from the previous run. µ, m, and k refer to the SI prefixes micro ($\times 10^{-6}$), milli ($\times 10^{-3}$), and kilo ($\times 10^{3}$), respectively. 4.1 is the best-performing trial of Search 4 (and likewise for 4.2, 6e.1, etc.).

D.5 Synthetic Languages

D.5.1 Definitions

We use four probabilistic synthetic languages which span a large portion of the Chomsky hierarchy ranging from trivial to beyond context-free. All synthetic languages contain a unique begin- and end-of-sentence token in each utterance.

Zipf-Mandelbrot Distribution The basis for our synthetic languages will be a Zipf–Mandelbrot distribution, a generalization of Zipf's law, where the unnormalized probability weight of the word w_i is

$$f(w_i) = \frac{1}{(i + \beta)^{\alpha}}, \tag{D.1}$$

where i is the 1-based index of the word, α controls the weight of the tail, and β shifts where the distribution starts (roughly speaking). Empirically, $\alpha = 1$ and $\beta = 2.7$ have been found to be good approximations for human language and will be the default parameters of the distribution unless otherwise specified (Piantadosi, 2014).

Bag of Words The simplest synthetic language we introduce is a bag-of-words language where each token in a sentence is sampled independently from the Zipf-Mandelbrot distribution. The length of the sentence is independent of the sampling method, so in interest of simplicity, we sample from a discrete uniform distribution.

Regular The simplest non-trivial language we introduce is a regular language which partitions the tokens uniformly at random into k different sets $(s_1, \ldots, s_k)$, keeping their initial Zipf–Mandelbrot-derived weight. Each sentence starts with a token sampled from s_1; each subsequent token is sampled from the next class $(s_i + 1)$ with probability c or sampled from the same class (s_i). After s_k, the sentence terminates. Thus, the language is defined by the regular expression

$$s_1^+ s_2^+ \dots s_k^+, \tag{D.2}$$

where $a^+ = aa^*$, s_i represents any token in the set s_i, and appropriate BoS and EoS tokens are added.

Dyck-n Dyck-n can be thought of as "balanced nested delimiters" (where the delimiters are the same token) (Schützenberger, 1963). Each token in the sentence is generated as follows: With probability p, a new token is sampled from the Zipf–Mandelbrot distribution and pushed onto a stack (the "opening delimiter"), and with probability $1-p$, the token on top of the stack is popped off. A sentence always begins with an "open" token and ends when the stack is empty. An example of such a sentence is $(3, 1, 1, 2, 1, 1, 2, 3)$ which could be illustrated as "{()[()]}".

Shuffle Dyck-n Finally, we use Shuffle Dyck-n as our last language which lies beyond context-free in the Chomsky hierarchy Suzgun et al. (2019). Technically speaking, this language should be called Shuffle of n Distinct Dyck-1 Languages since it is the result of randomly interleaving multiple Dyck-1 languages with distinct tokens. To generate a sentence in Shuffle Dyck-n, we first follow the same procedure as for Dyck-n but keep the individual tokens separate. We then interleave the separate strings by appending to the sentence uniformly at random from one of the individual strings until they are empty. For example, if Dyck-n generated "{([()])[]}", the separated strings would "{}", "(())", and "[][]", which could then be interleaved into "{[}(()])".

D.5.2 Hyperparameters

Each variation of the synthetic language maintains the default values while varying a single hyperparameter. We vary the common hyperparameters as follows:

Vocabulary size takes the values 10, 100, 1k, 5k, 10k, 30k (default: 30k). A vocab size of 10 is incompatible with the Regular language and was skipped.

Zipf–Mandelbrot α takes the values 0, 0.25, 0.5, 1, 2, and 4 (default: 1).

***n* tokens** (in the whole corpus) takes the values 1k, 10k, 100k, 1M, 5M, and 15M (default: 15M); this hyperparameter was not varied for the Unigram language.

The Unigram language has an additional hyperparameter stop probability which takes the values 0.05, 0.1, and 0.2 (default: 0.1). The Regular language has two additional hyperparameters: repeat probability (c) which takes the values 0.2, 0.4, 0.5, and 0.6 (default: 0.4), and n classes which takes the values 5, 10, 20, and 40 (default: 10). The Dyck and Shuffle Dyck languages take the additional hyperparameter open probability with values: 0.2, 0.3, 0.4, 0.5, and 0.6 (default: 0.5); Shuffle Dyck is not generated with the value 0.6 due to implementation constraints.

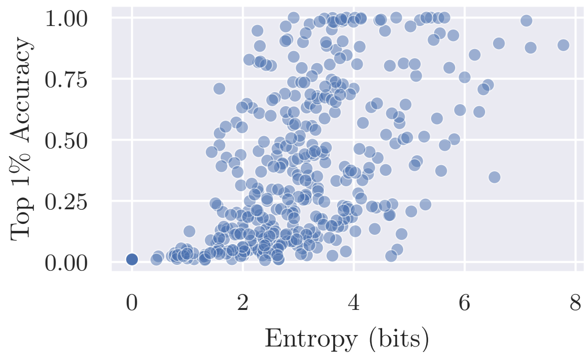


Figure D.1: Entropy versus accuracy for Search 5r.

D.6 Task Success and Entropy

Previous work (Kharitonov et al., 2020b; Chaabouni et al., 2021) has analyzed entropy minimization with respect to the amount of information or, roughly speaking, task success. We performed a brief analysis the relationship between entropy and accuracy (task success) shown in Figure D.1. While we do find significant correlation (Pearson's $r = 0.57$ for Search 5r), we would not characterize it as any strict sort of entropy minimization. That is, we observe many emergent languages which are from the Pareto frontier of high accuracy and low entropy. Hyperparameter search demonstrates itself to be a powerful tool for investigating such correlations since it is able to generate a wide variety of emergent languages with minimal additional work from the researchers. Nevertheless, more investigation would have to be done on this front to conclusively support or reject prior claims of entropy minimization.

D.7 Hyperparameter Scatter Plots

Figures D.2 to D.5 show the univariate scatter plots for hyperparameter Searches 1–4. The y-axis is XferBench-da score (or some smaller variation thereof, for Searches 1 and 2), and the x-axis is one of the hyperparameters varied for that search. Note that other variables are *not* held constant while one is varied; instead all hyperparameters are varied for each trial.

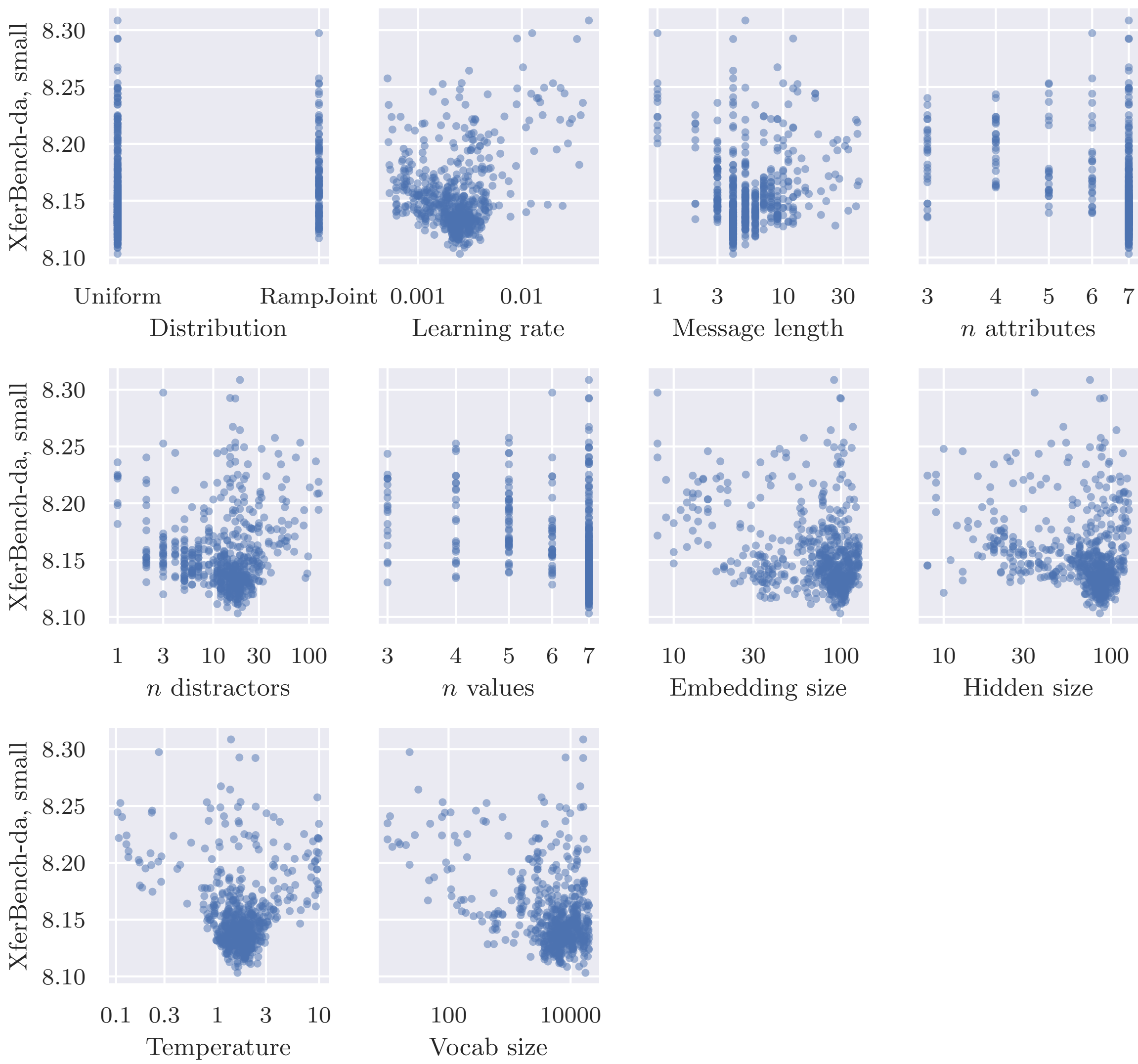


Figure D.2: Objective values for Search 1 by individual hyperparameter.

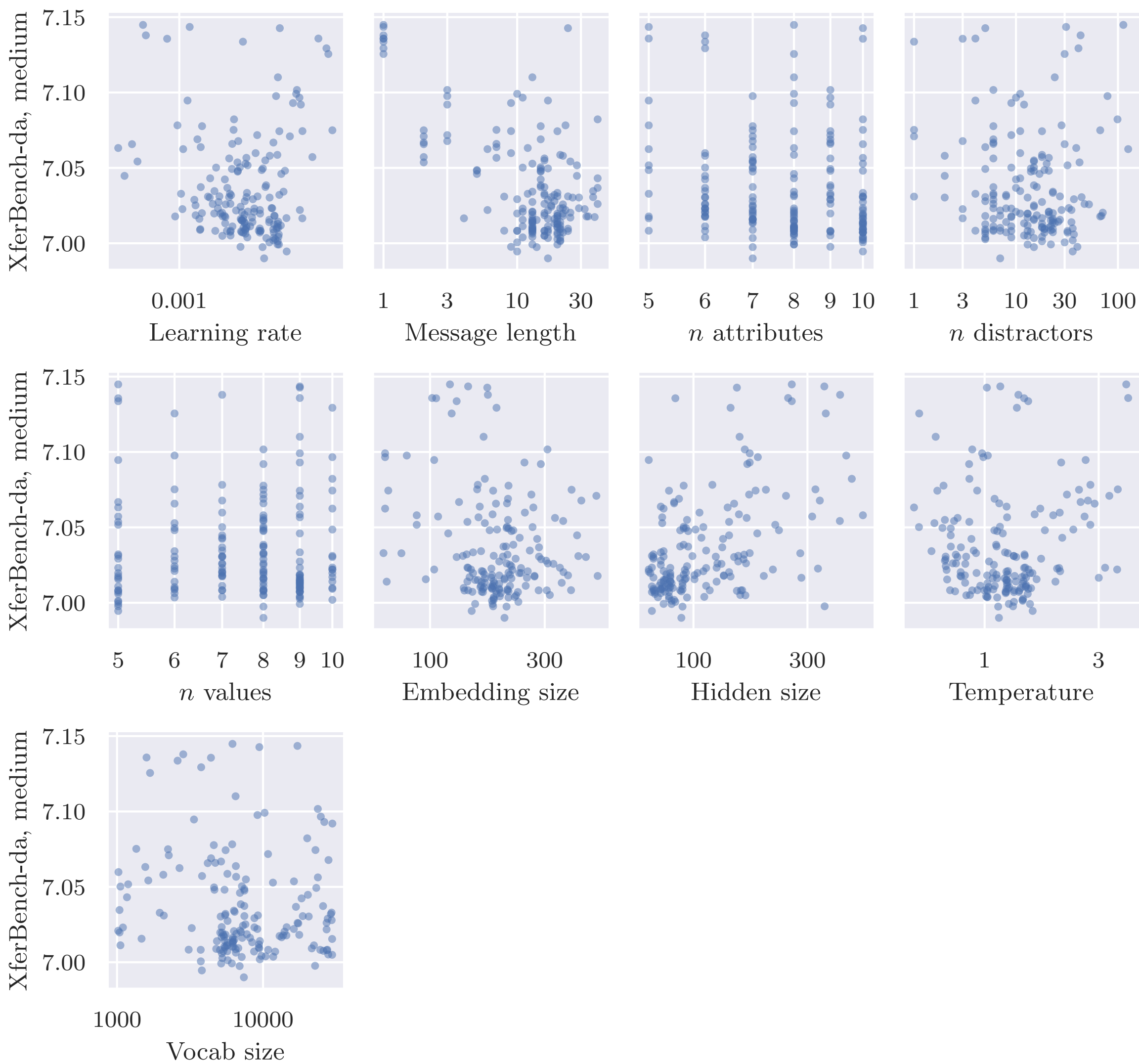


Figure D.3: Objective values for Search 2 by individual hyperparameter.

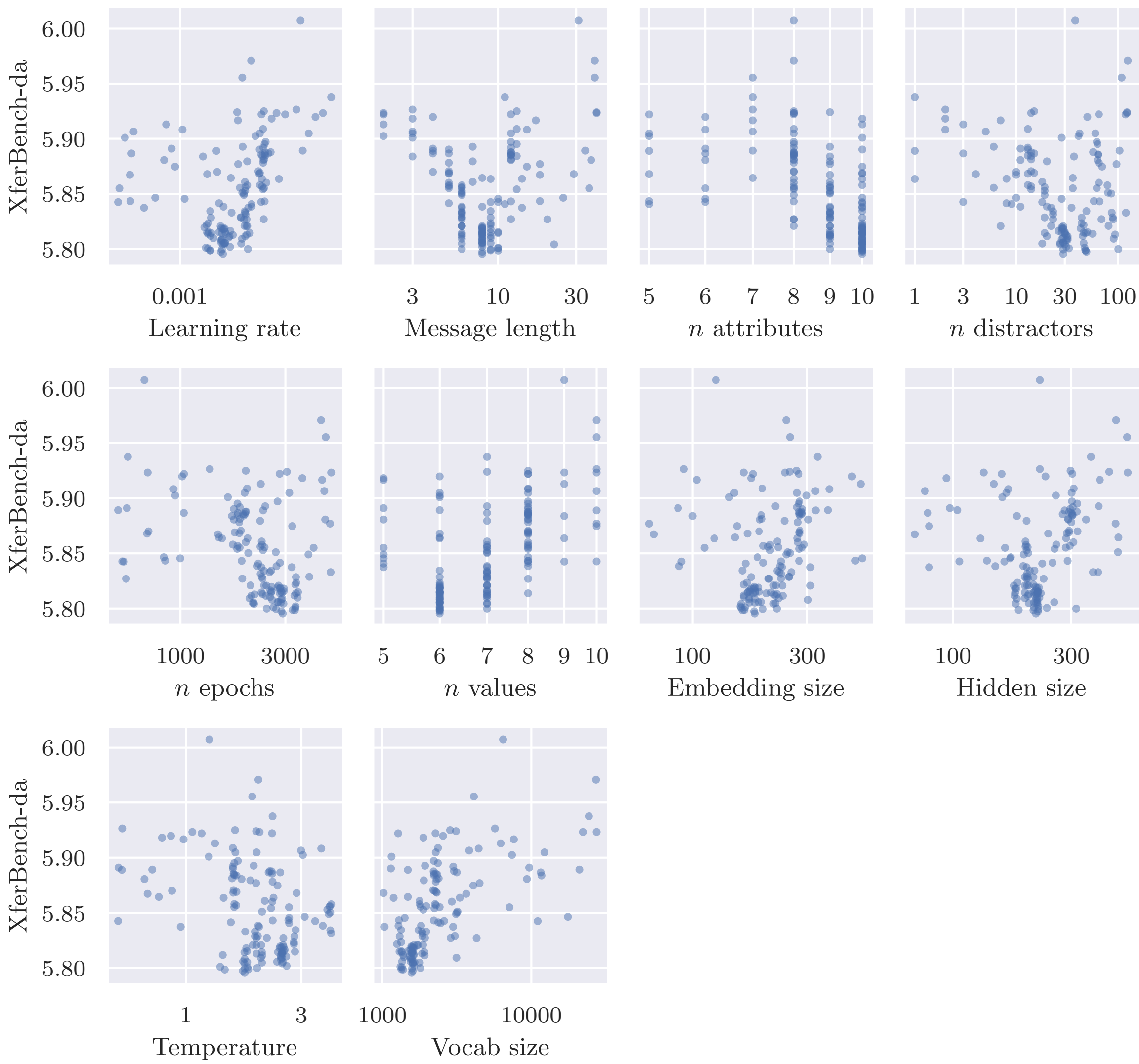


Figure D.4: Objective values for Search 3 by individual hyperparameter.

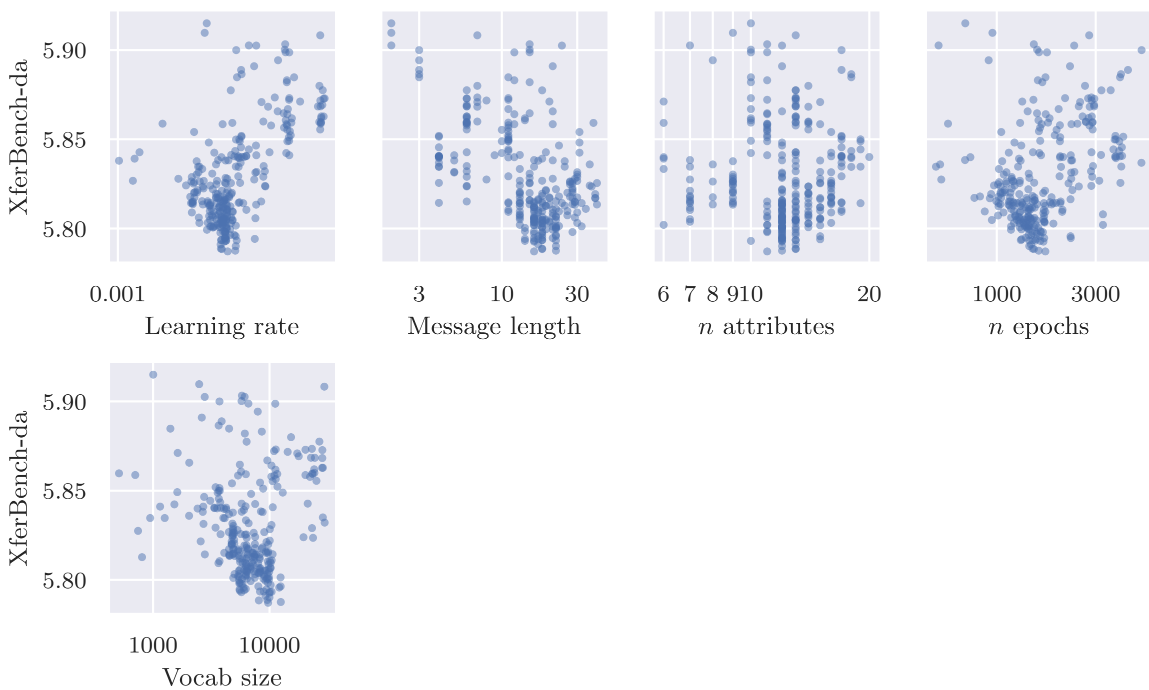


Figure D.5: Objective values for Search 4 by individual hyperparameter.

Appendix E

Discovering Morphemes in Rich Corpora

E.1 Algorithm

E.1.1 Candidate generation

For simplicity's sake (and inductive bias), we limit the candidate generation functions to all non-empty substrings for forms and all non-empty subsets for meanings. Nevertheless, we could extend form candidate generation to non-contiguous forms to detect non-concatenative morphology (e.g., the form "x.z" matching "xyz" and "xwz"). In fact, we could could use arbitrary regular expressions to represent forms (or meanings) such as "^..x" or "x+" to represent absolute position and optional repetitions, respectively. We could consider empty forms and empty meanings to explicitly identify forms and meanings which do not have mappings (as opposed to implicitly not including them in the morphology).

Of course, part of the difficulty of extending the complexity of the candidate generation is that it expands the already (sometimes intractably) large search space. One method of making this tractable, though, is adding heuristics that determine which form candidates should be considered rather than considering every possible candidate.

E.1.2 Ambiguous pair application

In some cases of applying a morpheme to record in the dataset, there are multiple applications possible. Say we have the utterance "x y z x y" meaning $\{A, B\}$ and we want to apply the morpheme ("x y", $\{A\}$). The form matches two substrings in the utterance, so there are two possible ways to apply the morpheme. As a heuristic for selecting the best application, CSAR break ties by selecting the substring least likely to be a morpheme (as determined by the morpheme weights). Going back to the above example, if it is the case the morpheme ("z x y", $\{B\}$) has a higher weight than ("x y z", $\{B\}$), then CSAR will apply ("x y", $\{A\}$) to the first instance of "x y" instead of the second.

This search can be very computationally expensive since it can entail going through a large number of morpheme candidates. Thus for the experiments with human language data, we do not perform this search and select the best form quasirandomly.

E.1.3 Heuristic optimizations

Below we include a summary of heuristic optimizations available in CSAR:

max input records Only consider a certain number of records from the input data; 20 000 for machine translation, image captions, and ShapeWorld.

max inventory size Stop after inducing a certain number of morphemes; 300 for image captions and machine translation settings.

***n*-gram semantics** Treat complete meanings as ordered and generate meaning candidates identically to forms (i.e., as n-grams); used for machine translation data where the "meanings" are sentences.

max form/meaning size Only consider form/meaning candidates up to a certain size; 3 for machine translation (form and meaning) and image captions (form only), 2 for image captions meaning.

no search best form When ablating a form with multiple matches in an utterance, do not search for best form, simply choose it randomly; no search for image captions and machine translation.

form/meaning vocabulary size Only consider the most common form/meaning candidates; 100 000 for image captions and machine translation.

token vocabulary size Only consider the most common form/meaning tokens and ignore an form meaning candidates which contain an unknown token; 1000 for image captions and 500 for machine translation.

co-occurrence threshold Zero out any co-occurrences which fall below a certain threshold (e.g., if a form and meaning candidate only occur once, treat it as never co-occurring); 1 for ShapeWorld, 10 for image captions, and 100 for machine translation.

E.2 Empirical Validation

E.2.1 Procedural dataset hyperparameters

The following hyperparameters were used for generating the procedural datasets. Each dataset uses 4 attributes and 4 values except for the sparse setting which uses 8 independent values.

Synonymy $\{1, 3\}$; forms per meaning

Polysemy $\{0, 0.15\}$; proportion of meanings mapped to an already-used form

Multi-token forms $\{\{1\}, \{1, 2, 3, 4\}\}$; possible tokens per form

Vocab size $\{10, 50\}$; only applies to non-unity multi-token forms

Sparse meanings $\{\text{true}, \text{false}\}$

Distribution imbalance $\{\text{true}, \text{false}\}$; non-uniform distribution is based on the ramp function, i.e., probability of given value for an attribute is proportional to its index $+ 1$.

Dataset size $\{50, 500\}$

Noise forms $\{0, 0.5\}$; $1 - p$ of parameter of geometric distribution

Shuffle form $\{\text{true}, \text{false}\}$

Non-compositionality $\{\text{true}, \text{false}\}$

Random seeds 3 per hyperparameter setting

	CSAR	IBM Model 1	IBM Model 3	Morfessor	BPE	ULM	Records
Exact F_1, form	0.868	0.616	0.595	0.827	0.624	0.670	0.133
Fuzzy F_1, form	0.960	0.899	0.893	0.949	0.890	0.891	0.637
Fuzzy prec., form	0.954	0.855	0.850	0.933	0.852	0.853	0.597
Fuzzy recall, form	0.967	0.952	0.946	0.967	0.934	0.938	0.701
Exact F_1	0.788	0.375	0.379	0.000	0.000	0.000	0.101
Fuzzy F_1	0.899	0.721	0.726	0.000	0.000	0.000	0.441
Fuzzy prec.	0.881	0.641	0.640	0.000	0.000	0.000	0.390
Fuzzy recall	0.921	0.855	0.866	0.000	0.000	0.000	0.543

Table E.1: Results of baseline methods on the procedural datasets.

Non-unity polysemy and synonymy rates for the non-compositional dataset implementation were not implemented and are excluded from the above grid.

E.2.2 Tokenizer vocabulary size

The heuristic for the tokenizer vocabulary size is as follows:

$$|V| = \left\lfloor \frac{|\mathcal{T}_{\text{meaning}}|}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \frac{|r_{\text{form}}|}{|r_{\text{meaning}}|} \right\rfloor + |\mathcal{T}_{\text{form}}|, \tag{E.1}$$

where $\mathcal{T}_{\text{meaning}}$ is the set of all meaning tokens in the dataset (likewise for $\mathcal{T}_{\text{form}}$), $\mathcal{R}$ is the multiset of records in dataset, r_{form} is the particular form (utterance) for an individual record (likewise for r_{meaning}. This heuristic can be interpreted as the mean form tokens per meaning tokens times the number unique meaning tokens added to the number of unique form tokens (since each of them will automatically be included in the vocabulary).

E.2.3 Additional procedural dataset results

Table E.1 shows all results of baseline methods on the procedural datasets. Figure E.1 visualizes the results of the baseline methods with exact F_1 score.

E.3 Analysis of Emergent Languages

E.3.1 Emergent language hyperparameters

Due to the computational constraints for some of the environments, we stop CSAR after inducing the top 200 morphemes for each language as well as disabling the lookahead substitution heuristic. Additionally, for the CUB (natural images) environment, the following optimization were employed: limiting to the maximum meaning size considered to 3, pruning morpheme candidates that had a prevalence $\leq$1%, and consider only the 1000 most frequent forms and 1000 most frequent meanings.

The following hyperparameters were used for the vector observation environment:

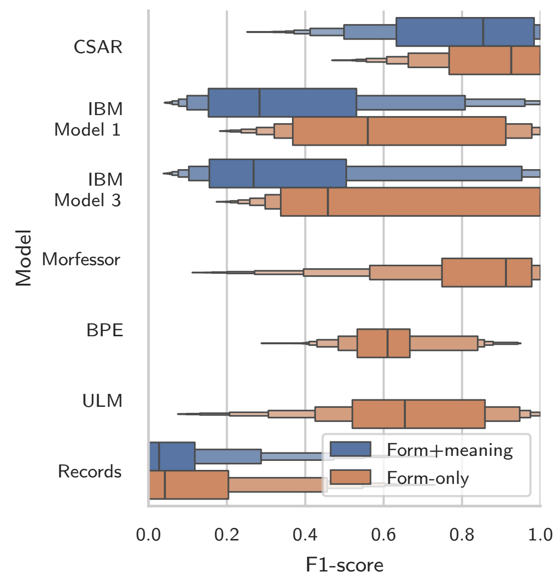


Figure E.1: Exact F_1 scores of baseline methods on the procedural datasets

***n* values** 4, 2 (sparse)
***n* attributes** 4, 8 (sparse)
***n* distractors** 3
vocab size 32
max sequence length 10
dataset size (CSAR input) 10 000 records

The ShapeWorld observation environment uses the following hyperparameters:

observations 5 shapes, 6 colors, 3 operators (and, or, not); *and* or *or* may only be used once
***n* examples** 20 total; 10 correct targets, 10 distractors
vocab size 30 (not including beginning- and end-of-sentence tokens)
max sequence length 8
dataset size (CSAR input) 1 200 records

The CUB observation environment uses the following hyperparameters (same as ShapeWorld unless noted):

observations 200 classes of birds with 40–60 images each; 312 binary attributes derived from the images
vocab size 18
***n* examples** 10 total
vision backbone ResNet-18

Both environments had any beginning-of-sentence and end-of-sentence tokens removed before being fed into CSAR.

	|Inv.|	|Form|
Vector, AV	94	3.59
Vector, sparse	126	3.51
SW, ref	2898	6.93
SW, setref	2920	8.13
SW, concept	1565	7.77

Table E.2: Metrics for form-only morpheme inventories generated by Morfessor across various emergent languages.

E.4 Morfessor Results on Emergent Language

In Table E.2 we show the results of running Morfessor on various emergent language corpora. Compared to the metrics for CSAR's output on the same corpora (Table 6.1), Morfessor's results do not match or even differ consistently (although Morfessor's forms do not have prevalence weighting like CSAR's). For the vector environments, Morfessor yields smaller inventories than CSAR yet larger inventories for ShapeWorld. Form lengths are similar for the vector environment, but for ShapeWorld, CSAR yields shorter forms than the vector environment while Morfessor yields much longer forms. Since we do not have ground truth morphemes for these emergent language corpora, we cannot definitively say one algorithm has performed better than the other. Yet Morfessor here is at a disadvantage here as it is not able to use the meanings of the utterances to guide its induction.

E.5 Morpheme Inventories

Top 100 morphemes induced by CSAR from human and emergent language datasets.

E.5.1 Human languages

Morpho Challenge (“’”, {+GEN}) (“ing$”, {+PCP1}) (“ed$”, {+PAST}) (“s”, {+PL}) (“er”, {er_s}) (“ly$”, {ly_s}) (“s$”, {+3SG}) (“ist”, {ist_s}) (“iz”, {ize_s}) (“ness”, {ness_s}) (“ion”, {ion_s}) (“ˆre”, {re_p}) (“ˆde”, {de_p}) (“ation”, {ation_s}) (“est$”, {+SUP}) (“ˆun”, {un_p}) (“less”, {less_s}) (“ful”, {ful_s}) (“ˆmis”, {mis_p}) (“head”, {head_N}) (“way”, {way_N}) (“ment”, {ment_s}) (“al”, {al_s}) (“it”, {ity_s}) (“ˆfire”, {fire_N}) (“ency$”, {ency_s}) (“hook”, {hook_N}) (“ish$”, {ish_s}) (“mind”, {mind_N}) (“ˆin”, {in_p}) (“at”, {ate_s}) (“if”, {ify_s}) (“able$”, {able_s}) (“ically$”, {ally_s}) (“ˆinter”, {inter_p}) (“ˆphoto”, {photo_p}) (“ˆhand”, {hand_N}) (“ˆscho”, {school_N}) (“house”, {house_N}) (“ical$”, {ical_s}) (“hold”, {hold_V}) (“long”, {long_A}) (“work”, {work_V}) (“up”, {up_B}) (“ag”, {age_s}) (“ant”, {ant_s}) (“ib”, {ible_s}) (“line”, {line_N}) (“ed$”, {ed_s}) (“er$”, {+CMP}) (“ˆover”, {over_p}) (“ˆdis”, {dis_p}) (“ˆsea”, {sea_N}) (“ˆim”, {im_p}) (“or”, {or_s}) (“pos”, {pose_V}) (“ence”, {ence_s}) (“ˆcardinal”, {cardinal_A}) (“ˆrational”, {rational_A}) (“ˆshoplift”, {shop_N}) (“conciliat”, {conciliate_V}) (“ˆmanicur”, {manicure_N}) (“ˆpredict”, {predict_V}) (“dressing”, {dressing_V}) (“ˆbuffet”, {buffet_V}) (“ˆcrimin”, {crime_N}) (“ˆentitl”, {entitle_V}) (“ˆfrivol”, {frivolous_A}) (“ˆheartb”, {heart_N}) (“ˆmaroon”,

{maroon_A}) (“ˆribald”, {ribald_A}) (“ˆspread”, {spread_V}) (“ˆsqueak”, {squeak_V}) (“ˆsquint”, {squint_V}) (“ˆstatue”, {statue_N}) (“ˆsummar”, {summary_A}) (“whisper”, {whisper_V}) (“ˆblink”, {blink_V}) (“ˆcarri”, {carry_V}) (“ˆcheer”, {cheer_V}) (“ˆfour-”, {four_Q}) (“ˆhitch”, {hitch_V}) (“ˆlouvr”, {louvre_N}) (“ˆmuzzl”, {muzzle_N}) (“ˆnihil”, {nihilism_N}) (“ˆtooth”, {tooth_N}) (“ˆwaist”, {waist_N}) (“guard$”, {guard_N}) (“ˆbull”, {bull_N}) (“ˆrail”, {rail_V}) (“ˆseri”, {series_N}) (“ˆtest”, {test_N}) (“ˆtwo-”, {two_Q}) (“ance$”, {ance_s}) (“board”, {board_N}) (“chain”, {chain_N}) (“eroom”, {room_N}) (“grand”, {grand_A}) (“order”, {order_V}) (“power”, {power_N})

Image captions (“tennis”, {tennis racket}) (“cat”, {cat}) (“train”, {train}) (“dog”, {dog}) (“pizza”, {pizza}) (“toilet”, {toilet}) (“man”, {person}) (“bus”, {bus}) (“clock”, {clock}) (“baseball”, {baseball glove}) (“frisbee”, {frisbee}) (“bed”, {bed}) (“horse”, {horse}) (“skateboard”, {skateboard}) (“laptop”, {laptop}) (“cake”, {cake}) (“giraffe”, {giraffe}) (“table”, {dining table}) (“bench”, {bench}) (“motorcycle”, {motorcycle}) (“bathroom”, {sink}) (“elephant”, {elephant}) (“umbrella”, {umbrella}) (“kitchen”, {oven}) (“kite”, {kite}) (“people”, {person}) (“ball”, {sports ball}) (“sheep”, {sheep}) (“zebra”, {zebra}) (“phone”, {cell phone}) (“surfboard”, {surfboard}) (“hydrant”, {fire hydrant}) (“zebras”, {zebra}) (“teddy”, {teddy bear}) (“truck”, {truck}) (“stop sign”, {stop sign}) (“sandwich”, {sandwich}) (“boat”, {boat}) (“street”, {car}) (“bat”, {baseball bat}) (“bananas”, {banana}) (“giraffes”, {giraffe}) (“living”, {couch}) (“snow”, {skis}) (“bird”, {bird}) (“elephants”, {elephant}) (“vase”, {vase}) (“cows”, {cow}) (“broccoli”, {broccoli}) (“computer”, {keyboard}) (“woman”, {person}) (“tie”, {tie}) (“horses”, {horse}) (“bear”, {bear}) (“desk”, {mouse}) (“plane”, {airplane}) (“luggage”, {suitcase}) (“airplane”, {airplane}) (“person”, {person}) (“hot”, {hot dog}) (“refrigerator”, {refrigerator}) (“wii”, {remote}) (“kites”, {kite}) (“boats”, {boat}) (“couch”, {couch}) (“traffic”, {traffic light}) (“plate”, {fork}) (“surf”, {surfboard}) (“umbrellas”, {umbrella}) (“wine”, {wine glass}) (“skate”, {skateboard}) (“bowl”, {bowl}) (“stuffed”, {teddy bear}) (“room”, {tv}) (“cow”, {cow}) (“scissors”, {scissors}) (“snowboard”, {snowboard}) (“chair”, {chair}) (“car”, {car}) (“banana”, {banana}) (“bicycle”, {bicycle}) (“birds”, {bird}) (“vegetables”, {broccoli}) (“microwave”, {microwave}) (“donuts”, {donut}) (“video”, {remote}) (“batter”, {baseball bat, person}) (“skateboarder”, {person, skateboard}) (“surfer”, {person, surfboard}) (“skis”, {skis}) (“motorcycles”, {motorcycle}) (“meter”, {parking meter}) (“suitcase”, {suitcase}) (“sink”, {sink}) (“bike”, {bicycle}) (“chairs”, {chair}) (“food”, {bowl}) (“dogs”, {dog}) (“oven”, {oven}) (“court”, {sports ball})

Machine translation (“and”, {und}) (“Commission”, {Kommission}) (“not”, {nicht}) (“Union”, {Union}) (“we”, {wir}) (“I”, {ich}) (“that”, {daß}) (“Mr”, {Herr}) (“I”, {Ich}) (“Parliament”, {Parlament}) (“President”, {Präsident}) (“Member States”, {Mitgliedstaaten}) (“report”, {Bericht}) (“European”, {Europäischen}) (“We”, {Wir}) (“or”, {oder}) (“in”, {in}) (“Europe”, {Europa}) (“the”, {der}) (“Council”, {Rat}) (“between”, {zwischen}) (“is”, {ist}) (“2000”, {2000}) (“Commissioner”, {Kommissar}) (“EU”, {EU}) (“for”, {für}) (“the”, {die}) (“The”, {Die}) (“also”, {auch}) (“with”, {mit}) (“like to”, {möchte}) (“you”, {Sie}) (“1999”, {1999}) (“directive”, {Richtlinie}) (“only”, {nur}) (“proposal”, {Vorschlag}) (“European”, {Europäische}) (“Madam”, {Präsidentin}) (“Mrs”, {Frau}) (“Kosovo”, {Kosovo}) (“but”, {aber}) (“new”, {neuen}) (“Group”, {Fraktion}) (“have”, {haben}) (“behalf”, {Namen}) (“Mr”, {Herrn}) (“women”, {Frauen}) (“has”, {hat}) (“regions”, {Regionen}) (“years”, {Jahren}) (“all”, {alle}) (“two”, {zwei}) (“cooperation”, {Zusammenarbeit}) (“if”, {wenn}) (“1”, {1}) (“new”, {neue}) (“Article”, {Artikel}) (“because”, {weil}) (“whether”, {ob}) (“Parliament”, {Parlaments}) (“a”, {eine}) (“measures”, {Maßnahmen}) (“but”, {sondern}) (“institutions”, {Institutionen}) (“social”, {sozialen}) (“to”, {zu}) (“political”, {politischen}) (“development”, {Entwicklung}) (“national”, {nationalen}) (“today”, {heute}) (“countries”, {Länder}) (“European”, {europäischen}) (“must”, {muß}) (“our”, {unsere}) (“as”, {wie}) (“problems”, {Probleme}) (“initiative”, {Initiative}) (“work”, {Arbeit}) (“be”, {werden}) (“very”, {sehr}) (“human rights”, {Menschenrechte}) (“of the”, {des}) (“us”, {uns}) (“three”, {drei}) (“debate”, {Aussprache}) (“other”, {anderen}) (“hope”, {hoffe}) (“already”, {bereits}) (“question”, {Frage}) (“this”, {diesem}) (“debate”, {Debatte}) (“are”, {sind}) (“will”, {wird}) (“proposals”, {Vorschläge}) (“If”, {Wenn}) (“Prodi”, {Prodi}) (“Council”, {Rates}) (“rapporteur”, {Berichterstatter}) (“INTERREG”, {INTERREG}) (“role”, {Rolle})

E.5.2 Emergent languages

Below are the top 20 morphemes for the various emergent languages described in Section 6.5.2. For attr–val environments, a semantic component of “1_2” means the 1st attribute has a value of 2.

Vector, Attr–val, Discrim, Run 0 (“17”, {3_3}), (“16”, {1_3}), (“22”, {0_1}), (“24”, {3_1}), (“40”, {2_1}), (“22 39”, {0_0}), (“22”, {1_0}), (“46 44 46 46 44 46 46”, {0_1, 1_0, 3_2}), (“46”, {0_1}), (“31 31 31 31”, {3_2}), (“46”, {1_0}), (“22 5 11 5”, {0_0, 2_2}), (“24 24”, {2_1}), (“24 44”, {3_0}), (“11”, {2_2}), (“11”, {2_3}), (“5”, {3_1}), (“39 39”, {0_0, 1_0}), (“24 22 24”, {2_1}), (“44 31”, {0_2, 1_1})

Vector, Attr–val, Discrim, Run 1 (“7”, {0_2}), (“58 58”, {0_1}), (“4 4 4”, {0_0}), (“52”, {2_1}), (“4 0”, {3_3}), (“4 58 4”, {0_3}), (“19 19 40”, {1_3, 2_2}), (“19 19 40 40”, {1_2, 2_2}), (“21 21 21 21 21 21 21 0”, {3_0}), (“60”, {2_3}), (“19 40 40 40”, {1_2, 2_0}), (“40 40 40 40 40”, {1_2, 2_3}), (“21 21”, {3_1}), (“19 40 40”, {1_3, 2_0}), (“19 19 19”, {1_2}), (“21 21 21”, {3_0}), (“19 40 4”, {1_1, 2_0}), (“21”, {3_2}), (“48 40”, {1_1, 2_3}), (“58 21 58”, {0_1})

Vector, Attr–val, Discrim, Run 2 (“10”, {1_1}), (“58 58 58 58 58 0”, {2_1}), (“24 24 24 24 24 0”, {2_3}), (“63 26”, {1_3}), (“53”, {1_0}), (“35”, {3_0}), (“46”, {1_2}), (“45 45”, {0_3}), (“56”, {2_0}), (“58”, {2_2}), (“24 0”, {2_0}), (“26 26 26”, {1_3}), (“45 3 3”, {1_2, 3_3}), (“45 63 63”, {3_2}), (“45 26 3”, {1_0, 3_3}), (“45”, {0_2, 3_1}), (“3 3 24 24”, {3_3}), (“24 24 24”, {3_3}), (“63 63 63 63”, {3_0}), (“53 63 53 63”, {0_0, 3_0})

Vector, Attr–val, Recon, Run 0 (“12 12 12”, {2_2}), (“29 29 29”, {1_2}), (“40 40 40”, {2_1}), (“11”, {1_1}), (“13”, {0_2}), (“14”, {0_1}), (“16”, {3_1}), (“17”, {2_0}), (“31”, {3_0}), (“37”, {1_0}), (“4”, {0_3}), (“41”, {3_2}), (“44”, {2_3}), (“59”, {3_3}), (“8”, {0_0}), (“50”, {1_3}), (“41 8 41 8 41 8”, {1_3}), (“17 16”, {1_3}), (“4 41 4 41 4 41 4 41”, {1_3}), (“41 13 41 13 17 41 17”, {1_3})

Vector, Attr–val, Recon, Run 1 (“21 21”, {0_2}), (“10”, {2_1}), (“13”, {1_1}), (“14”, {3_1}), (“37”, {0_0}), (“42”, {2_2}), (“46”, {3_0}), (“47”, {0_1}), (“48”, {3_3}), (“6”, {1_3}), (“61”, {0_3}), (“16”, {2_3}), (“38”, {1_2}), (“17”, {2_0}), (“17”, {3_2}), (“17”, {1_0}), (“28 28 28 28 0”, {1_2}), (“28”, {2_3})

Vector, Attr–val, Recon, Run 2 (“27 27 0”, {0_0}), (“63 63 0”, {0_2}), (“4 4”, {3_0}), (“61 61”, {3_2}), (“11”, {3_1}), (“14”, {1_1}), (“47”, {1_0}), (“6”, {1_2}), (“56”, {1_3}), (“1”, {2_1}), (“26”, {2_0}), (“21”, {2_3}), (“18 0”, {0_3}), (“22 22 22 22 0”, {0_3, 2_2}), (“31 31 31 31 0”, {0_1, 2_3}), (“10 10 10 0”, {0_1, 2_2}), (“29”, {2_2}), (“40 40 40 40”, {0_1, 2_0}), (“55 55”, {0_1, 2_1}), (“55”, {2_1})

Vector, Sparse, Discrim, Run 0 (“5 5”, {1, 5}), (“11”, {1, 2}), (“52”, {7}), (“2”, {2, 5}), (“5”, {1}), (“6”, {2}), (“16”, {2, 4}), (“20”, {5}), (“34”, {3, 4}), (“34 34 34”, {4}), (“34 34”, {4}), (“52 38”, {0, 1, 4}), (“52 52 52”, {5}), (“34”, {4}), (“5”, {5}), (“2”, {2}), (“26 45 26”, {5, 7}), (“11”, {1}), (“11”, {2}), (“38 38”, {0})

Vector, Sparse, Discrim, Run 1 (“50”, {2}), (“16”, {4}), (“24”, {2, 7}), (“25”, {1}), (“42”, {0}), (“54”, {7}), (“24”, {2}), (“45 45”, {7}), (“36 36”, {0}), (“24”, {7}), (“36”, {5}), (“42”, {2}), (“34 34 42”, {1, 4, 5}), (“18”, {1, 5}), (“25 25 25 25 25”, {4}), (“42 42 42”, {5}), (“10 34”, {1, 4}), (“36”, {0}), (“52 52”, {1, 7}), (“52 9 52 9 52”, {0, 4, 5})

Vector, Sparse, Discrim, Run 2 (“9 0”, {2, 5}), (“34”, {5}), (“32 32 32”, {1, 7}), (“7”, {4}), (“33 33”, {7}), (“1”, {1}), (“5 5”, {2}), (“9”, {5}), (“55 0”, {1, 2, 6}), (“32”, {3, 6}), (“55 55”, {0, 5}), (“36”, {1}), (“49 49 48”, {0, 3, 4, 6}), (“9 9 9 9”, {2}), (“2 2 2”, {0, 4}), (“48 48 48 48 48”, {2, 4}), (“33 32”, {7}), (“33”, {3}), (“14 5”, {6}), (“55 55”, {5, 6})

Vector, Sparse, Recon, Run 0 (“2”, {6}), (“53”, {5}), (“27”, {0}), (“28”, {1, 2}), (“28”, {1}), (“26”, {1, 5}), (“26”, {5}), (“28”, {2, 3}), (“22 8”, {2, 5}), (“49 53”, {2, 3}), (“12 12 58”, {0}), (“58 5 58”, {2}), (“50”, {4}), (“5 49 5”, {2, 4}), (“5 5”, {2, 4}), (“36”, {7}), (“8 53 8”, {1, 2}), (“38 5 38”, {4}), (“49 58”, {3}), (“8 26”, {1})

Vector, Sparse, Recon, Run 1 (“8”, {4, 5}), (“50”, {1, 2}), (“21”, {3, 7}), (“42”, {1}), (“50”, {1}), (“31 31”, {5}), (“19”, {4}), (“33”, {2}), (“1”, {0}), (“23”, {3, 6}), (“62”, {7}), (“50”, {2}), (“37”, {5, 6}), (“23”, {7}), (“37”, {4, 6}), (“47”, {7}), (“23”, {6}), (“8”, {4}), (“21”, {3, 5}), (“47”, {3})

Vector, Sparse, Recon, Run 2 (“19 19”, {2, 6}), (“4”, {4, 7}), (“58”, {3}), (“12”, {4}), (“14”, {7}), (“10 10”, {6}), (“34”, {2}), (“56 56”, {0, 1, 3}), (“56 23”, {0, 1, 5}), (“4”, {4}), (“23”, {0, 5}), (“4”, {7}), (“56”, {1, 3}), (“59 59”, {3, 5}), (“30”, {5}), (“3”, {3}), (“23”, {1, 5}), (“51”, {1}), (“56 58”, {0, 7}), (“8 8”, {0, 3})

Image, Synth, Ref, Run 0 (“6”, {gray}), (“3”, {square}), (“4”, {rectangle}), (“6 15 6”, {circle, yellow}), (“4 8”, {square}), (“3 8 8 4”, {circle, red}), (“5 6 15 5 6”, {blue, triangle}), (“14 10 14 10”, {triangle}), (“16 8 6”, {green, square}), (“4 10 10 4”, {circle, red}), (“8 6 10”, {ellipse}), (“6 12 5”, {yellow}), (“8 4 13”, {square, white}), (“4 4 4”, {green}), (“12 12 12 6 12”, {circle, green}), (“4 6”, {square}), (“5 14 5”, {red}), (“5 5 15 14”, {ellipse, red}), (“4 8 8 4”, {triangle, yellow}), (“15”, {rectangle})

Image, Synth, Ref, Run 1 (“8 11”, {square}), (“16 4 16”, {square}), (“16 4”, {square}), (“4 16”, {square}), (“5 16”, {square}), (“16 4”, {triangle}), (“13 0 11”, {gray, rectangle}), (“16 0 16”, {white}), (“0 16”, {ellipse}), (“0 8 8”, {green}), (“4 7 4 4 7”, {ellipse, gray}), (“11 11 4 16 11”, {ellipse, gray}), (“16 11 4 4”, {green, triangle}), (“8 16 16”, {ellipse}), (“4 8 8 8”, {green, rectangle}), (“5 8 7 5 7”, {red, square}), (“14”, {square}), (“0 0 11 16”, {ellipse}), (“7 7 4 4 7”, {red, triangle}), (“13 14 16 16”, {triangle, yellow})

Image, Synth, Ref, Run 2 (“7”, {rectangle}), (“4”, {gray}), (“15 15 9”, {square}), (“9”, {square}), (“9”, {ellipse}), (“9”, {gray}), (“9”, {rectangle}), (“7 15 15”, {red, square}), (“13 8 8”, {red, square}), (“4 7 7 15”, {blue, circle}), (“14 14 13 7”, {circle, green}), (“7 7 7 15 7”, {square, yellow}), (“15 15 15 9”, {blue, circle}), (“0 0 0 8 0”, {white}), (“0 0 0 4 8”, {red, square}), (“13 15 13”, {green}), (“15 4 4 4 15”, {blue, triangle}), (“0 0 13 0 0”, {rectangle, yellow}), (“13 15 0”, {red}), (“0 0 7 4”, {green, triangle})

Image, Synth, Set-ref, Run 0 (“13”, {circle, not}), (“11 11”, {blue, not}), (“12 12”, {ellipse}), (“3 3 3”, {gray, not}), (“10”, {not, or}), (“12”, {not, rectangle}), (“8 11 7”, {green, or, yellow}), (“7”, {not, red}), (“11 8 8”, {or, red, yellow}), (“9”, {triangle}), (“14 8”, {gray, or, yellow}), (“8”, {and, not, white}), (“16 7 7”, {blue, or, white}), (“14 14 7”, {or, white, yellow}), (“11 3 3”, {green, or, red}), (“16 14 14”, {gray, or, white}), (“7 8”, {blue, or, yellow}), (“14 14 11”, {and, blue, green}), (“10”, {not, triangle}), (“3 3 8”, {blue, or, red})

Image, Synth, Set-ref, Run 1 (“9”, {or, yellow}), (“10”, {not, yellow}), (“3 3”, {gray, not}), (“14 14”, {blue, not}), (“7”, {and, not, white}), (“5”, {not, red}), (“12”, {not, rectangle}), (“11”, {or, white}), (“12 12”, {ellipse}), (“10”, {red}), (“4 4”, {triangle}), (“5 5 5”, {blue}), (“11 7”, {gray}), (“12”, {ellipse, or}), (“4”, {or, triangle}), (“7 7”, {green, not, white}), (“5”, {blue, or}), (“4”, {circle, not}), (“9 9”, {yellow}), (“10 11”, {and, green, yellow})

Image, Synth, Set-ref, Run 2 (“12”, {not, red}), (“8 8”, {blue}), (“10 10”, {not, yellow}), (“0 0”, {blue, not}), (“15”, {or, red}), (“5 5”, {triangle}), (“3”, {not, rectangle}), (“13”, {circle, not}), (“15”, {gray, not}), (“15”, {and, not, white}), (“3 10”, {circle, ellipse, or}), (“3 3 3”, {ellipse}), (“9 9”, {white}), (“13”, {or, yellow}), (“3”, {ellipse, or}), (“12 12”, {red}), (“9 10”, {ellipse, not, triangle}), (“5”, {square}), (“10 12 9”, {gray, or, white}), (“0 10”, {and, blue, yellow})

Image, Synth, Concept, Run 0 (“13 14”, {not, yellow}), (“15 13”, {not, red}), (“16”, {yellow}), (“15 15”, {gray, not}), (“14 3”, {red}), (“16”, {rectangle}), (“3”, {blue, not}), (“16”, {circle}), (“4 4”, {white}), (“5 5 5”, {and, not, white}), (“9 9 9”, {green}), (“5 15 15”, {blue, or, yellow}), (“16”, {triangle}), (“16”, {ellipse}), (“5 3 3”, {or, red, yellow}), (“15 3”, {yellow}), (“13 13”, {not, yellow}), (“3 13 13”, {gray, or, white}), (“3 5”, {gray, or, yellow}), (“3 15 15”, {or, white, yellow})

Image, Synth, Concept, Run 1 (“11 11”, {blue}), (“4”, {not, yellow}), (“16 16”, {not, red}), (“16 11”, {gray, not}), (“8”, {or, yellow}), (“14 14”, {blue, not}), (“8”, {not, white}), (“14 4”, {white}), (“3 3”, {green, or, red}), (“11 8”, {and, green, white}), (“8 8 8”, {yellow}), (“11 3”, {and, white, yellow}), (“3 3”, {and, blue, white}), (“13”, {red}), (“5 5”, {not, or}), (“14 14 16”, {white, yellow}), (“10 10 10 10”, {and, green}), (“4”, {or, white}), (“3”, {blue, not}), (“4”, {and, yellow})

Image, Synth, Concept, Run 2 (“8 8”, {not, yellow}), (“6”, {not, red}), (“0 0”, {blue, not}), (“0 11”, {gray}), (“6 6”, {gray, not}), (“15”, {yellow}), (“15”, {rectangle}), (“3 0”, {or, white, yellow}), (“0”, {white}), (“11 6 6”, {blue, or, yellow}), (“11 11 8”, {blue, or, white}), (“15”, {circle}), (“11”, {and, green, not}), (“15”, {triangle}), (“15”, {ellipse}), (“8 6”, {red}), (“8 3 3”, {green, not, or}), (“11 11”, {blue}), (“11 0”, {gray, or, yellow}), (“15”, {square})

Image, Natural, Ref, Run 0 (“22”, {221}), (“22”, {226}), (“22”, {46}), (“7”, {204, 46}), (“22”, {255}), (“22”, {236}), (“12”, {146, 152, 245}), (“18”, {146, 236}), (“6 22”, {102, 165, 195}), (“22”, {102, 228}), (“22 22”, {152}), (“6”, {46}), (“22”, {6}), (“6”, {204}), (“22”, {150, 306}), (“7”, {236}), (“12”, {127, 55}), (“3”, {152, 245, 8}), (“22”, {146, 152, 55}), (“20”, {146, 221})

Image, Natural, Ref, Run 1 (“6”, {221}), (“6”, {226}), (“11”, {46}), (“11”, {236}), (“11”, {22, 306}), (“6”, {236}), (“11”, {6}), (“11”, {226}), (“10”, {236}), (“6”, {146, 152}), (“6”, {146}), (“22”, {146, 255}), (“11”, {222}), (“6”, {152}), (“11”, {150, 30, 306}), (“7”, {5}), (“20”, {211, 262, 76}), (“18”, {204}), (“6”, {219}), (“11”, {146, 7})

Image, Natural, Ref, Run 2 (“17”, {221}), (“7”, {221}), (“17”, {236}), (“0”, {5}), (“18”, {221}), (“17”, {204}), (“7”, {226}), (“0”, {165}), (“17”, {146, 152}), (“0”, {146, 222}), (“18 18”, {102, 92}), (“0”, {195, 306}), (“16 3”, {102, 210, 92}), (“0”, {146, 245, 8}), (“10”, {6}), (“18”, {255}), (“18”, {261, 52, 92}), (“7”, {148}), (“18”, {117, 164, 36}), (“16”, {222})

Image, Natural, Set-ref, Run 0 (“7 7 7 7 7 7 7 7 7”, {204, 236, 255}), (“6 6”, {241, 260, 91}), (“5”, {221}), (“10”, {5}), (“5”, {226}), (“10”, {146, 195, 245}), (“15 15 15”, {249, 294}), (“12”, {219}), (“6”, {245, 260, 91}), (“10”, {30, 306}), (“3 3”, {310}), (“15”, {210, 7}), (“10”, {221, 245}), (“21”, {150, 195, 55}), (“3 3”, {11, 60}), (“14”, {166}), (“7”, {150, 37}), (“7”, {204, 255}), (“7 7”, {236, 241}), (“7”, {236, 255, 46})

Image, Natural, Set-ref, Run 1 (“6”, {236}), (“17 3”, {255}), (“6”, {221}), (“4 4”, {307}), (“18 18”, {165, 55}), (“13”, {241, 245, 260}), (“17”, {46}), (“6 6”, {226}), (“12 18”, {245, 306}), (“17”, {146, 236}), (“4 4”, {166, 219}), (“19”, {7}), (“5”, {310}), (“15”, {55}), (“19 19”, {153, 80}), (“18”, {146, 210}), (“6”, {152, 219}), (“17”, {204}), (“6”, {219}), (“19”, {236, 245, 52})

Image, Natural, Set-ref, Run 2 (“18”, {236}), (“17”, {255}), (“18 21”, {226}), (“18”, {221}), (“14 14”, {245, 260, 275}), (“18”, {219}), (“0 18”, {165, 55}), (“14 14”, {275, 55, 70}), (“18”, {7}), (“18”, {146, 152}), (“0 0”, {245, 306}), (“10 10”, {307}), (“17”, {146, 236}), (“16 16”, {10, 153}), (“0”, {260}), (“16”, {7}), (“18”, {152}), (“0”, {36, 70}), (“0”, {146, 245, 30}), (“22”, {112, 236})

Image, Natural, Concept, Run 0 (“14 17”, {255}), (“15 15”, {245, 306}), (“9”, {219}), (“9”, {221}), (“15”, {146, 195, 245}), (“20”, {305, 309, 36}), (“15 15”, {165, 55}), (“17”, {46}), (“20”, {164, 179, 241}), (“10”, {166}), (“15”, {150, 306}), (“8 8”, {117, 152, 55}), (“10”, {307}), (“15”, {146, 195, 55}), (“8”, {132, 209, 219}), (“8”, {132, 219, 51}), (“8”, {102, 132, 275}), (“16”, {118, 236, 241}), (“20”, {194, 309}), (“17”, {204})

Image, Natural, Concept, Run 1 (“19 19 19 19”, {226}), (“12”, {255}), (“11 11 11 11 11”, {245, 306}), (“11 11”, {146, 195, 55}), (“6 6”, {260, 290, 55}), (“12”, {46}), (“11”, {5}), (“6 6 6”, {290, 55, 70}), (“6 6”, {194, 209, 219}), (“15”, {204}), (“12”, {112}), (“19”, {221}), (“0”, {146, 236, 7}), (“6”, {219, 245, 51}), (“16”, {249, 294}), (“15”, {146, 236}), (“16”, {10, 152}), (“19”, {219}), (“20”, {118, 236, 241}), (“4”, {210, 236})

Image, Natural, Concept, Run 2 (“11 11 11 11 11 11 11 11”, {221}), (“8 5”, {112, 255}), (“8”, {255, 46}), (“5”, {112}), (“8”, {255}), (“5”, {204}), (“8”, {46}), (“16 16 16”, {152, 249}), (“15”, {53}), (“21”, {210, 236, 52}), (“16 16”, {249, 294}), (“11 11 11”, {179, 51}), (“17”, {219, 241, 52}), (“16”, {10, 152}), (“15”, {119}), (“9”, {36}), (“11 11”, {21, 51}), (“9”, {179}), (“9”, {21}), (“9”, {222})

Appendix F

Communicating with Induced Morphological Phrasebooks

F.1 Environment Details

The following hyperparameters (for the main experiment) were selected to optimize for online and phrasebook agent performance base, in part, on the environment and agent ablation experiments. The neural network architecture of the online agents is as follows:

- Observation (concatenated one-hot vectors)
- Sender
  - Feed-forward layer (embedding layer)
  - Tanh activation layer
  - Feed forward layer
  - RNN (w/ Gumbel-Softmax layer)
- Message (max length +1 by vocabulary size matrix; last token is an end-of-sentence token.)
- Receiver
  - RNN
  - Feed-forward layer (embedding layer)
  - Tanh activation layer
  - Feed forward layer
  - Projection layer to observation space
- Prediction (same dimension as observation)

The offline agents have the same architecture. The offline sender is trained with cross-entropy loss after the bottleneck layer. The offline receiver is trained with mean squared error after the projection layer (like the online receiver).

Our environment uses the following hyperparameters.

accuracy threshold 0.9999; training accuracy value at which next agent reset is performed or training is stopped if all resets have been performed.

neural network resets (receiver, receiver); sequence of resets of sender/receivers after accuracy threshold is reached.

max epochs 300; epochs after which training is stopped and the trial is considered a failure.

observation distribution uniform; distribution from which the values for the attributes are drawn from.

***n* attributes** 4; number of attributes in the observation.

***n* values** 4; number of distinct values in that each attribute can take.

test proportion 10%; proportion of unique observations that are held out from training for the test set to gauge generalization performance.

batch size 2^{10}; batch size of the neural network agents during training.

loss function mean squared error; objective function used for backpropagation during training; the other loss function implemented and tested is cross-entropy loss.

learning rate 1.8×10^{-3}; learning rate for neural network agents.

learning rate schedule none; learning rate was left constant during training.

***n* examples per epoch** 2^{14}; number of examples shown to the neural network agents before checking the accuracy and possibly performing rests or concluding training.

optimizer AdamW; optimizer for the neural network agents.

weight decay 1×10^{-10}; weight decay applied through the optimizer.

reset optimizer yes; whether or not to reset the optimizer's parameters for the agent that is reset as determined by the reset schedule.

max message length 8; maximum length of the message from the sender agent; the final message passed to the receiver is always one token longer than the maximum length as the message from the sender is padded with one or more end-of-sentence/pad tokens (the same token).

discrete optimization method Gumbel-Softmax; the method used for handling the discrete nature of the senders methods; REINFORCE is another method but was not used in the experiments.

sender RNN cell type GRU; other options include Elman ("RNN") and LSTM.

receiver RNN cell type *as above*.

sender RNN layers 1.

receiver RNN layers 1.

sender hidden size 256; number of hidden units in the sender's RNN.

receiver hidden size *as above, mutatis mutandis*.

sender embedding size 128; number of units in the embedding (feed-forward) layers in the sender.

receiver embedding size *as above, mutatis mutandis*.

sender *n* embedding layers 2; number of embedding layers before the RNN.

receiver *n* embedding layers 2; number of embedding layers after the RNN.

Gumbel-Softmax straight through no; whether or not to use sampled one-hot values for the vectors during the forward pass while using the categorical values during the backward pass.

Gumbel-Softmax temperature 1.

vocabulary size 64; number of possible distinct types in the message, that is, the size of the one-hot vectors forming the message; this number includes the reserved end-of-sentence/pad token.

F.2 Alternative Sender Algorithms

Both implementations of more sophisticated algorithms for the phrasebook-based sender resulted inferior performance. We speculate that the primary reason for this is that the greedy approach of selecting the highest weighted morphemes (which match the desired meaning) and repeating them as much as would fit matches the strategy of conveying meaning more than the inductive biases of the more sophisticated approaches.

Search This algorithm implemented best-first search where the objective was to account for each component of the observation by selecting a morpheme while minimizing the sum of morpheme costs, defined as the negative weight of the morpheme assigned by CSAR.

Integer Programming In this sender algorithm the meaning of each morpheme is treated as binary vector scaled by the weight given by CSAR for that morpheme. An integer programming solver would then try to maximize the distance per attribute between the correct value and the highest incorrect value by selecting a non-negative number of occurrences for each morpheme while staying within the max length.

F.3 Computing Resources

The environment employed for this paper requires a GPU with modest memory requirements (<1 GiB). Altogether, the experiments used ∼200 GPU-hours on an Nvidia L40S or equivalent.

F.4 Qualitative Results

Example inventories given in Tables F.1 to F.3. Example of messages generated by the neural and phrasebook agents (derived from Table F.1) given below:

Example 1 Neural sender: "59 59 22 22 2 2 61 61 0" interpreted as: 3133. Phrasebook sender: "59 59 59 59 22 2 61 61 0" interpreted as: 3133. Used the following morphemes: ("59 59", .1..) ("2", ...3) ("22", ..3.) ("61 61", 3...) ("59 59", .1..)

Example 2 Neural sender: "25 25 46 62 2 46 2 62 0" interpreted as: 0213. Phrasebook sender: "25 25 46 46 62 62 2 2 0" interpreted as: 0213. Used the following morphemes: ("25", .2..) ("46", ..1.) ("62", 0...) ("2", ...3) ("25", .2..) ("46", ..1.) ("62", 0...) ("2", ...3)

Example 3 Neural sender: "59 59 62 31 38 62 38 31 0" interpreted as: 0122. Phrasebook sender: "59 59 59 59 31 31 62 38 0" interpreted as: 0122. Used the following morphemes: ("31", ..2.) ("59 59", .1..) ("62", 0...) ("38", ...2) ("31", ..2.) ("59 59", .1..)

Example 4 Neural sender: "30 30 4 2 4 2 35 35 0" interpreted as: 1003. Phrasebook sender: "30 30 4 4 2 2 35 35 0" interpreted as: 1003. Used the following morphemes: ("30", .0..) ("4", ..0.) ("2", ...3) ("35", 1...) ("30", .0..) ("4", ..0.) ("2", ...3) ("35", 1...)

Form	Meaning	Weight	Proportion
22	2...	8.20 e−1	2.55 e−1
29	.1..	8.20 e−1	2.55 e−1
47	...1	8.20 e−1	2.55 e−1
28	.3..	8.13 e−1	2.51 e−1
40	..0.	8.13 e−1	2.51 e−1
60	0...	8.13 e−1	2.51 e−1
48	...2	7.99 e−1	2.42 e−1
7	.0..	7.99 e−1	2.42 e−1
8	..2.	7.99 e−1	2.42 e−1
9	..1.	7.99 e−1	2.42 e−1
61	3...	7.77 e−1	2.29 e−1
3	1...	7.65 e−1	2.55 e−1
54	...0	7.45 e−1	2.42 e−1
27	..3.	5.08 e−1	1.99 e−1
15	...3	5.02 e−1	1.90 e−1
43	.2..	5.02 e−1	1.90 e−1
2	.2.3	3.30 e−1	6.06 e−2
34	..3.	1.34 e−1	6.49 e−2
2	.2..	1.30 e−1	6.06 e−2
2	...3	1.30 e−1	6.06 e−2
33	1..0	3.92 e−2	8.66 e−3
33	...0	3.92 e−2	8.66 e−3
33	1...	3.92 e−2	8.66 e−3

Table F.1: A typical morpheme inventory.

Example 5 Neural sender: "22 30 22 30 57 19 8 8 0" interpreted as: 2031. Phrasebook sender: "30 30 22 22 57 57 8 8 0" interpreted as: 2031. Used the following morphemes: ("30", .0..) ("8", 2...) ("22", ..3.) ("57", ...1) ("30", .0..) ("8", 2...) ("22", ..3.) ("57", ...1)

Example 6 Neural sender: "22 30 22 30 38 8 38 8 0" interpreted as: 2032. Phrasebook sender: "30 30 22 22 38 38 8 8 0" interpreted as: 2032. Used the following morphemes: ("30", .0..) ("8", 2...) ("22", ..3.) ("38", ...2) ("30", .0..) ("8", 2...) ("22", ..3.) ("38", ...2)

F.5 Quantitative Summary of Results

In Figs. F.1 and F.2, we illustrate the distributions of various quantitative metrics from the main experiment (Section 7.4.1) using various metrics defined below. With the exception of *Inventory size*, the following are weighted by the *normalized prevalence*[1] of the morpheme (how often it occurred in the corpus during induction).

Inventory size Number of morphemes in the inventory.

[1]Raw prevalences do not add up to 1 since multiple morphemes can occur in one message, thus we normalize them to sum to 1 to treat them as a probability measure.

Form	Meaning	Weight	Proportion
34	...2	8.26 e−1	2.60 e−1
7	..0.	8.26 e−1	2.60 e−1
25	..2.	8.20 e−1	2.55 e−1
36	3...	8.13 e−1	2.51 e−1
43	...0	8.06 e−1	2.47 e−1
1	...1	7.99 e−1	2.42 e−1
47	..1.	7.92 e−1	2.38 e−1
16	.1..	7.85 e−1	2.34 e−1
40	1...	5.60 e−1	2.08 e−1
58	..3.	5.46 e−1	1.99 e−1
48	.2..	4.99 e−1	1.99 e−1
53	0...	4.63 e−1	1.77 e−1
17 17	02..	3.63 e−1	6.93 e−2
24 24	1.3.	2.55 e−1	4.76 e−2
54	.0..	2.49 e−1	2.55 e−1
54	...3	2.44 e−1	2.51 e−1
54	2...	2.39 e−1	2.47 e−1
17	0...	1.52 e−1	6.93 e−2
17	.2..	1.42 e−1	6.93 e−2
24	..3.	1.21 e−1	4.76 e−2
24	1...	1.18 e−1	4.76 e−2
34 34	.3..	1.79 e−2	8.66 e−3
54	.3..	8.92 e−3	9.09 e−2
47 34 53	.3..	8.89 e−3	4.33 e−3
43 43	.3..	8.89 e−3	4.33 e−3
40 34	.3..	8.89 e−3	4.33 e−3
53	.3..	7.83 e−3	4.76 e−2
36 43 58	.3..	1.95 e−3	4.33 e−3
43 47 40	.3..	1.95 e−3	4.33 e−3
47	.3..	5.88 e−4	8.66 e−3
25 1	.3..	3.87 e−4	4.33 e−3
36	.3..	3.42 e−4	3.46 e−2
25 34	.3..	3.33 e−4	4.33 e−3
25	.3..	8.07 e−5	4.33 e−3
7 1	.3..	6.62 e−5	4.33 e−3
40	.3..	1.84 e−5	4.33 e−3
34	.3..	1.84 e−5	4.33 e−3
1	.3..	3.40 e−6	4.33 e−3

Table F.2: An outlier inventory with very poor performance.

Form	Meaning	Weight	Proportion
4 4	...3	8.33 e−1	2.64 e−1
31	...0	8.26 e−1	2.60 e−1
40 40	3...	8.20 e−1	2.55 e−1
28	2...	8.20 e−1	2.55 e−1
49	1...	8.13 e−1	2.51 e−1
60	...1	8.13 e−1	2.51 e−1
17	0...	7.92 e−1	2.38 e−1
41	...2	7.69 e−1	2.25 e−1
1 1 1	.00.	3.63 e−1	6.93 e−2
26 26 26	.30.	3.63 e−1	6.93 e−2
42 42 42	.10.	3.63 e−1	6.93 e−2
61 61 61	.01.	3.63 e−1	6.93 e−2
15 15 15	.33.	3.47 e−1	6.49 e−2
18 18 18	.31.	3.47 e−1	6.49 e−2
33 33 33	.22.	3.47 e−1	6.49 e−2
43 43 43	.20.	3.47 e−1	6.49 e−2
8 8 8	.11.	3.47 e−1	6.49 e−2
10 10 10	.13.	3.30 e−1	6.06 e−2
13 13 13	.21.	3.30 e−1	6.06 e−2
23 23 23	.32.	3.30 e−1	6.06 e−2
29 29 29	.12.	3.12 e−1	5.63 e−2
38 38 38	.02.	3.12 e−1	5.63 e−2
16 16 16	.03.	2.95 e−1	5.19 e−2
55 55 55	.23.	2.95 e−1	5.19 e−2
61	.0..	8.22 e−2	3.90 e−2
26	.3..	7.91 e−2	3.90 e−2
13	.2..	7.37 e−2	3.46 e−2
10	..3.	6.67 e−2	3.03 e−2
23	..2.	6.50 e−2	3.03 e−2
42	.1..	6.25 e−2	3.03 e−2
8	.1..	6.25 e−2	3.03 e−2
15	..3.	5.69 e−2	2.60 e−2
16	..3.	5.69 e−2	2.60 e−2
33	..2.	5.54 e−2	2.60 e−2
29	..2.	5.54 e−2	2.60 e−2
1	.0..	5.40 e−2	2.60 e−2
18	..1.	5.20 e−2	2.60 e−2
43	.2..	4.53 e−2	2.16 e−2
55	..3.	3.75 e−2	1.73 e−2
38	..2.	3.66 e−2	1.73 e−2

Table F.3: First 40 entries of inventory with a larger degree of fusionality (multiple atomic meanings per morpheme).

Vocabulary size Number of distinct form tokens.
Inventory entropy Entropy of morphemes by normalized prevalence.
Form length Mean length of the morphemes' forms.
Meaning size Mean size of the morphemes' meanings.
Morpheme bijectivity Defined in Section 7.4.4.
Forms per meaning Mean number of distinct forms which map to a given meaning.
Meanings per form Mean number of distinct meanings which map to a given form.
Synonymy entropy Entropy of forms conditioned on a particular meaning (across all meanings).
Polysemy entropy Entropy of meanings conditioned on a particular form (across all forms).

F.6 Environment Ablation Results

The full results for the ablation of the morpheme induction algorithm and phrasebook agents are shown in Figs. F.3 and F.4.

F.7 Phrasebook Agent Ablation Details

Weight method The default method of weighting morphemes in CSAR is mutual information. Other methods include provided and tested include:

NPMI Normalized pointwise mutual information of the form and meaning.
Joint probability The probability of the form and meaning occurring together in the corpus.
PMIM Pointwise mutual information mass; the PMI of the form and meaning weighted by the their joint probability.
Applicability Defined as weighted probability of the meaning occurring given the form less the probability of the meaning not occurring given the form and the meaning occurring given the form is *not* present; i.e., $p(m,f)p(m|f) - p(\neg m,f)p(\neg m|f) - p(m,\neg f)p(m|\neg f)$.

Results The full results for the ablation of the morpheme induction algorithm and phrasebook agents are shown in Figs. F.5 and F.6.

F.8 Compositionality Metrics

Scatter plots and R^2 values given in Fig. F.7.

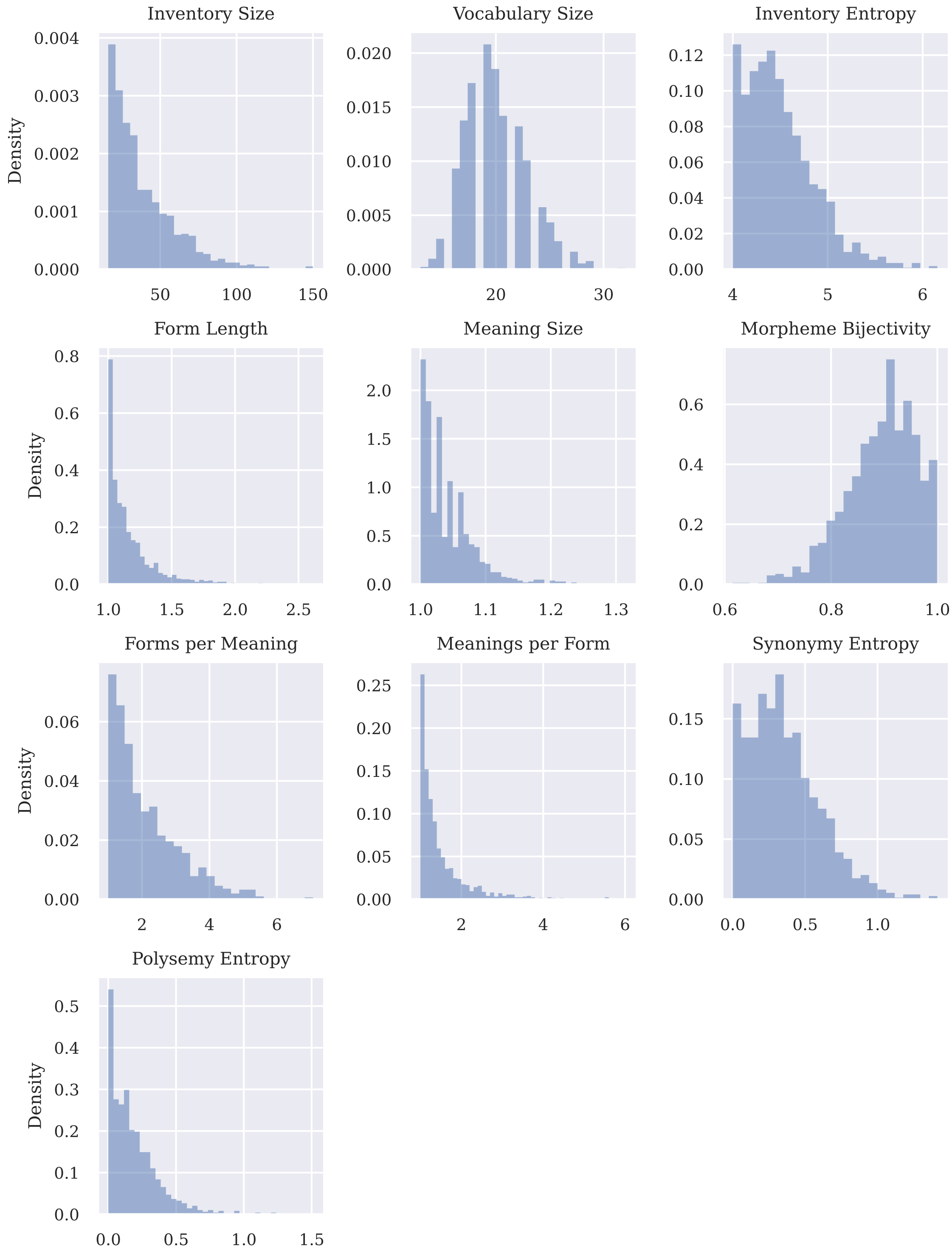


Figure F.1: Histograms of metrics from main experiment.

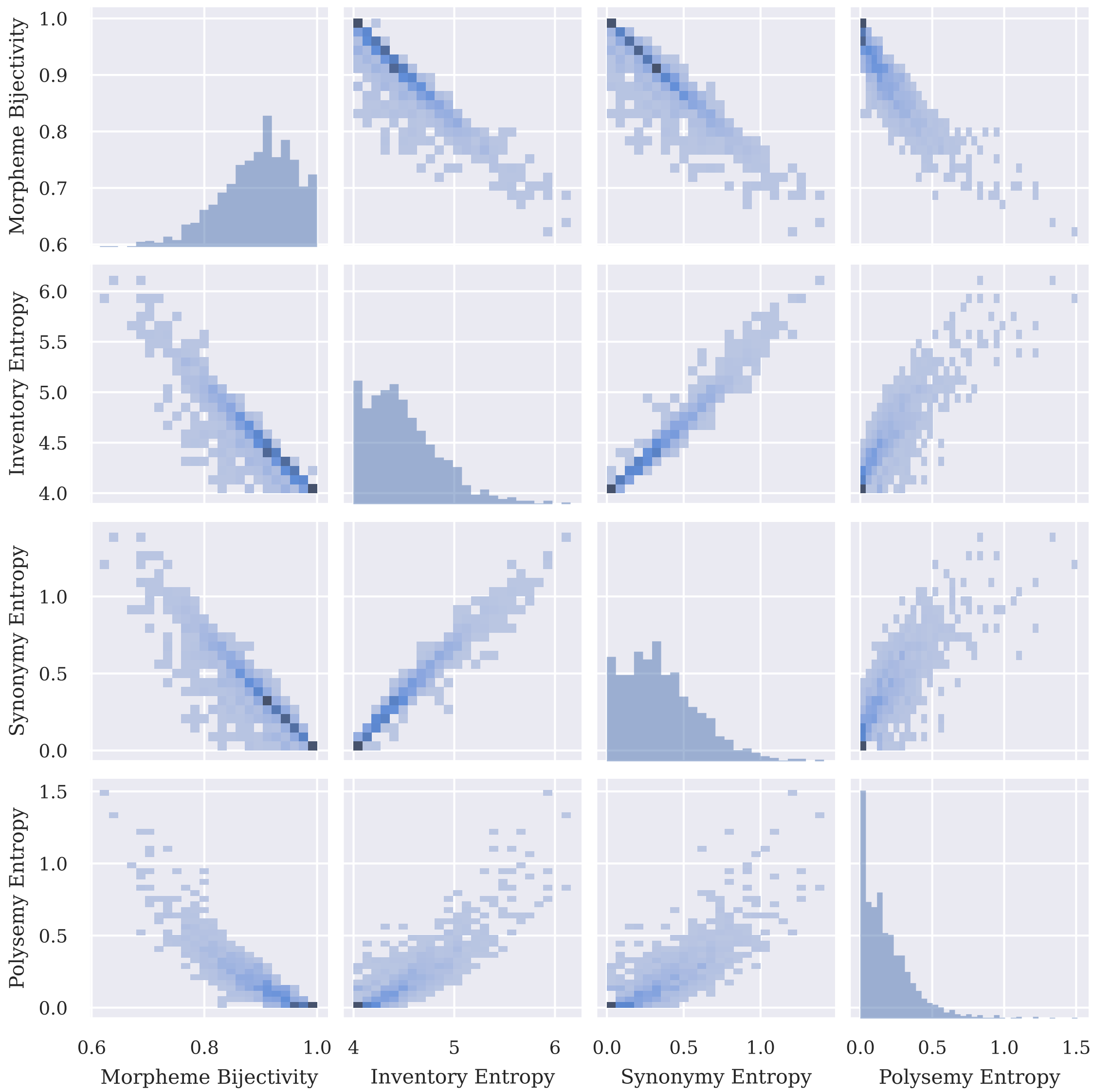


Figure F.2: Bivariate histograms of selection of metrics from main experiment.

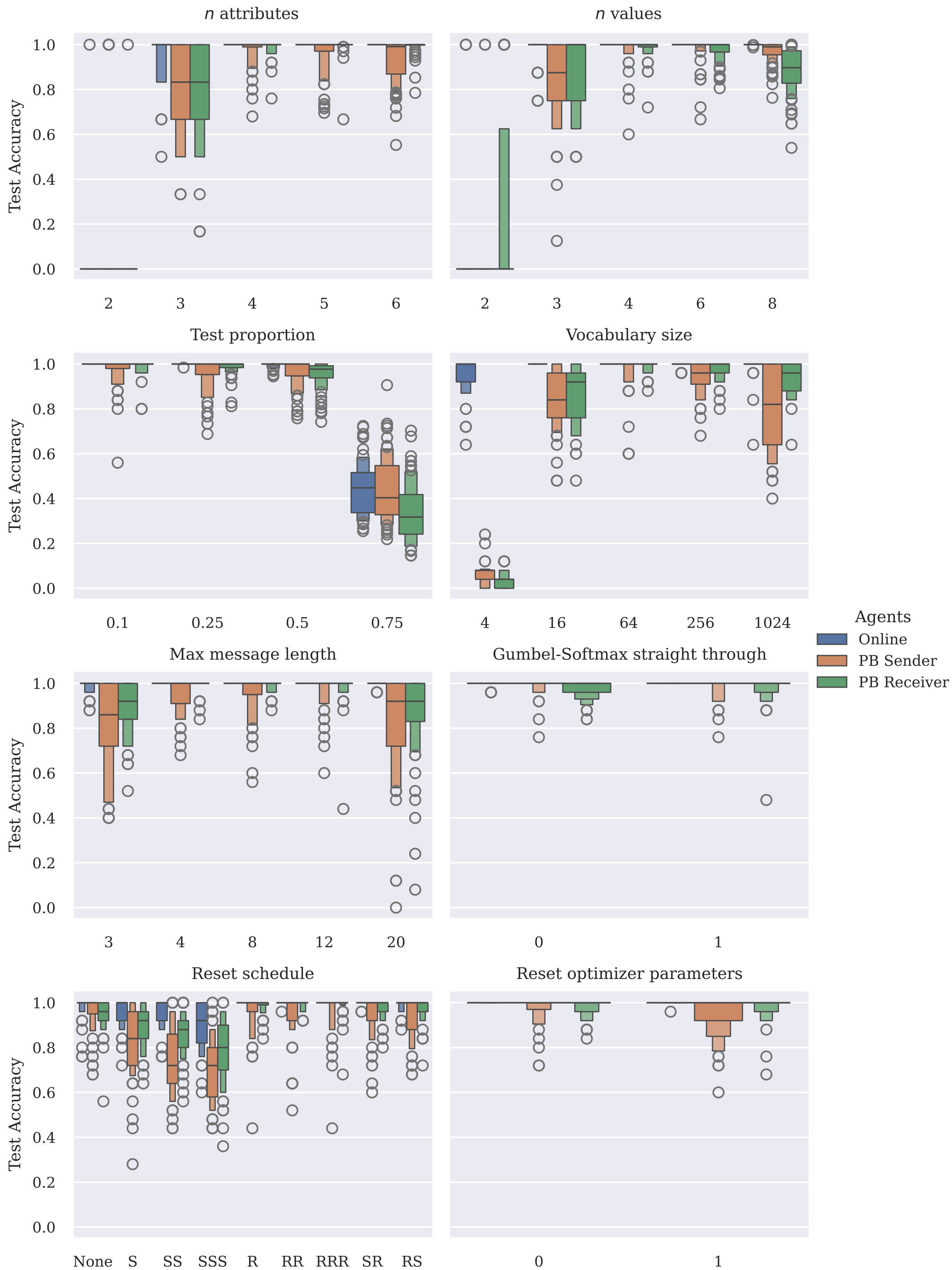


Figure F.3: Phrasebook agents' test accuracy across morpheme induction and phrasebook agent ablations. *Default values.

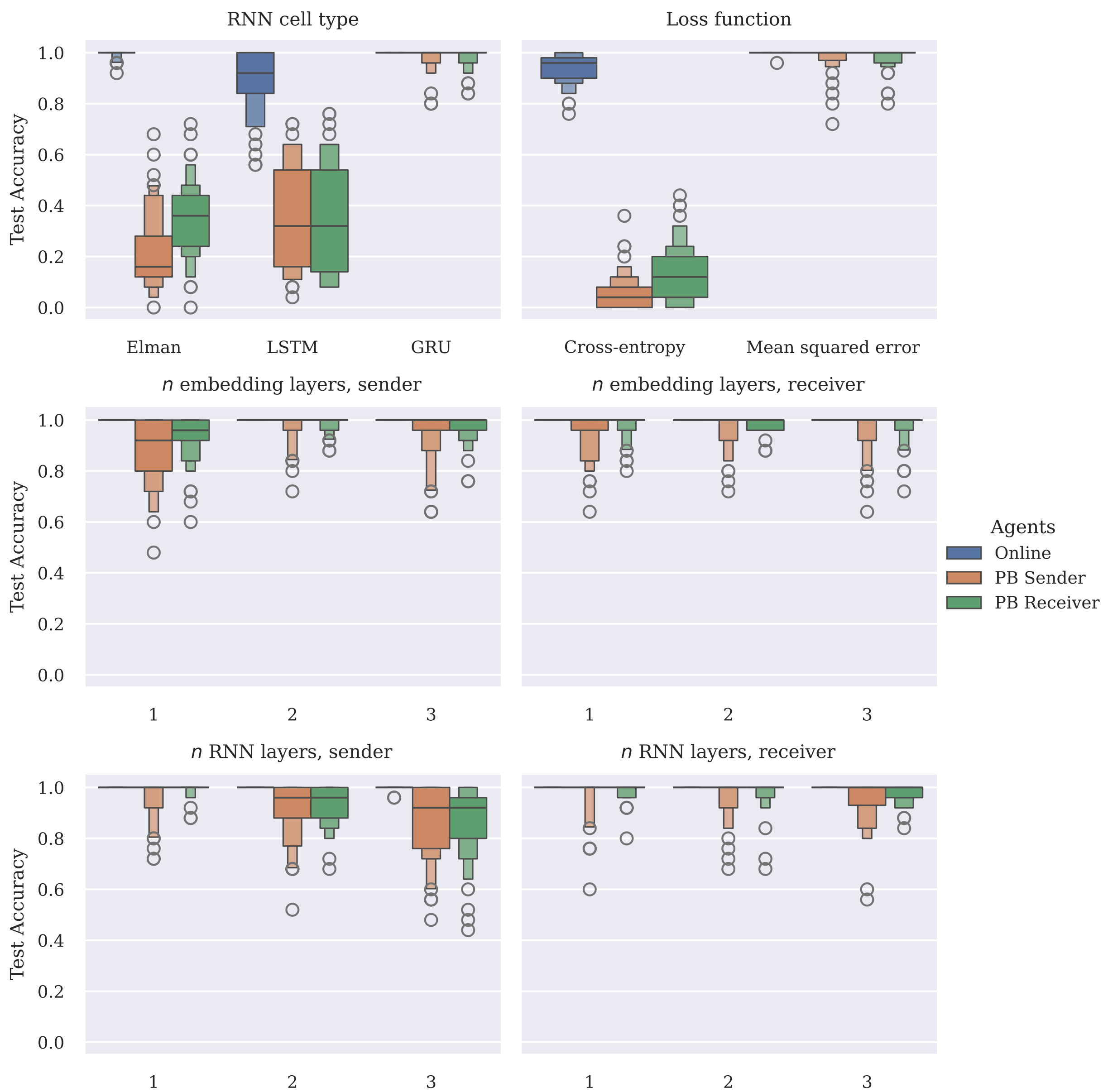


Figure F.4: Phrasebook agents' test accuracy across morpheme induction and phrasebook agent ablations. *Default values.

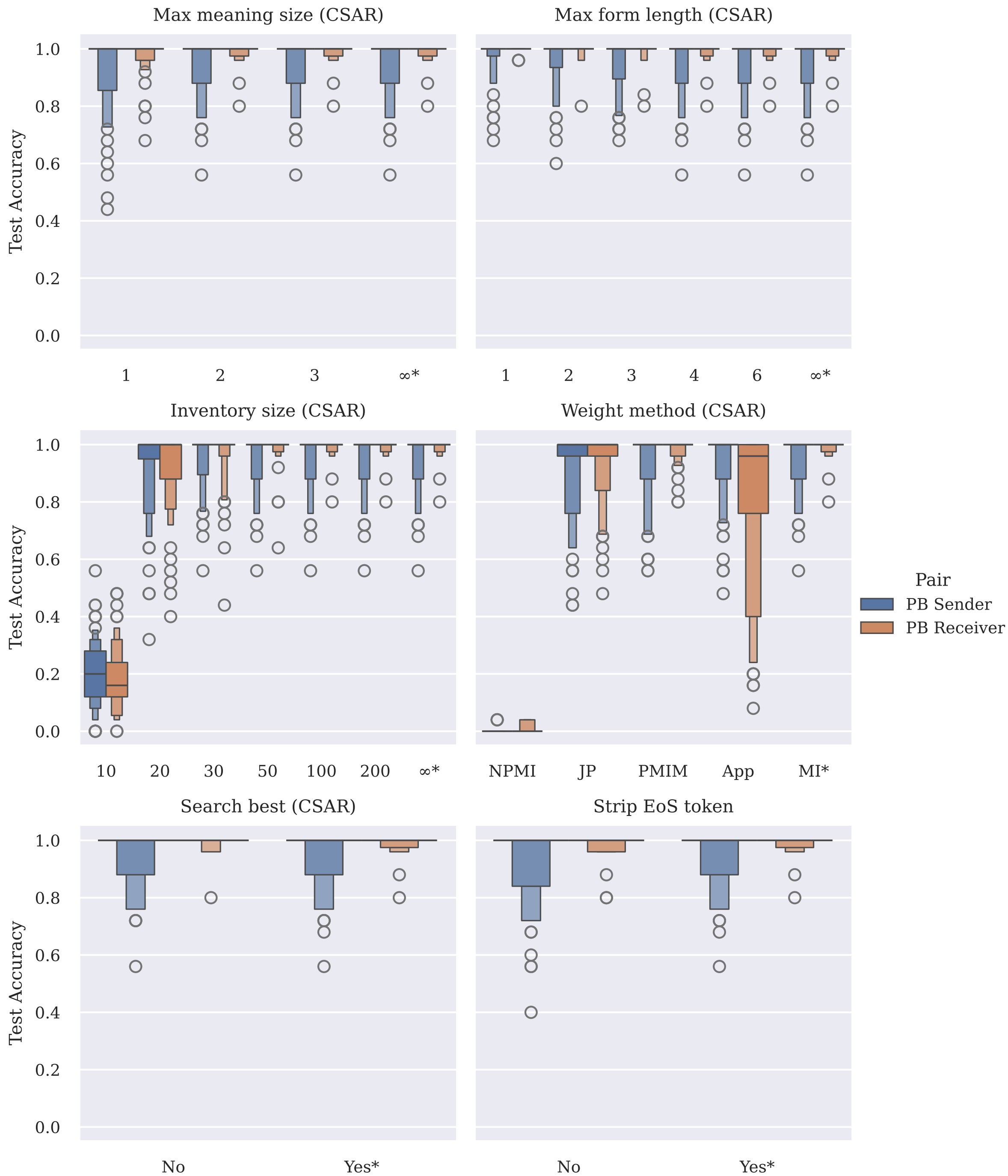


Figure F.5: Phrasebook agents' test accuracy across morpheme induction and phrasebook agent ablations. *Default values.

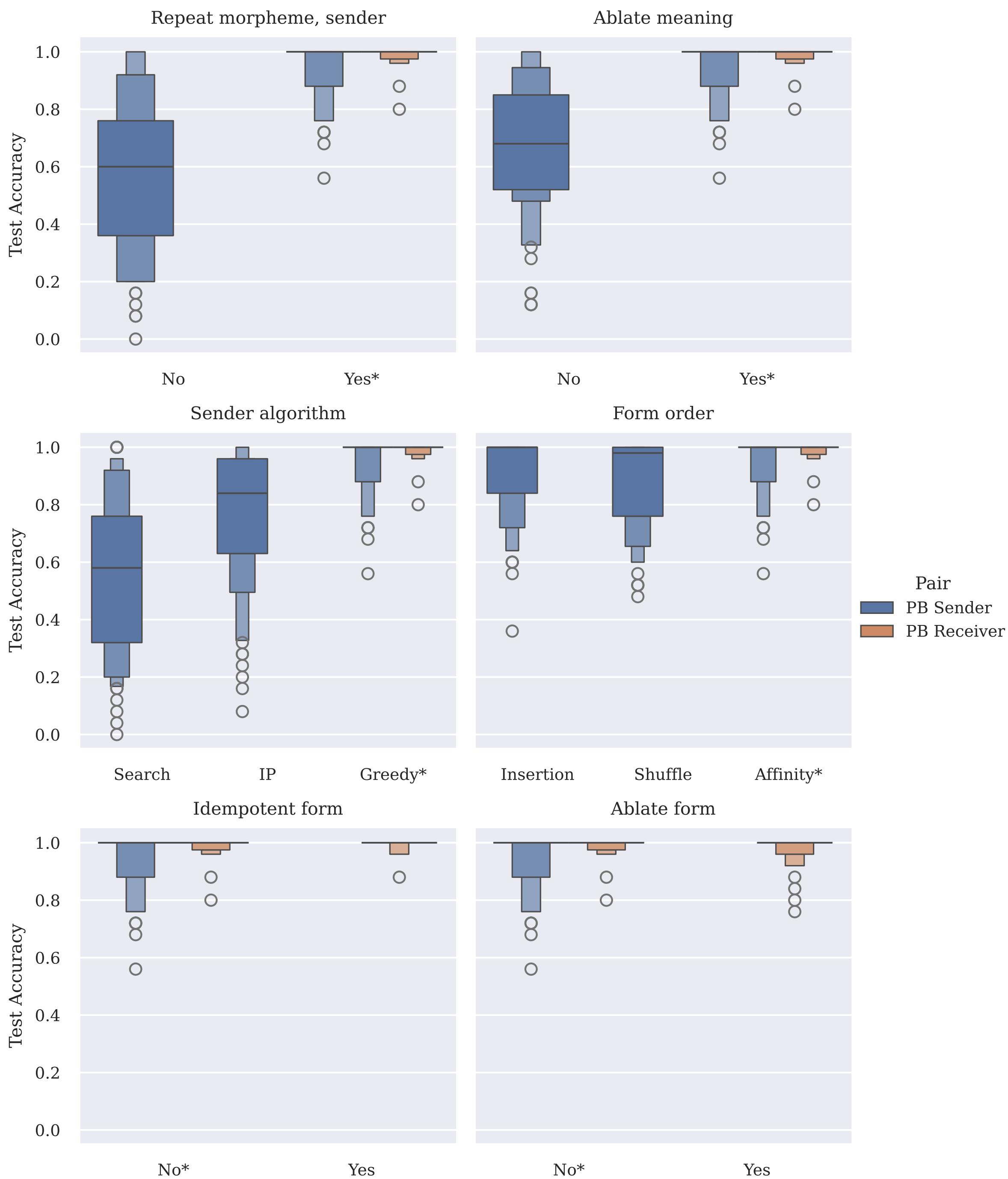
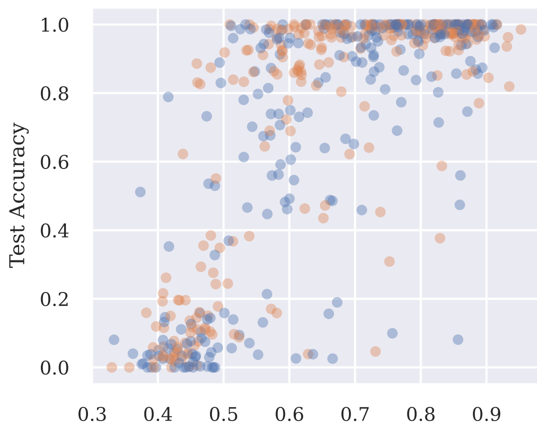
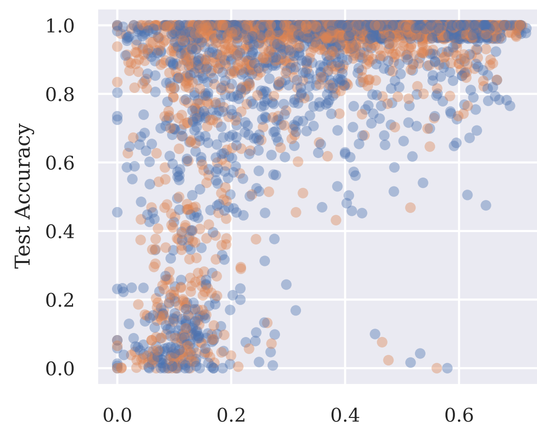


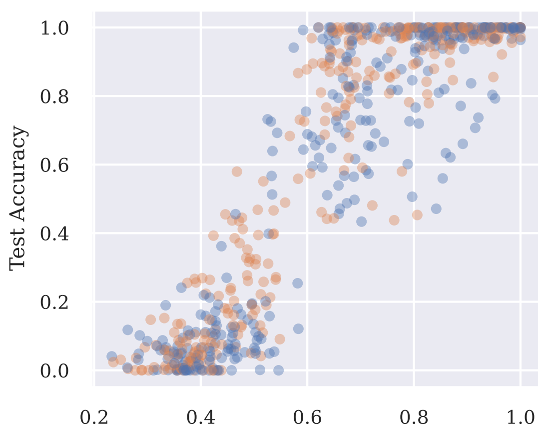
Figure F.6: Phrasebook agents' test accuracy across morpheme induction and phrasebook agent ablations. *Default values.



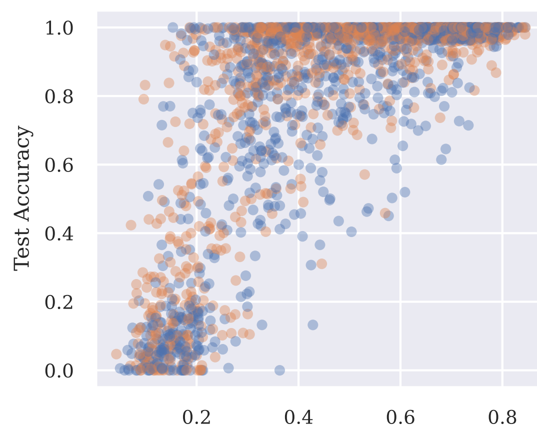
(a) Corpus topographic similarity ($R^2 = 0.57$)



(b) Bag-of-symbols disentanglement (bosdis) ($R^2 = 0.29$)



(c) Morpheme bijectivity ($R^2 = 0.85$)



(d) Morpheme mutual information ($R^2 = 0.54$)

Figure F.7: Plot of different compositionality metrics vs. phrasebook sender (blue) and receiver (orange) accuracy. Note the difference in number of points is caused by stratified sampling which keeps a constant number of values per bucket on the x-axis.