ALiEN
ERC Advanced Grant 2020
Part B2

Section a. State-of-the-art and objectives

a.1 Objectives

A decade of deep learning ([1]) has led to enormous progress in domains such a computer vision and natural language processing. Modern artificial neural networks successfully tackle difficult challenges such as image classification ([2, 3]) and machine translation ([4, 5]). As *deep neural networks* (DNNs) become better at specialized tasks, the natural next frontier is to use them together to address more complex challenges. Currently, this is achieved by manually combining specialized DNNs into complex architectures. For example, impressive results in visual question answering, the task of answering questions about visual scenes, have been achieved by architectures that combine convolutional networks for visual processing and recurrent networks for language processing ([6]). Moving forward, the manual approach will not scale up. In a not-too-distant future, specialized DNNs will likely be deployed as black boxes on various devices and appliances. To leverage their different abilities in user-specific applications, we need automated ways to make them share information, without requiring a machine learning expert to design a new interface protocol from scratch each time.

ALiEN takes natural language as a blueprint for a general DNN interface protocol. The main idea is to wrap a light set of communication layers around existing out-of-the-box DNNs. DNNs are then encouraged through appropriate training methods to use these layers to evolve an information sharing code. Unlike *ad-hoc* interfaces, the emergent "language" is optimized for its ease of learning by a large set of DNNs with different architectures, different pre-trained parameters and different functionalities. Once an interface language has evolved in a large community of DNNs, it will be possible to directly teach it to other DNNs in a supervised way, moving towards the ultimate goal of a **universal DNN language**. Unlike traditional manually coded interface protocols ([7]), ALiEN languages autonomously emerge when DNNs are trained to solve a cooperative task. They should thus be more flexible and scalable, for the same general reasons for which learning-based models tend to outperform hand-crafted systems ([8]): it is difficult to predict and hard-code everything that DNNs will encounter in the noisy, changing world in which they are deployed.

Ultimately, a fully functional protocol, just like human language, should be usable for a virtually infinite number of purposes. However, we must start from somewhere, and I think it would be a mistake to pursue a full-fledged, multi-functional communication code from the very start. Indeed, if we look at humans on both the phylogenetic and the ontogenetic scale, their language does not emerge all at once fully formed. Most researchers conjecture that, as a species, we went through a long phase in which our communication protocol consisted of a limited set of constructions with reduced functionality ([9-12]). Similarly, children pass through systematic stages in which they communicate by reduced means, such as the one-word and two-word phases ([13, 14]). Children and (presumably) our ancestors still manage(d) to get lots of things done with their *proto-languages*. In a similar spirit, I want to develop a **DNN proto-language**, limited in its capabilities, but possessing some crucial properties that make it *useful.* In particular, I will focus on two aspects: the ability to **refer** to objects in the perceptual surroundings of the DNNs, and **ease of transmission** across a varied set of DNNs.

Let's start with reference. Given its usefulness (e.g., to signal nearby dangers), reference to concrete objects plays a central role in the communication code of other primates ([15-17]). Similarly, object naming is central to child language ([18: ch. 4]). Referential nouns are also at the core of impoverished human communication systems, such as pidgins ([19: ch. 5]). It is hard to think of applications of DNN communication that do not require the ability to signal the things agents need to communicate about. Simple reference is of use on its own in various practical scenarios (such as the use cases I study in ALiEN). It is moreover a core building block for more ambitious applications. For example, if we want DNNs to give each other navigation instructions, we must first endow them with the ability to refer to landmarks along the navigation paths. For reference to be maximally useful, it should be **open-ended**, in the sense that DNNs are not constrained to communicate about things they've been exposed to in the language emergence phase (we cannot *a priori* foresee a closed set of things DNNs might need to refer to in the real world, except in very specialized applications,). Open-ended reference requires the ability to generalize to unseen object classes, still an open challenge in object classification ([20]). As I discuss in the WP1 outline below, I expect

discriminative training in referential games, not being tied to a fixed set of class labels, to lead to languages that pick up granular attributes of objects, thus supporting seamless extension to new class reference.

Ease of transmission is equally important. Unlike an ad-hoc interface between two or more DNNs, an ALiEN protocol should be a relatively stable representation that can be easily acquired by many different DNNs (just like large communities of humans converge on mutually-intelligible communication codes, such as the English language). I will first develop language emergence methods such that a language evolved with them has the following property: When a DNN joins a new community (e.g., a new smart appliance enters a household), it can acquire the language of that community (the protocol of existing appliances) much faster than if it had to build an interface to its new community from scratch. The core insight here is that, just like for humans ([21]), the more diverse the environment DNNs are exposed to during training is, in particular in terms of number of DNNs that are co-trained, the more general the languages they develop will be. A language that is more general in this sense should also be easier to pick up by a new DNN. Even better, varied training in terms of both inputs and DNN community size should lead to the emergence of languages that are not only easier to transmit (preliminary evidence in [22, 23]), but also "well-behaved" in other ways, for example: more accurate, interpretable, compositional (preliminary evidence in [23-25]). Further preliminary support for this claim comes from a recent pilot study in which we studied the effect of community-based training on learning, generalization and interpretability. The study adopted a symbolic-input version of the classic referential game (see WP1 below). We found that training a whole community of DNN agents together, instead of just a pair of agents, led to *faster convergence*, *better generalization to unseen inputs* and *a more compositionally transparent protocol*. The results were even stronger if further variety was added by letting different agents also engage in slightly different tasks. Thus, *training a large and varied set of agents together seems to lead to emergent languages that are both easier to learn and more transparent and expressive.* The converse also holds. In a recent experiment, we showed that languages that chanced upon the desirable property of compositionality were also significantly easier to learn by new agents, even when the latter were DNNs with different architectures with respect to the ones that initially evolved the compositional languages ([26]). The goals of large-scale reference and ease of transmission are thus not disconnected: those protocols that are better from a "linguistic" perspective (and thus better in terms of referential accuracy and generalization) are also easier to transmit. Conversely, easy-to-transmit codes also feature other linguistically desirable properties. Input and agent variety is key to the evolution of such well-behaved protocols. Note that communication protocols emerge in virtually all animal species under evolutionary pressures not unlike those simulated in ALiEN ([15]). There is no reason why DNNs should behave differently from natural species in this respect.

Summarizing, ALiEN has first and foremost the objective to **develop a general methodology to let DNNs evolve a communication protocol with two fundamental characteristics**: 1) **open-ended reference**: the ability to refer to things in the perceptual environment of the DNNs, independently of whether they have been observed before; 2) **model-agnostic communication protocol sharing:** the protocol can be used by standard DNN models without deep alterations to their inner makeup, and it is optimized for fast acquisition by DNNs that are added to a set of co-trained systems. These objectives will be first pursued in tightly controlled experiments focusing on generalization to new referents and across DNN agents. Having demonstrated the feasibility of the approach in this setup, I will work out concrete use cases of language emergence applied to 1) state-of-the-art pre-trained DNNs (showing how to add a communication layer to real-life out-of-the-box DNN models) and 2) a (simplified) home automation scenario (to demonstrate the usefulness of the emergent protocol in a realistic problem). ALiEN will thus lay the basis for a new approach to tackling complex tasks with DNNs, emphasizing fast interfacing of specialized models through emergent languages. The technology will be applicable nearly *as-is* in scenarios in which the protocol only needs to support referential information exchange (e.g., information retrieval applications). It will moreover lay the fundamental groundwork for a next generation of emergent languages that are beyond my current scope (e.g., languages supporting navigation instructions over an extended time span).

The other objective of ALiEN is to **systematically characterize the properties of emergent DNN languages**. As I want to re-focus the field form ad-hoc interfaces to transferable protocols, general tools to understand such protocols are of fundamental importance. First, if things don't work as expected, we need to decode what DNNs are communicating about (if they are communicating at all: cf. [27]). Moreover, we should find out which protocol properties are good predictors of desirable features, such as the ability to generalize to new objects and agents. In this way, we can design training methods that favor the emergence of such properties, or at least select languages that developed them. Understanding DNN language is also a pre-condition towards the long-term goal of building a language-based interface between DNNs and humans (to be clear: ALiEN lays the foundational work towards this goal, which is however well beyond its scope).

Furthermore, under the reasonable hypothesis that a general enough communication code will be more transparent, the best ALiEN protocols should also be *a priori* more interpretable, providing a new playground for interpretability research ([28]). From yet another perspective, analyzing emergent language has already proven useful to the exploration of the inherent preferences of deep architectures. For example, work from my team has shown how sequence-to-sequence DNNs have, all else being equal, a counterintuitive preference for longer sequences ([29]). The large number of setups developed in ALiEN will allow the most systematic exploration ever of deep networks' behavior from the perspective of the signals they develop. This type of analysis is relevant to "super linguistics" ([30]), the comparative study of the linguistic properties of different communication systems. An expressive DNN language is of great interest both with respect to the properties it shares with human language (as such properties will not depend on specific biological features of humans), and with respect to the aspects in which it departs from the latter (thus highlighting peculiarities of human language that cannot be a necessary consequence of the communicative constraints shared with DNNs). As an example of the former case, recent work from my team shows that the same lexicon complexity minimization constraint at work in human language is present in emergent DNN codes, pointing at its universality ([31]). Conversely, our recent work also shows that DNN languages might achieve referential generalization by other means than human-like compositionality, suggesting that the latter is not a necessary effect of the pressure to generalize ([26]).

a.2 State of the art

Deep Learning. Artificial neural networks, in their new "deep learning" incarnation ([1]), have swept the machine-learning board of the last decade, reaching super-human performance on tasks that were thought to be beyond the scope of current computational systems, such as the game of Go ([32]). A number of standard architectures have emerged as particularly fit for specific tasks, such as convolutional neural networks to extract visual information from images ([3]) or recurrent and transformer networks to process text ([4, 33]). It is customary, when pursuing a complex task, for example involving integrating vision and language, to use these out-of-the-box architectures as building blocks of a composite system. Indeed, virtually all the models used in the large variety of vision+language tasks surveyed in [34] follow this modular approach. In a bid to reduce the amount of supervision required for the end task, it has also recently become common to reuse not only architectures, but pre-trained *instances* of specific architectures. In computer vision, instances of successful architectures such as VGG, ResNet and Inception ([35-37]) trained on ImageNet object classification ([2]) are commonly used as components of systems performing tasks ranging from visual question answering ([38]) to relative camera pose estimation ([39]). This approach is even more ubiquitous in NLP, with pre-trained language models such as BERT ([40]) having been directly *developed* to be fine-tuned for different tasks. The issue of combining separate neural network modules also arises at a more granular level (e.g., specialized components handling different linguistic sub-tasks; [41]), and it will be increasingly encountered at the macro-level of coordination between complex architectures (e.g., neural networks controlling separate self-driving cars or home appliances). The Deep Learning community has recognized the scalability issue involved in ad-hoc fine-tuning of pre-trained models. Several recent papers proposed specialized light layers to fast-tune such models, while leaving the core pre-trained parameters unchanged ([42-44]). ALiEN is related to this research line, with i) a specific focus on information exchange between models that are jointly solving a task (as opposed to fine-tuning a single architecture for a new task), ii) the explicit goal of developing communication protocols that can generalize across different (pre-trained) models and iii) special emphasis on decodable protocols.

Representation learning. The code evolved by ALiEN DNNs to exchange information is obviously a representation of this information. Thus, ALiEN is closely related to *representation learning* ([45]), a field focusing on the development of computational representations with various desirable properties. Some of these properties, such as supporting generalization and interpretability (often pursued by means of *disentanglement:* [46, 47]), are also central to my pursuit. What characterizes ALiEN within this more general domain is the emphasis on *interface* and *shareable* representations. The idea of using evolved representations as a target for supervised learning is also, to the best of my knowledge, unique. Before the emphasis shifted to representation *learning*, the problem of information sharing was addressed through hand-crafted protocols (e.g., [7]). It is however not clear how such pre-determined protocols could be adapted to DNNs, nor that they would possess the large-scale flexibility I am pursuing.

Language evolution, communication games. A setup in which two or more agents must evolve a communication protocol in order to accomplish a goal (often, a referential game such as the one in my WP1) has been extensively explored by a number of research traditions, including linguistics, philosophy, cognitive science, AI and game theory (e.g., [10, 15, 48-56], among many others). Some of the computational work in this area has a purely theoretical bent, deliberately using simple agents and setups (e.g., agents consisting of

probabilistic policies over 2- or 3-input states), in order to focus on mathematical properties such as the equilibria of the communication system. Other work is more empirical, sometimes using, as in ALiEN, computer vision (e.g., [53]) and neural networks (e.g., [49]). Despite many similarities, my focus is different, as I am not interested in multi-agent communication in general, but in how existing DNNs, already optimized for various functions, can communicate with each other through an emergent language. Still, I am inspired by work in this area, and I expect to give back to it. For example, one of the most robust results in human and computational simulations of language evolution, namely that cultural transmission plays an important role in stabilizing a transparent language (e.g., [57-59]), encourages me to adopt large mixed-community training as a tool to evolve better languages. Conversely, ALiEN defines new problems that should be of interest to theoreticians in the area. For example, the impact of mutants on the stability of a signaling system has been extensively studied in evolutionary game theory ([15]). ALiEN is interested in the related but distinct and novel problem of what makes the language of a community well-suited to be assimilated by a "mutant" (a DNN from a different community) as smoothly as possible.

Multi-agent systems. ALiEN also fits into the wide area of multi-agent systems research, particularly cooperative learning and the decentralized POMDP setup ([60-62]). Recently, the problem of optimizing the behavior of a *diverse* set of cooperating agents has attracted interest in this field (e.g., [63]). I will stay on top of this literature, as it relates to the ALiEN problem of developing a shared protocol for agents with different functions and structures.

Emergent DNN language. The advent of deep learning has led to a new wave of computational simulations of language emergence with the ultimate goal of teaching DNNs to autonomously interact with each other and with us through communication (foundational work includes: [64-67]; Lazaridou and I have recently surveyed the area in [68]). ALiEN belongs to this research line, bringing about a novel high-level vision and scope. In particular, I identified large-scale, perceptually realistic reference and ease-of-learning by new agents as the two pillars for a *practical* DNN proto-language. There is some work on emergent communication with realistic pictures as input (including the early study from my team in [66]). However, the topic has not been recently pursued, with current work in the area privileging artificial symbolic input. In particular, I am aware of no work testing the generalization abilities of emergent DNN languages with realistic referents. Our results in [69] suggest that small-scale experiments with real pictures will result in *ad-hoc* codes without any generality, stressing the need to scale up. On the other hand, we recently showed (albeit with symbolic data) that rich input variety might suffice to precipitate generalizing languages ([26]). Concerning protocol sharing, previous work has asked whether cultural transmission leads to "better" languages, e.g., languages that are more compositional or easier to learn (the answer is positive: e.g., [23-25]). In ALiEN, the question is rather which conditions will *a priori* make languages easier to transmit across communities. There is proof-of-concept work showing that DNNs can spread protocols across separately trained communities ([22]). Some of my recent work indicates that languages that chance upon a compositional encoding are easier to learn and use even for DNNs with different architectures ([26]). The question of fast protocol sharing is however largely unexplored. To the best of my knowledge, the idea of teaching "good" emergent languages to new agents in a supervised way, although related to classic insights from the iterated learning literature ([57-59]), is completely novel. To make my pursuit feasible, I adopt the "proto-language" view I defended in the *Objectives* section above. ALiEN will ignore important aspects of full-fledged linguistic communication addressed in the recent literature, such as deciding *when* ([70]) to communicate what *to whom* [71], adding communication to dynamic grounded scenarios ([72]), non-aligned interests [73] and direct mutual intelligibility with humans ([74]). I expect that these topics will be easier to address after ALiEN delivers its foundational results.

Interpretability and BlackBox NLP. I discussed above reasons why understanding the emergent codes is crucial for my enterprise. Conversely, building a general-purpose interface protocol between DNN modules (as opposed to ad hoc connections between specific architectures) will facilitate the study of their behavior, and thus constitutes a contribution to interpretability research and explainable AI in general ([28, 75]). To decode ALiEN languages, I will build upon several approaches to emergent language analysis that have recently been proposed. These include principled ways to measure properties such as compositionality (e.g., [26, 76], the former from my team), learning a translation model from "neuralese" to natural language ([77]) and interventional studies measuring the causal effect of message manipulation ([27, 78], the latter from my team). A large research community focusing on understanding how DNNs process natural language has recently coalesced around the BlackBox NLP workshops ([79, 80]; I contributed extensively to the area, e.g.: [81-86]). This research line investigates how internal DNN components process a known linguistic protocol (e.g., the English language). In ALiEN, the challenge is to analyze the emergent protocol itself. Still, many of the same methods can be applied (e.g., in [78] we applied diagnostic classifiers, a popular tool in

BlackBox NLP, to decode the referents of an emergent language). More generally, emergent communication provides an alternative way to investigate the linguistic skills of DNNs, with a clearer focus on the communicative function of language, complementing the more standard approach of feeding DNNs large amounts of human-produced text and analyzing the statistical generalizations they extract from it.

Super linguistics. Comparative communication research has a rich tradition, and it is increasingly adopting sophisticated tools from linguistics to understand non-human communication (e.g., [15-17, 30, 87-89]). As already suggested by McCloskey in 1991 ([90]), we can look at artificial neural networks as a different "species", whose behavior can provide original insights into the space of possible solutions to various cognitive challenges. DNNs are increasingly studied from this perspective (e.g., [91-94]). Letting them evolve signals to solve a shared task is a tremendous opportunity to get new evidence on which aspects of communication are universal, and which vary from species to species. I am exploring this approach in my current work (e.g., [95, 96]), and ALiEN will provide a wealth of new data for investigations in the area.

a.3 Progress beyond the state of the art

ALiEN proposes a **paradigm shift** in managing complex deep learning architectures by evolving **general-purpose interface protocols** that are robust to variations in input information and in the specifics of the neural network components being connected.

To achieve this novel goal, ALiEN relies on the know-how we recently accumulated in the study of emergent DNN language (and, more broadly, language evolution simulations and multi-agent communication). However, it pushes for radical advancement in the area, tackling the issues of **large-scale reference in a perceptually rich world** and **easy transmission across DNNs**. With respect to both goals, the focus is on fast generalization to unseen scenarios, supporting the robustness required by real-world applications. This focus is ambitious but deliberately narrow (following the idea of a functional "proto-language"), in order to maximize feasibility.

ALiEN language emergence is a form of representation learning. The project significantly innovates in this area by introducing the idea of "**culturally" transmitted representations** (transmitted both via community mixing and direct supervision on emerged languages). The idea of a general-purpose code that might be **intrinsically more interpretable**, and for which we could learn a translation model onto natural language, will also constitute an advance towards explainable AI.

The large-scale language analysis effort will further provide **general tools** not only for **emergent language research** and **representation learning**, but also for **linguistic and cognitive studies** of communication and, again, in the domain of **AI interpretability**.

a.4 Impact

ALiEN will impact all the research communities mentioned above. It should trigger a shift in the development of complex **deep learning** architectures **from ad-hoc interfaces to flexible connectivity** and, ultimately, **genuinely autonomous AI agents** able to interact with each other and with us. At the same time, the new emphasis on persistent, shared representations provides new perspectives and defines new problems in **representation learning** and **interpretability studies**. ALiEN gives **cognitive science**, **language evolution research** and **linguistics** a new body of evidence on **the limits of communication**, and new tools to analyze it. These tools might also prove useful to characterize other types of communication systems, such as animal signaling, or even natural languages themselves.

From an applied perspective, I foresee **coordination between DNN-controlled devices** to become one of the major challenges in the industrial deployment of AI in the coming years. Beyond the **multiagent information retrieval** and **home automation** scenarios simulated in ALiEN, the coordination problem is pervasive. Indeed, a communication-based approach to coordination has recently been proposed for **self-driving cars** (e.g., [97]) and **robot arms** ([98]). As other classic domains in computer science and information technologies scale up to large communities of actors including machine-learning components (e.g., in **communication networks** or **finance**: [99, 100]), the problem of a scalable and flexible coordination protocol will become more and more pressing. **ALiEN puts Europe at the forefront of this important next frontier in AI**. Fittingly, it does so by **building on a long European tradition of studies in language evolution**, communication games and cross-species linguistics.

Section b. Methodology

b.1 Work plan

High-level rationale for work package organization

Computational simulations are organized into 3 Work Packages. **WP1** lays the foundations of the project by exploring under which conditions DNN emergent language can generalize to new referents and agents in tightly controlled experiments. In practical applications, however, successful referential communication on its own will not suffice, and the emergent language must prove its usefulness by adapting to existing state-of-the-art DNN architectures and application-specific demands. **WP2** explores how emergent communication supported by a light interface layer can help interfacing out-of-the-box pre-trained DNN models. **WP3** looks at language emergence in a home automation scenario, considering some challenges that will often arise in practical contexts, such as the need to denote multiple referents in the same message, bidirectional information exchange and integrating short-term and long-term world knowledge. **WP4** runs in parallel to and complements the experimental work packages, decoding and analyzing their emergent languages. **WP5** prepares the software developed in the project for public release. **WP6** oversees project management, including acquiring infrastructure, hiring personnel, steering the project, and organizing dissemination.

NB: In what follows, I skip important technical issues, such as architectural details or hyperparameter search, in order to focus on the core outline of each experiment. I led the development of the EGG toolkit for emergent DNN language simulations ([101]). EGG provides a solid technical core that can be extended to implement the experiments schematically described below.

Work Package 1: Growing General Deep Network Languages (all project personnel)

WP1 studies the emergence of languages that generalize across inputs and agents in one of the simplest setups where agents need to share information about referents in other to accomplish a task. Namely, I use a classic referential game ([15]), extended to realistic input images and DNN agents. A round of the game is illustrated in Fig. 1(a) of B1. A DNN is randomly picked to play *Sender*, another *Receiver* (the same DNN might play Sender and Receiver in different rounds). The Sender gets a natural image (the *target*) as input, and it produces a message. Receiver gets the message and sees two or more images in random order (target and *distractor*(*s*)). Receiver points at an image, and error is backpropagated based on whether the chosen image is the target or not. There is no direct supervision on the message itself. Basic agent architecture: see Fig. 1 here. Communication: We explore 4 types of messages, increasingly close to those of natural language: 1) a dense continuous vector; 2) a sparse continuous vector (obtained, for example, by softmax filtering with low temperature); 3) a single discrete symbol; 4) a variable-length sequence of discrete symbols (the latter case is illustrated in Fig. 1 here and in all the figures of B2). Optimization: Receiver target guessing is straightforwardly implemented with the cross-entropy cost function. When the communication channel is continuous (cases 1/2 above), the whole architecture is optimized via backpropagation with standard gradient descent techniques. Gradient backpropagation is not possible through discretely sampled symbols (cases 3/4). Following common practice in DNN language emergence research, the Receiver will then be trained via standard backpropagation, whereas the Sender will get its error signal either through the REINFORCE method ([102]) or via the Gumbel-Softmax relaxation technique ([103, 104]). Testing regimes: In *direct testing*, DNN weights are fixed after training, and system accuracy is evaluated on new inputs. In *language tuning*, the linguistic layers of the DNN (corresponding to the RNNs in the architecture depicted in Fig. 1) continue being updated in the test phase, and the main success metric is time until training-set-level accuracy is reached. Data: standard image databases such as ImageNet [105] and Visual Genome [106]. Note that we will not use the *annotation* provided with these databases to train the DNN agents (although we might use it for analysis). The methods developed here can thus be easily adapted to any relevant image input, including unannotated data.

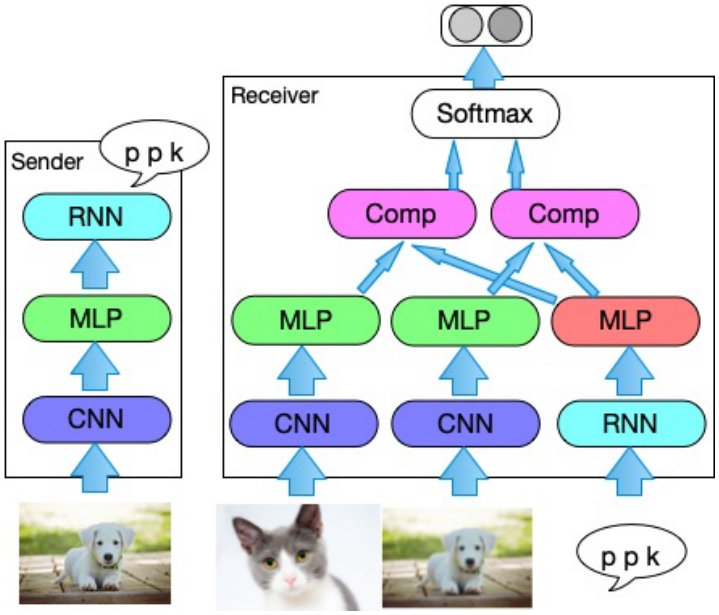

Fig. 1 ***Schematic illustration of WP1 agent architecture.*** **When an agent is used as Sender, a CNN processes the target image (here and below, the last layer of the CNN is not necessarily a classification softmax). The resulting representation is elaborated by a multi-layer perceptron whose output vector conditions a RNN generating a symbol sequence (the RNN is replaced by the appropriate components for different types of messages). In Receiver role, input images (two in the figure) are processed by a CNN-MLP pipeline, outputting a vector per image. The message is processed by a RNN whose final state feeds a MLP generating a vector to be compared to the image representations through a comparison operation (that could be as simple as a dot product, or consist of another MLP). One score per image is produced. The latter are concatenated and softmax-converted into a probability distribution over target position in the image array. During training, the cross-entropy cost function compares the resulting output vector to the ground truth target location. Components sharing at least some weights are color-coded. Some components are (partially) shared across roles (but *not* across DNN agents). Various details ignored (e.g., binary input signaling agent role in episode).**

Experiment Set 1: *input generalization.* 1) Test performance is measured on new instances of training classes, as in standard image classification (this was the setup of our work in [66], and it should be easy to replicate). 2) Instances of new classes are presented at test time (e.g., at test time the target is a selfie stick, an object that was not seen during training; cf. Fig. 1(b) in B1). This 0-shot generalization setup is extremely challenging ([20]), although it is fundamental for practical applications where there is no pre-determined set of categories the agents might encounter. Unlike in standard supervised image classification, our DNNs must learn to *discriminate* images without relying on a closed set of labels. I conjecture that, if the training input is sufficiently varied, it will be better for agents to use messages denoting discriminative attributes such as colors, shapes or texture, in addition to high-level classes ("the target is the furry thing with pointy ears"). Such class-agnostic attributes should then generalize to instances of new classes. If necessary, various training strategies can be used to encourage this behavior. For example, training DNNs to discriminate between targets and distractors sampled from the same category (e.g., two cats) should lead the agents to develop a more granular language, better suited for generalization. If we want to avoid relying on image labels to pick training examples (as we'd have to do to choose same-class distractors), we can also select distractors based on automatically computed high visual similarity to the target.

Experiment Set 2: *agent generalization*. An agent *community* includes multiple DNNs that are trained together (that is, in each training round two DNNs from the community are randomly sampled to play the game together). 1) Intra-community. Some pairs of agents are never sampled together during training. If the community developed a shared language, they should be able to communicate 0-shot at test time. 2) Across communities. At test time, an "immigrant" agent trained in community X plays the game with randomly sampled agents from community Y (Fig. 1(c) in B1). Language tuning is applied to the X agent. How long before it adapts to its new community? Is this faster than (i) fine-tuning X's whole architecture or (ii) training a new agent from scratch? Does the immigrant forget its original language? What happens if the Y agents are also language-tuned? 3) Different architectures. Same paradigms as in 1) and 2), but now the

agents differ (within and/or across communities) in their architectural details: e.g., different CNN and RNN models, different number of hidden layers. (How) does this affect the results?

Experiment Set 3: *putting it all together*. Exploring simultaneous generalization across inputs and agents at training and/or test time. For example: Immigrant agent X (cf. ES2) is exposed to a new community *and* to new categories not familiar to its source community: how does this affect adaptation?

Experiment Set 4: *ALiEN supervision*: A large DNN community X is trained to convergence on the referential game. Supervised data are generated by recording the messages generated by a majority of agents in response to various inputs. New communities Y and Z are (separately) trained by mixing rounds of referential game playing and *supervised teaching to produce/process community X messages.* 1) Do the Y and Z languages drift away from X? 2) Does convergence occur faster in this setup than without supervision? 3) Can Y and Z agents communicate with each other zero-shot? How about with X agents? 4) How does supervised training affect different DNN architectures, and generalization to new inputs?

Milestones:

Referential game with standard test set	M12
0-shot input generalization	M36
Intra-community new agent pair communication	M36
Communication across communities	M48
Sharing languages with mixed supervised/game-based training	M52

Work Package 2: ALiEN Communication for Pre-trained DNNs (PI, PD1, PhD1)

ALiEN communication should be agnostic to the specifics of the models relying on it. I pursue the vision of a communication protocol supported by light layers that can be easily added to any out-of-the-box DNN, without the need to re-train the whole model. Demonstrating the feasibility of this approach is an important step towards a future in which arbitrary DNN-driven devices can quickly coordinate through language (NB: this also implies important innovations in hardware and software design well beyond the scope of ALiEN). WP2 moves in this direction by studying *communication between pre-trained state-of-the-art DNNs processing natural language and visual data.* As discussed above, in both domains pre-trained DNNs are increasingly used as readymade modules to be joined together in ad-hoc architectures to perform complex tasks. Although we will finalize the list of state-of-the-art models to be used at the start of WP2, examples might include ImageNet-pre-trained instances of the ResNet, Inception and VGG visual processing networks ([35-37]; note that, when used as general-purpose visual processors, the topmost softmax classification layer is often chopped off). On the text processing side, we can use BERT and other pre-trained language models ([40, 107]). Multiple versions of the same models trained in different ways will be added to the set, considering both large differences (e.g., language models pre-trained on languages other than English) and minor ones (e.g., the same model pre-trained on smaller and larger datasets).

As the focus is on the emergence of a shared code, I use a simple task similar to the referential game of WP1, which might still be of direct interest in (multimodal) information retrieval applications such as caption or image retrieval. I'll refer to the input referents used here as *documents*. A document is a multi-modal object composed of a picture and a text caption (possibly, multiple captions in different languages). Vision models receive only the picture as input. Textual models are presented with captions only (in the appropriate language). In each round of the game, the *Querier* DNN is given a *target* document in its modality, and it produces a message. The *Retriever* DNN gets the message, and it is exposed to a randomized list of documents, including the target and one or more *distractors* (all in the appropriate modality). The Retriever must point at the position of the target in the list. The task is schematically illustrated in Fig. 2 of B1 for the case where there is one distractor only. Basic agent architecture: A pre-trained DNN is augmented with extra layers to produce and process messages (*EmeLang Decoder* and *Encoder*, respectively, in Fig. 2 of B1). The nature of these modules depends on message type. If the latter consists of a sequence of symbols, the modules will be MLPs followed/preceded by RNNs (for Encoder/Decoder, respectively). This is analogous to what I envisage for WP1, as in Fig. 1. When a DNN plays Retriever, it needs a mechanism to select the target (*Select Module* in Fig. 2 of B1). This will be analogous to the Comparison module in WP1 (Fig. 1 here). I am particularly interested in using a parameter-free comparison operation, such as the dot product, so that the only layers to optimize in the simulations are those in the language modules. Communication: As in WP1. Optimization: Essentially as in WP1, but only the parameters of the EmeLang modules are updated

(the pre-trained models are treated as unalterable black boxes). Data: standard image+caption databases such as Microsoft COCO [108], possibly multilingual [109].

Relation between emergent ALiEN language and natural language: There are two types of "language" in WP2. On the one hand, text-based DNNs will process captions in *natural language*. On the other, all DNNs will communicate with each other through their *emergent code*. This is illustrated in a cartoon-like way in Fig. 2 of B1, where text-based DNN *NatLang2* reads the English caption "*Pizza has been baked*" and produces the emergent-language message "*ww*". I hypothesize that it is better to let DNNs develop their communicative code rather than forcing them to coordinate through natural language. Independently of this, it is not clear where the right data to train DNNs to use natural language in this interactive scenario would come from (vs. the virtually infinite amount of data we can generate to let DNNs induce their own language). An interesting question for WP4 (protocol analysis) is to what extent an emergent language is affected by the fact that some of the agents evolving it are NLP models (for example: what happens if both Querier and Retriever are text processors pre-trained on English: will the emergent language look more like English?).

Experiment Set 1: *one community*. At each training round, two DNNs are randomly picked as Querier and Retriever, respectively (two rounds with different models are shown in Fig. 2 of B1). Only the EmeLang layers of the models are updated. At test time, again two DNNs are randomly paired, and accuracy on a held-out document set is computed. Does this work? How does community-based EmeLang training compare to other interfacing strategies, such as adapter methods specifically trained for each DNN pair ([42-44]) or full-architecture tuning? Are some DNNs easier to pair than others? Does the same EmeLang component design work for all models?

Experiment Set 2: *agent generalization*. 1) Intra-community. Random DNN *pairs* are held-out during training, and only coupled at test time (they are however trained with other DNNs). Can they communicate zero-shot? Does this depend on the underlying models? 2) Across communities. Like in WP1, an immigrant agent is brought into a host community. How long before the immigrant adapts to its new community through language tuning? Does this depend on the immigrant model? What's the effect of having models of all kinds in both source and target communities vs. systematically different architectures? Is the ALiEN approach faster than training specialized connections for each possible pair from scratch?

Experiment Set 3: *ALiEN supervision*. 1) WP2-to-WP2. A community is language-tuned as in ES2. The resulting code is used as a source of supervision for other models, whose training mixes imitation learning of the existing ALiEN language and performing the multi-modal document retrieval task. Is this better than training on the latter objective alone? Does it help fast agent generalization? If supervision works, does it help different DNNs equally, or is it model-dependent? 2) WP1-to-WP2. Same experiments, but now we use languages that emerged in the WP1 referential games for supervision. This is especially important, because, if it works, it suggests that we could develop a "universal" language in the lab, and then use it to pre-train the interface layers of DNNs destined to various applications.

Milestones:

Basic multimodal retrieval setup	M30
Intra-community new agent pair communication	M48
Communication across communities	M50
Sharing WP2 languages with mixed supervised/task-based training	M54
Sharing WP1 languages with mixed supervised/task-based training	M58

Work Package 3: The Grocery Challenge (PI, PD1, PhD2)

Any real-life application will impose specific constraints on top of the basic need for accurate referential communication. WP3 explores a use case loosely inspired by home automation ([110]), which demonstrates how ALiEN protocols can help agents to accomplish a practical goal, namely, keeping a simulated refrigerator well-stocked. This scenario does not feature, of course, all possible demands that will come up in applications, but it does add some complexities that will often arise in practice: i) reward is given for achieving a task distinct from successful communication, and different communicative strategies might be used to solve it; ii) communication pertains to multiple object classes and instances, with agents seeing inputs in different modalities (pictures vs. symbolic representations); iii) two-way information exchange (possibly leading to multiple conversation turns) is necessary for high performance; iv) short-term information about the current situation must be integrated with long-term general knowledge.

Experiments revolve around a task I call *the Grocery Challenge*. In each episode, a *Fridge* agent gets as input the current state of its stock, represented as a randomly ordered list of natural images (Fig. 3 in B1). A *Shopper* agent gets as input a full list of (symbolically represented) product categories with their current price. See ES1 and ES2 below for different episode dynamics. At the end of the episode, the Shopper must specify, for each product category, the number of items to be bought. If the overall price of this shopping list exceeds a fixed budget, the episode ends with negative reward. Otherwise, the purchased items are added to the Fridge, and positive reward is shared by both agents in function of the final state of the Fridge. Importantly, goods have different values (e.g., milk is more important than peanut butter). At the same time, the value of a product decays logarithmically with item count (if we have 4 milk bottles but no peanut butter, it is better to buy a jar of the latter instead of more milk). Fridge and Shopper have access to partial information (current stock levels and prices, respectively), so they must collate knowledge through communication. Communication: Same as in WP1. I will also test the potential benefits of pre-training the communication layers using WPs1-3 languages for supervision (not further discussed below). I/O representations and architecture (main features): Fridge uses a RNN-controlled CNN to read the input image sequence representing its state. Shopper gets product prices as a fixed-dimensionality real number list. The shopping list it produces is also a fixed-dimensionality vector of real values, rounded to nearest integers to denote product quantities. Both agents use RNNs/MLPs as appropriate to produce/parse messages. Optimization: Reward is backpropagated with policy gradient methods [111, ch. 13]. Data: as in WP1.

Experiment Set 1: *simplified setup***.** Fridge sends a single message to Shopper. Shopper must pick a single item to buy. Agents must learn the relation between product instances and classes, estimate product values over multiple episodes, and integrate this long-term information with episode-specific quantities and prices. Different strategies might arise, e.g., Fridge directly tells Shopper what to buy vs. Fridge provides Shopper with current inventory information. Given knowledge inferred over multiple episodes about product values, degenerate strategies in which communication is ignored might also arise, e.g., Shopper always picks the highest-value item, independently of current Fridge stock ([78]).

Experiment Set 2: *full-fledged setup* (Fig 3 in B1). The agents can exchange multiple messages (starting with Fridge), until one of the two emits an end-of-conversation symbol, at which point Shopper outputs the shopping list. This setup is ambitious, but it is an extension (in a different context) of the one Bouchacourt and I developed in [78], and I am thus confident I can make it work. Again, different communication strategies might arise, e.g., Fridge provides a full description of its current state to Shopper, and the latter prepares the shopping list on its own vs. Shopper provides the full price listing to Fridge, and Fridge responds with a linguistically-encoded shopping list. Intermediate strategies are also possible (e.g., Fridge only asks Shopper for prices of low-stock items), as well as degenerate ones, like in ES1.

Experiment Set 3: *agent generalization*. This simulates scenarios in which a household buys a new appliance, it replaces an old one, or there is more than one appliance of a kind (e.g., multiple Shoppers linked to different stores). 1) Generalizing across communities. Separate communities of agents are trained (in each round, random Fridges and Shoppers from the same community are paired). The main success measure is whether two appliances from different communities converge faster to high reward through language tuning than agents trained from scratch through an ad-hoc interface layer. 2) Different architectures. A single community might include, for example, Fridges with different kinds of CNN designs, as well as agents with different numbers of hidden layers. Agents are identified to each other by unique IDs. The focus is on language analysis after convergence: does a single language emerge across architectures? Do communication strategies change depending on architectural differences? For example, could a Fridge learn to give a full shopping list to a "dumb" small-capacity Shopper, but simply provide information on low-stock items to a "smart" larger-capacity one? 3) Different communities and architectures. Putting together the previous setups, we finally test cross-community communication when all communities include agents with different architectures (either spread across communities, or, in the most challenging setup, with different architectures split into different communities).

Milestones:

Simplified Grocery Challenge	M30
Full-fledged Grocery Challenge	M42
Agent generalization in Grocery Challenge	M54

Work Package 4: ALiEN Analysis (PI, PD2, PhD3)

WP4 runs in parallel to WPs1-3, analyzing the corresponding protocols, while building general tools for emergent language research. Activity 1 develops the necessary analytical tools. Activities 2-4 build upon them to focus on specific properties of the emergent languages.

Activity 1: *building general analysis tools.* I will collate, systematize and extend methods from the fields of emergent language/language evolution, representation learning, BlackBox NLP and linguistics, in order to develop a battery of tests to thoroughly profile emergent languages. Relevant methods include: diagnostic classifiers that probe whether a property of interest is present in the emergent code ([83]); methods to quantify systematic relations between input (or output) representations and messages ([112]), to be extended to also measure inter-language similarity; measures of disentanglement, compositionality and hierarchical structure in the way messages encode input or output information ([26, 76]); causal intervention methods that manipulate inputs and messages, both to assess the effective degree of communication between agents and to measure mutual intelligibility between agents and languages ([27, 78]); techniques to translate emergent language into natural language ([77]); methods to segment symbol sequences into "morphemes" or words ([113, 114)]; methods to measure the complexity and efficiency of languages, based on ideas from both linguistics ([29, 115]) and information theory ([31, 116]). Recall that ALiEN explores four types of messages: dense and sparse vectors, single discrete symbols and discrete symbol sequences. Not all analytical techniques apply to all settings, and direct comparison is problematic. Considerable attention will be paid to how to best characterize the different message types, and when it is appropriate to compare them.

Activity 2: *analyzing generalization.* Are there characteristics (e.g., degree of compositionality, low complexity, low symbol interdependence) that correlate with the ability of languages to generalize across agent architectures, roles, inputs, tasks and communities? Can such characteristics be used as predictors of the ability to generalize? Can their emergence be encouraged during training? If so, does this lead to better languages? Can we predict which protocols are easier to transfer across communities, and best for supervised training?

Activity 3: *interpretability*. Interpretability scores include measures of disentanglement, of consistent input-message association, of segmentation ease of discrete messages into meaningful parts and of decomposition ease of continuous signals into meaningful independent factors. Which training regimes, architectures and environments lead to more interpretable languages? Are there ways to encourage more interpretable languages to emerge? Is there a correlation between interpretability and other desirable properties of an emergent language (accuracy, generalization)? Are discrete languages inherently more interpretable than continuous ones? Can a mapping between emergent languages and English be induced?

Activity 4: *ALiEN vs. humans*. A3 focuses on ease of emergent language decoding, A4 focuses on similarities and differences with human language. These are related but different questions. For example, a dense-vector representation might be highly interpretable thanks to an easy factorization into meaningful components, while not being natural-language-like in the least. Questions: 1) Do ALiEN languages exhibit duality of patterning ([117, 118]), the human language property whereby meaningful units (words or morphemes) can be further decomposed into meaningless ones (phonemes)? 2) Do emergent languages develop words referring to generic categories that abstract away from instance-specific details? Can we distinguish different classes of words (e.g., proper-noun-like, generic-noun-like, and adjective-like)? If so, are such classes marked by rudimentary forms of morphology or syntax? 3) How does generalization to new inputs work? Do we observe the roots of compositionality in how emergent languages refer to new items? 4) Are emergent languages efficient in some of the many ways in which human languages are argued to be ([119])? Specifically, at the form level, do agents evolve codes minimizing average message length ([120])? At the meaning level, do they partition their perceptual worlds in complexity-reducing ways (e.g., requiring as few word types as strictly needed to accurately communicate? [121]). Are agents *pragmatically* efficient ([122]), only transmitting information that is not redundant given the context? 5) Are natural-language-like properties more likely to emerge when emergent communication involves DNNs pre-trained on English (or other natural languages; cf. WP2)?

Milestones:

Analytical infrastructure ready	M36
Intervention study: encouraging the emergence of generalization	M48
Intervention study: favoring interpretability	M48
Characterization of ALiEN language generalization means	M60
Assessment of interpretability of different ALiEN languages	M60
ALiEN vs. human languages: a final report	M60

Work Package 5: ALiEN Software (all project personnel)

ALiEN intends to stir further interest in emergent DNN language by providing well-documented software to run and analyze simulations. The Grocery Challenge will be released as a standalone environment to aid systematic comparison of different approaches. Here, I can build on my long experience of providing software and data to the community (cf. http://marcobaroni.org/tools_and_resources.html), and in particular in leading the development of the EGG toolkit for DNN language emergence simulations (https://github.com/facebookresearch/egg). Software development is mostly conducted in WPs1-4. WP5 focuses on preparing resources for release, following good software engineering practices.

Activity 1: *ALiEN toolkit development*. Two main releases are envisioned. The first will primarily support computational simulations, paying special attention to efficiency considerations, e.g., parallel GPU training of agent populations. The second will also include the analytical tools produced in WP4.

Activity 2: *Grocery Challenge environment development.* The WP3 environment will be packaged with a set of pre-defined tasks, standard training/development/test splits and automated evaluation measures (this infrastructure will be fully compatible with the ALiEN toolkit). Particular care will be taken in defining evaluation metrics that do not only measure task success, but also other desirable properties of the emergent code, such as ease of learning for new agents. The Grocery Challenge will be proposed as a shared task at the second Emergent Communication workshop we will sponsor (see A3 of WP6).

Milestones:

ALiEN toolkit v1: computational simulation environment	M36
ALiEN toolkit v2: including analytical tools	M54
Grocery Challenge environment	M54

Work Package 6: Project Overseeing (all project personnel)

Activity 1: *Infrastructure.* At the beginning of the project, assisted by the UPF IT department, I will acquire and test the GPU nodes to be added to the UPF cluster (see *Section c. Resources*). Infrastructure overseeing will continue all through the project, to deal with changing technologies and unforeseen roadblocks.

Activity 2: *Project management.* I will adopt the informal, hands-on, lead-by-example approach that helped me running the Composes ERC project and to supervise a large researcher group as FAIR manager. We will hold weekly project meetings. I will also hold 1:1 meetings with all team members (weekly with PhD students, bi-weekly with post-docs). We will organize reading groups and brainstorming sessions on demand. At the end of each year except the last, we will hold an informal internal 2-day workshop to assess project progress and identify critical issues.

Activity 3: *Dissemination*. A website will serve as the central hub linking publications and software from the project. The main instrument of dissemination will be publications in the proceedings of general ML/AI conferences (ICML, ICLR, NeurIPS, AAAI…), as well as NLP and possibly computer vision events (ACL, NAACL, EMNLP, CVPR, ECCV…). The findings from linguistic analysis of emergent protocols will also be reported in journal articles aimed at the cognitive science community (e.g., in *Cognition* or *Psychological Review*). Towards the end of the project, we will submit a paper with main findings and broader implications to a maximum-impact general science journal. The highly relevant Emergent Communication workshop series associated with the NeurIPS conference is now approaching its 4th edition. ALiEN will host special sessions dedicated to the project, featuring presentations by relevant invited speakers, in two editions of this workshop (one at year 2, the other at year 5). Associating these sessions with the workshop will guarantee a larger audience, and it will reduce overhead, compared to organizing independent events.

Milestones:

Website in place	M2
Computational infrastructure in place	M6
Post-docs start	M6
PhD students start	M12
Annual progress assessment reviews	M12, M24, M36, M48
ALiEN events at Emergent Communication workshop	M18, M54

Timeline

NB: some WP or sub-task names are shortened to fit table column width.

	M6	M12	M18	M24	M30	M36	M42	M48	M54	M60
WP1: Growing DNN Languages										
ES1: input generalization										
ES2: agent generalization										
ES3: putting it all together										
ES4: ALiEN supervision										
WP2: ALiEN for Pre-Trained DNNs										
ES1: one community										
ES2: agent generalization										
ES3: ALiEN supervision										
WP3: The Grocery Challenge										
ES1: simplified setup										
ES2: full-fledged setup										
ES3: agent generalization										
WP4: ALiEN Analysis										
A1: analysis tools										
A2: generalization analysis										
A3: interpretability analysis										
A4: ALiEN vs. humans										
WP5: ALiEN Software										
A1: ALiEN toolkit										
A2: Grocery Challenge environment										
WP6: Project Overseeing										
A1: Infrastructure										
A2: Project management										
A3: Dissemination										

b.2 Research environment

Universitat Pompeu Fabra (UPF), established in 1990, is a public university with strong commitment to research and teaching excellence. It is the first Spanish university in the world Top 200 (THE20) and 15th worldwide among universities under 50 (THE19). It is ranked 2nd in Europe (U-Multirank 2019) and 1st in

Spain for teaching and research (U-Ranking, BBVA Foundation & Ivie, 2019). UPF will give me access to existing cluster infrastructure. It will also provide office space and technical and administrative support. I belong to the **Computational Linguistics and Linguistic Theory** group (COLT; https://www.upf.edu/web/colt). Established in 2017, the group combines quantitative and computational methods to study linguistic problems. It is currently composed of two permanent faculty members, two post-doctoral researchers, and five doctoral students. UPF provides a rich research ecosystem, including NLP groups in the Technology department and groups studying computational evolution and language evolution in the Biology department. **Barcelona** is a lively research hub, with top-level company and academic research labs in AI, computer vision and NLP, many located just a short walk from the UPF campus.

b.3 Risk table

Risk	*Mitigation actions*
WP1, WP2, WP3: Dependencies?	Although the experiments in the three simulation WPs are related and some techniques should ideally be prototyped in WP1 and then applied to WP2 and WP3, there is no crucial dependency such that delays in a WP would prevent concurrent progress in the other WPs.
WP1: Problems generalizing to new referents	i) Work with (still useful) protocols limited to a large but fixed number of object classes. ii) Special training methods to encourage 0-shot generalization: in particular, add many training examples where target and distractors are same-class or extremely similar, to spur emergence of a granular attribute-level language. iii) Study problem at the class level: are there specific classes where fast generalization works better? Does this depend on similarity to training classes? Can we capitalize on this observation, if confirmed?
WP1, WP2, WP3: Problems generalizing to new agents	i) Explore simplified setups, e.g., limit DNN architecture variety. ii) Focus on supervised imitation learning. iii) Study if community-evolved languages have other advantages, even if they are not as fast to transmit as hypothesized.
WP1, WP2: Supervision is not beneficial.	For the time being, we won't get a single "universal" language, but methods to evolve useful languages will still be delivered. Extensive study of *why* supervision does not help: Is it because language drift undoes its benefits? Does supervision hamper generalization?
WP2: Language layer tuning does not suffice to let communication emerge.	Consider both the full-architecture re-training approach (emergent language should still have beneficial properties) and problem simplification (e.g., limit to visual models only, etc.).
WP3: Cannot scale to full Grocery Challenge setup.	Identify problematic aspects and simplify (e.g., simplify reward function).

Section c. Resources: *see separate form.*

References

[1] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 2015.
[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg and L. Fei-Fei. ImageNet Large Scale Visual Recognition challenge. *International Journal of Computer Vision*, 2015.
[3] A. Krizhevsky, I. Sutskever and G. Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 2017.
[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser and I. Polosukhin. Attention is all you need. *NIPS*, 2017.
[5] S. Edunov, M. Ott, M. Auli and D. Grangier. Understanding back-translation at scale. *EMNLP*, 2018.
[6] J. Johnson, B. Hariharan, L. van der Maaten, J. Hoffman, L. Fei-Fei, L. Zitnick and Ross Girshick. Inferring and executing programs for visual reasoning. *ICCV*, 2017.
[7] T. Finin, R. Fritzson, D. McKay and R. McEntire. KQML as an agent communication language. *CIKM*, 1994.
[8] S. Russel and P. Norvig. *Artificial Intelligence: A modern approach, 4th edition*. Pearson, 2020.
[9] D. Bickerton. *More than nature needs: Language, mind, and evolution*. Harvard University Press, 2014.
[10] J. Hurford. *The origins of language*. OUP, 2014.
[11] R. Jackendoff and E. Wittenberg. What you can say without syntax: A hierarchy of grammatical complexity. In *Measuring grammatical complexity*. OUP, 2014.
[12] D. Brentari and S. Goldin-Meadow. Language emergence. *Annual Review of Linguistics*, 2017
[13] L. Bloom. Language development: Form and function. In *Emerging grammars*. MIT Press, 1970.
[14] S. Berk and D. Lillo-Martin. The two-word stage: Motivated by linguistic or cognitive constraints? *Cognitive Psychology*, 2012.
[15] B. Skyrms. *Signals: Evolution, learning, and information*. OUP, 2010.
[16] K. Zuberbühler, D. Cheney and R. Seyfarth. Conceptual semantics in a nonhuman primate. *Journal of Comparative Psychology*, 1999.
[17] S. Steinert-Threlkeld, P. Schlenker and E. Chemla. Referential and general calls in primate semantics. *Lingbuzz preprint*, 2020.
[18] P. Bloom. *How children learn the meanings of words*. MIT Press, 2000.
[19] J. Holm. *An introduction to pidgins and creoles*. CUP, 2000.
[20] Y. Xian, B. Schiele and Z. Akata. Zero-shot learning: The Good, the Bad and the Ugly. *CVPR*, 2017.
[21] P. Bakker, A. Daval-Markussen, M. Parkvall and I. Plag. Creoles are typologically distinct from non-creoles. *Journal of Pidgin and Creole Languages*, 2011.
[22] L. Graesser, K. Cho and D. Kiela. Emergent linguistic phenomena in multi-agent communication games. *EMNLP*, 2019.
[23] F. Li and M. Bowling, M. Ease-of-teaching and language structure from emergent communication. *NeurIPS*, 2019.
[24] O. Tieleman, A. Lazaridou, S. Mourad, C. Blundell and D. Precup. Shaping representations through communication. *arXiv preprint*, 2018.
[25] Y. Ren, S. Guo, S. Havrylov, S. Cohen and S. Kirby. Enhance the compositionality of emergent language by iterated learning. *NeurIPS Emergent Communication Workshop*, 2019.
[26] R. Chaabouni, E. Kharitonov, D. Bouchacourt, E. Dupoux and M. Baroni. Compositionality and generalization in emergent languages. *ACL*, 2020.
[27] R. Lowe, J. Foerster, Y Boureau, J. Pineau and Y. Dauphin. On the pitfalls of measuring emergent communication. *AAMAS*, 2019.
[28] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint*, 2017.
[29] R. Chaabouni, E. Kharitonov, E. Dupoux and M. Baroni. Anti-efficient encoding in emergent communication. *NeurIPS*, 2019.
[30] P. Schlenker, S. Mascarenhas and E. Chemla. Super semantics: A unifying framework for meaning phenomena in nature. *Lingbuzz preprint*, 2020.
[31] E. Kharitonov, R. Chaabouni, D. Bouchacourt and M. Baroni. Entropy minimization in emergent languages. *ICML*, 2020.
[32] D. Silver, J. Schrittwieser, K. Simonyan, ... and D. Hassabis. Mastering the Game of Go without human knowledge. *Nature*, 2017.
[33] A. Graves. *Supervised sequence labelling with recurrent neural networks*. Springer, 2012.

[34] A. Mogadala, M. Kalimuthu and D. Klakow. Trends in integration of vision and language research: A survey of tasks, datasets, and methods. *arXiv preprint*, 2019.
[35] K. He, X. Zhang, S. Ren, J. Sun. Deep residual learning for image recognition. *CVPR*, 2016.
[36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich. Going deeper with convolutions. *CVPR*, 2015.
[37] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.
[38] J.H. Kim, J. Jun and B.T. Zhang. Bilinear attention networks. *NeurIPS*, 2018.
[39] I. Melekhov, J. Ylioinas, J. Kannala and E. Rahtu. Relative camera pose estimation using convolutional neural networks. *ACIVS*, 2017.
[40] J. Devlin, M. Chang, K. Lee and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL*, 2019.
[41] J. Andreas, M. Rohrbach, T. Darrell and D. Klein. Neural module networks. *CVPR*, 2016.
[42] S.-A. Rebuffi, H. Bilen and A. Vedaldi. Learning multiple visual domains with residual adapters. *NIPS*, 2017.
[43] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan and S. Gelly. Parameter-efficient transfer learning for NLP. *ICML*, 2019.
[44] J. Pfeiffer, A. Rücklé, C. Poth, A. Kamath, I. Vulić, S. Ruder, K. Cho and I. Gurevych. AdapterHub: A framework for adapting transformers. *arXiv preprint*, 2020.
[45] Y. Bengio, A. Courville and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.
[46] I. Higgins, D. Amos, D. Pfau, S. Racaniere, L. Matthey, D. Rezende and A. Lerchner. 2018. Towards a definition of disentangled representations. *arXiv preprint*, 2018.
[47] F. Locatello, S. Bauer, M. Lucic, G. Rätsch, S. Gelly, S. Schölkopf, O. Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *ICML*, 2019.
[48] D. Lewis. *Convention*. Harvard University Press, 1969.
[49] J. Batali. Computational simulations of the emergence of grammar. In *Approaches to the evolution of language: Social and cognitive bases*. CUP, 1998.
[50] M. Nowak, and D. Krakauer. The evolution of language. *PNAS*, 1999.
[51] A. Cangelosi and D. Parisi (editors). *Simulating the evolution of language*. Springer, 2002.
[52] M. Christiansen and S. Kirby (editors). *Language evolution*. OUP, 2003.
[53] L. Steels. Evolving grounded communication for robots. *Trends in Cognitive Science*, 2003.
[54] K. Wagner, J. Reggia, J. Uriagereka and G. Wilkinson. Progress in the simulation of emergent communication and language. *Adaptive Behavior*, 2003.
[55] S. Nolfi and M. Mirolli (editors). *Evolution of communication and language in embodied agents*. Springer, 2010.
[56] L. Steels (editor). *Experiments in cultural language evolution*. Benjamins, 2012.
[57] S. Kirby and J. Hurford. The emergence of linguistic structure: An overview of the iterated learning model. In *Simulating the evolution of language*, 2002.
[58] K. Smith and E. Wonnacott. Eliminating unpredictable variation through iterated learning. *Cognition*, 2010.
[59] S. Kirby, T. Griffiths and K. Smith. Iterated learning and the evolution of language. *Current Opinion in Neurobiology*, 2014.
[60] Y. Shoham and K. Leyton-Brown. *Multiagent systems*. CUP, 2009.
[61] L. Panait and S. Luke. 2005. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 2005.
[62] F. Oliehoek and C. Amato. *A concise introduction to decentralized POMDPs*. Springer, 2016.
[63] R. Canaan, J. Togelius, A. Nealen and S. Menzel. Diverse agents for ad-hoc cooperation in Hanabi. *CoG*, 2019.
[64] J. Foerster, I. Assael, N. de Freitas and S. Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *NIPS*, 2016.
[65] E. Jorge, M. Kågebäck and E. Gustavsson. Learning to play Guess Who? and inventing a grounded language as a consequence. *NIPS Deep Reinforcement Learning Workshop*, 2016.
[66] A. Lazaridou, A. Peysakhovich and M. Baroni. Multi-agent cooperation and the emergence of (natural) language. *ICLR*, 2017.
[67] S. Havrylov and I. Titov. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. *NIPS*, 2017.

[68] A. Lazaridou and M. Baroni. Emergent multi-agent communication in the deep learning era. *arXiv preprint*, 2020.
[69] D. Bouchacourt and M. Baroni. How agents see things: On visual representations in an emergent language game. *EMNLP*. 2018.
[70] A. Singh, T. Jain and S. Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *ICLR*, 2019.
[71] A. Das, T. Gervet, J. Romoff, D. Batra, D. Parikh, M. Rabbat and J. Pineau. TarMAC: Targeted multi-agent communication. *ICML*, 2019.
[72] I. Mordatch and P. Abbeel. Emergence of grounded compositional language in multi-agent populations. *AAAI*, 2018.
[73] K. Cao, A. Lazaridou, M. Lanctot, J. Leibo, K. Tuyls and S. Clark. Emergent communication through negotiation. *ICLR*, 2018.
[74] A. Lazaridou, A. Potapenko and O. Tieleman. Multi-agent communication meets natural language: Synergies between functional and structural language learning. *ACL*, 2020.
[75] N. Xie, G. Ras, M. van Gerven and D. Doran. Explainable Deep Learning: A field guide for the uninitiated. *arXiv preprint*, 2020.
[76] J. Andreas. Measuring compositionality in representation learning. *ICLR*, 2019.
[77] J. Andreas, A. Dragan and D. Klein. Translating neuralese. *ACL*, 2017.
[78] D. Bouchacourt and M. Baroni. Miss Tools and Mr Fruit: Emergent communication in agents learning about object affordances. *ACL*, 2019.
[79] T. Linzen, G. Chrupała and A. Alishahi (editors). *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. ACL, 2018.
[80] T. Linzen, G. Chrupała, Y. Belinkov and D. Hupkes (editors). *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. ACL, 2019.
[81] B. Lake and M. Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. *ICML*, 2018.
[82] K. Gulordava, P. Bojanowski, E. Grave, T. Linzen and M. Baroni. Colorless green recurrent networks dream hierarchically. *NAACL*, 2018.
[83] A. Conneau, G. Kruszewski, G. Lample, L. Barrault and M. Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. *ACL*, 2018.
[84] Y. Lakretz, G. Kruszewski, T. Desbordes, D. Hupkes, S. Dehaene and M. Baroni. The emergence of number and syntax units in LSTM language models. *NAACL*, 2019.
[85] M. Baroni. Linguistic generalization and compositionality in modern artificial neural networks. *Trans. Royal Soc. B*, 2020.
[86] T. Linzen and M. Baroni. Syntactic structure from Deep Learning. *Annual Review of Linguistics*, 2020 (in press).
[87] J. Bolhuis, G. Beckers, M. Huybregts, R. Berwick, M. Everaert. Meaningful syntactic structure in songbird vocalizations? *PLoS Biology*, 2018.
[88] S. Townsend, S. Engesser, S. Stoll, S. Zuberbühler and B. Bickel. Compositionality in animals and humans. *PLOS Biology*, 2018.
[89] R. Heesen, C. Hobaiter, R. Ferrer-i-Cancho and S. Semple. Linguistic laws in chimpanzee gestural communication, *Trans. Royal Soc. B*, 2019.
[90] M. McCloskey. Networks and theories: The place of connectionism in cognitive science. *Psychological Science*, 1991.
[91] A. Kell, D. Yamins, E. Shook, S. Norman-Haignere and J. McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 2018.
[92] R. Cichy and D Kaiser. Deep neural networks as scientific models. *Trends in Cognitive Science*, 2019.
[93] C. Manning, K. Clark, J. Hewitt, U. Khandelwal and O. Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. *PNAS*, 2020.
[94] S. Steinert-Threlkeld and J. Szymanik. Ease of learning explains semantic universals. *Cognition*, 2020.
[95] Y. Lakretz, D. Hupkes, A. Vergallito, M. Marelli, M. Baroni and St. Dehaene. Exploring processing of nested dependencies in neural-network language models and humans. *arXiv preprint*, 2020.
[96] R. Chaabouni, E. Kharitonov, A. Lazaric, E. Dupoux and M. Baroni. Word-order biases in deep-agent emergent communication. *ACL*, 2019.
[97] S. Sukhbaatar, A. Szlam and R. Fergus. Learning multiagent communication with backpropagation. *NIPS*, 2016.

[98] W. Huang, I. Mordatch and D. Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. *ICML*, 2020.
[99] J. Cortes, S. Martinez, T. Karatas and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 2004.
[100] T. Lux and M. Marchesi. Scaling and criticality in a stochastic multi-agent model of a financial market. *Nature*, 1999.
[101] E. Kharitonov, R. Chaabouni, D. Bouchacourt and M. Baroni. EGG: A toolkit for research on Emergence of lanGuage in Games. *EMNLP*, 2019.
[102] R. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992.
[103] E. Jang, S. Gu and B. Poole. Categorical reparameterization with Gumbel-Softmax. *ICLR*, 2017.
[104] C. Maddison, A. Mnih and Y.W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. *ICLR*, 2017.
[105] J. Deng, W. Dong, R. Socher, L.-J. Li and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. *CVPR*, 2009.
[106] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. Shamma, M. Bernstein and L. Fei-Fei. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 2017.
[107] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint*, 2019.
[108] T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, L. Zitnick and Piotr Dollar. Microsoft COCO: Common objects in context. *ECCV*, 2014.
[109] D. Elliott, S. Frank, S. Khalil and L. Specia. Multi30K: Multilingual English-German image descriptions. *ACL Workshop on Vision and Language*, 2016.
[110] Wikipedia contributors. Home automation. In *Wikipedia: The Free Encyclopedia*, 2020.
[111] R. Sutton and A. Barto. *Reinforcement Learning: An introduction (2nd ed.)*. MIT Press, 2018.
[112] H. Brighton and S. Kirby. Understanding linguistic evolution by visualizing the emergence of topographic mappings. *Artificial Life*, 2006.
[113] R. Eskander, J. Klavans and S. Muresan. Unsupervised morphological segmentation for low-resource polysynthetic languages. *ACL Workshop on Computational Research in Phonetics, Phonology, and Morphology*, 2019.
[114] J. Goldwater, T. Griffiths and M. Johnson. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 2009.
[115] F. Newmeyer and L. Preston (editors). *Measuring grammatical complexity*. OUP, 2014.
[116] Z. Goldfeld, E. Van Den Berg, K. Greenewald, I. Melnyk, N. Nguyen, B. Kingsbury and Y. Polyanskiy. Estimating information flow in deep neural networks. *ICML*, 2019.
[117] C. Hockett. The origin of speech. *Scientific American*, 1960.
[118] B. de Boer, W. Sandler, S. Kirby. New perspectives on duality of patterning: Introduction to the special issue. *Language and Cognition*, 2014.
[119] E. Gibson, R. Futrell, S. Piantadosi, I. Dautriche, K. Mahowald, L. Bergen and Roger Levy. How efficiency shapes human language. *Trends in Cognitive Science*, 2019.
[120] R. Ferrer i Cancho, A. Hernández-Fernández, D. Lusseau, G. Agoramoorthy, M. Hsu and S. Semple. Compression as a universal principle of animal behavior. *Cognitive Science*, 2013.
[121] N. Zaslavsky, T. Regier, N. Tishby and C. Kemp. Semantic categories of artifacts and animals reflect efficient coding. *CogSci*, 2019.
[122] P. Grice. Logic and conversation. In *Syntax and semantics 3: Speech acts*. Academic Press, 1975.