Transferable Emergent Communication in Multi-Agent Reinforcement Learning

Lawrence Liao, Zhuohan Wang

1 Challenge

Multi-agent systems can greatly benefit from emergent communication, where agents develop their own language-like messages to coordinate and achieve goals. However, a key challenge is that these communication protocols, once learned, are often narrow and task specific. In current research, agents usually learn to communicate from scratch for each new task, leading to slow convergence and little reuse of the learned skills.

We propose to tackle this by enabling the transfer of communication ability between tasks. The novelty lies in combining two components: (1) agents that comprehend and produce messages (a learned ”language”), and (2) a method to transfer this learned communication to new games, improving learning speed and final performance. By establishing that agents who mastered communication in one game can rapidly adapt to another, we address a gap in current MARL research and provide a strong motivation for our project. If successful, this would be novel and worthy of a workshop publication, as it suggests a stepping stone toward more general communicative agents that learn coworking styles usable across tasks.

2 Problem

We focus on the problem of transferable communication in MARL. Specifically: Can agents learn a communication protocol in one environment that accelerates learning and improves cooperation in a new, different environment?

Traditional MARL has shown that allowing messages between agents can lead to emerging languages and improved teamwork. However, these protocols are usually learned in isolation for one task. Our problem statement zeros in on multitask communication learning: after agents fully train in Game A (source task) with communication, we place them (or their communication skills) in Game B (target task) to see if they converge faster or achieve higher win rates than agents without this *prior*. Essentially, we treat communication ability as a skill that can be learned once and **reused**.

The key challenges in this problem include the following.

- **Generalization:** Ensuring the learned messages from Game A are still meaningful or useful in Game B. If the games are too different, naive transfer may fail. We need to define the problem such that some aspects of communication carry over (e.g. similar cooperative elements or strategies).
- **Baseline Comparison:** We will compare against a baseline where the agents in Game B learn from scratch without prior communication experience. Showing a significant improvement over this baseline is critical to demonstrate the value of transferable communication.
- **Efficiency vs. Complexity:** We desire a simple approach and environment for rapid iteration. However, if the task is too simple, even the baseline agents learn quickly, leaving little room for improvement. Therefore, the problem setting likely needs moderate complexity enough that communication helps and transfer provides an edge, but not so complex that experiments are unmanageable.

3 Modeling and Formulation

We consider a standard multi-agent reinforcement learning setup with communication. Formally, we have a set of agents (two or more) interacting in an environment (the game). At each time step, an agent observes its state (or partial observation) and can take two types of actions:

1. **Environment Actions:** Moves that affect the environment, which influence the game's state and reward.

2. **Communication Actions:** Emitting a message to be received by other agents.

We will model messages as discrete tokens or sequences from a vocabulary (potentially emerging as a proto-language). Each agent has an encoder module to encode its internal information into a message, and a decoder module to interpret messages from others into useful internal signals.

The multi-agent setup can be cooperative (agents share a team reward) or competitive. For simplicity and because communication is most naturally beneficial in cooperation, we will focus on a collaborative game setting: all agents work toward a common goal. The joint reward R at each time step (or game outcome) is shared among agents.

We formulate this as a Dec-POMDP (Decentralized Partially Observable Markov Decision Process) with an added communication channel:

- **State s:** the true environment state (not fully observed by any single agent if partial observability).

- **Observations o_i:** what agent i perceives from the state.

- **Action $a_i = (a_i^{env}, a_i^{comm})$:** each agent's combined action, with a_i^{env} the environment action and a_i^{comm} the communication message (which could be None or a token).

- **Transition:** $s_{t+1} = T(s_t, a_1^{env}, \dots, a_N^{env})$ as usual for environment actions. Communication actions don't directly change the world state but are received by agents.

- **Communication reception:** we assume at each step, after all agents emit messages, each agent i receives the set of messages from others (possibly with noise or limited bandwidth).

- **Policy:** Each agent has a policy $\pi_{\theta_i}(a_i^{env}, a_i^{comm} \mid o_i, m_{-i})$ where m_{-i} denotes messages received from other agents (at the previous step or concurrently). θ_i includes parameters for both deciding on environment action and what message to send. We might share policy parameters across agents or not, depending on whether agents are homogeneous.

- **Learning objective:** maximize expected cumulative reward $E[\sum_t R_t]$ for the team.

The policy parameters will be learned via reinforcement learning (e.g., policy gradient or Q-learning adapted to multi-agent setting).

To incorporate LLM-based prior in this formulation, we augment the model: the encoder/decoder for communication can be partly initialized or guided by a Large Language Model. For example, the encoder might map an agent's observation to a message embedding using a transformer pre-trained on language, so that messages lie in a space of meaningful semantics. The decoder might use an LLM to interpret incoming messages.

Another approach: use the LLM as an action advisor - at early training stages, the LLM (with a prompt describing the situation) suggests what message an agent should send, providing a supervised signal to mimic. This effectively jump-starts the communication with some meaningful structure before pure RL fine-tuning takes over. Formally, we could add an auxiliary loss L_{LM} at training time that measures divergence between the agent's message distribution and an LLM-suggested distribution (teacher forcing the communication to be closer to natural language instructions that an outside observer might give).

We will consider two games (environments) in our formulation:

- **Game A (Source task):** A game with two (or more) agents that requires communication to achieve high performance. For instance, a cooperative navigation game where agents must rendezvous or a strategy game where each agent has partial information and must share clues to solve a puzzle.

- **Game B (Target task):** A different game where communication is also beneficial. It should share some common structure with A (so that communication skills transfer) but not be identical. For example, if Game A is a cooperative navigation in one map, Game B could be a similar navigation in a new map with new obstacles, or a slightly different objective requiring teamwork. Alternatively, A and B could be two different levels of a game (like two scenarios in Overcooked or two different collaborative Atari games).

We formally define transfer as follows: Let θ^A be the learned parameters (policies including encoders/decoders) after training on Game A. For Game B, instead of starting with random initialization θ_0^B, we initialize (part or all of) the agent parameters with θ^A. Particularly, the communication-related parts (message encoder/decoder) will be reused. We then train on Game B to get $\theta^B_{finetuned}$.

We measure:

- **Convergence rate:** e.g., number of episodes to reach a certain performance threshold in Game B.
- **Final win rate / return:** the asymptotic performance after training.

The hypothesis in our formulation is $\theta^B_{finetuned}$ (with transfer from A) will outperform a baseline $\theta^B_{scratch}$ trained from scratch, in terms of both learning speed and final performance.

In summary, our modeling sets up a MARL with communication framework, and explicitly defines a transfer learning scenario across two tasks within that framework. This provides a testbed to evaluate if an emergent communication protocol (encoded in θ^A) can be considered a form of knowledge that generalizes to a new task.

4 Proposed Approach

Our approach can be divided into two phases: learning in the source game (with possible LLM guidance), and transfer & adaptation in the target game. We emphasize simplicity in design, using relatively straightforward architectures and small-scale games initially, while acknowledging that a sufficiently complex task is needed to showcase improvements.

4.1 Learning Communication in Game A

We will first train agents on Game A until they reach mastery (near-optimal performance). To facilitate rapid learning of a useful communication protocol, we incorporate an LLM-based prior as a form of guided learning:

- **LLM as a Prior:** At the start of training, we initialize the agents' communication policy using an LLM. Concretely, we might use a pre-trained transformer (like GPT-style model) as part of the encoder/decoder. For example, the encoder could convert an agent's observation into a natural language description (just as an auxiliary output), which the LLM has been pretrained to do for generic situations. While the agents won't literally output English sentences to each other (to keep the message space small), the internal weights from the LLM could bias them toward "meaningful" representations.
- **Prompting:** Another way is prompting: during early episodes, we can query an LLM with the full state (as an oracle) to suggest what each agent should communicate ("Agent1 should tell Agent2 to go left"). These suggestions train a preliminary communication policy via supervised loss.
- **Reinforcement Learning:** With or without LLM initialization, the agents then train with RL on Game A. We can use policy gradient methods (like Multi-Agent PPO or COMA) that handle multi-agent credit assignment. The reward signal comes from the game outcomes, but we also might add a small penalty for overly long messages or an information bottleneck to encourage concise, relevant communication. Over time, agents refine both their action policy and their messaging strategy. We expect to see emergent co-working styles e.g., one agent consistently taking a leader role and instructing the other, or specialized codes for frequent actions indicating the agents have developed a communication protocol tailored to Game A.

By the end of Phase 1, we will have agents that are experts in Game A, with an established communication pattern. We will analyze this protocol to understand it. While interpretability is a bonus, our main goal is to see that it improves performance in Game A compared to agents that couldn't communicate.

4.2 Transfer to Game B

In Phase 2, we take the trained agents (or specifically their learned parameters) and deploy them in Game B. We consider a few variations:

- **Direct Fine-Tuning:** Use the entire policy from Game A as the starting policy for Game B and continue training on B. This way, both the action policy and communication policy have a head-start.
- **Frozen Communication, New Action Policy:** To isolate the effect of communication transfer, we could freeze the communication-related parameters (encoder/decoder) learned from A, and only reinitialize and train the action decision layers for Game B. This tests if the language from Game A is directly useful in Game B without further modification.
- **Fine-Tune Communication:** Alternatively, allow the communication protocol to evolve slightly in Game B through fine-tuning, which might be necessary if B has some new concepts. The hope is the fine-tuning will be faster than learning from scratch, since it's adjusting an existing protocol.

During training on Game B, we will measure the performance periodically. We expect that agents with the transferred comm protocol (from A) learn B faster than a baseline where communication is enabled but initialized randomly. Ideally, they might also achieve a higher final reward because they don't get stuck re-discovering basic coordination tricks.

4.3 Evaluation Plan

To claim success, we will evaluate:

- **Learning Curve Comparison:** Plot cumulative reward (or win rate) vs. training episodes for: (a) baseline MARL on Game B from scratch, (b) our transferred communication approach. A steeper learning curve for (b) would indicate positive transfer.
- **Final Performance:** Compare the asymptotic performance of (a) vs (b). It's important the transfer doesn't just learn faster but also at least reaches equal or higher performance.
- **Ablations:** We will test variations to ensure the improvements come from the communication transfer. For example, what if we transfer the policy but not the communication channel (agents start with no comm in B)? What if we transfer only partially? Also, comparing with transferring without LLM guidance in A will show if the LLM prior provided any extra boost (maybe the LLM makes the protocol more general, as hypothesized).

Complexity and Cost We plan to keep individual experiments relatively lightweight. In early development, we'll likely use a simplified grid-world game (e.g., a treasure hunt where two agents must meet at a point with each seeing different clues). This allows quick iterations to debug the training and transfer process. Once stable, we will move to a more complex domain to truly test the method's advantage for instance, a multi-agent Atari game or a more complex cooperation environment (possibly using the Melting Pot or PettingZoo MARL benchmark tasks). We anticipate that in a trivial game, even random initialization might learn quickly, so the benefit of transfer won't be obvious. A slightly complex game (with partial observability and requiring coordination) will create a scenario where learned communication gives a measurable leg up.

Emergent Behaviors We will also qualitatively examine if the nature of communication in Game B for the transferred agents differs from scratch. Perhaps they carry over some ”words” or signals from Game A. We might observe emergent co-working styles for example, if in Game A the agents established a leader-follower dynamic through communication, do they quickly re-establish a similar dynamic in Game B, whereas scratch agents take longer to figure out roles?

5 Conclusion

We propose a project to investigate emergence and transfer of communication among cooperative agents across tasks. The project addresses a novel challenge ensuring that agents not only learn to communicate, but also retain and reuse that ability in new contexts, much like humans do. We will model the problem in a MARL framework with encoders/decoders for messages and utilize an LLM for initial guidance. The approach will be implemented in increasingly complex games, starting simple for rapid iteration and scaling up to demonstrate clear benefits over baselines.

If successful, this work would be one of the first to show that emergent multi-agent communication can be a transferable skill, marking it as a noteworthy contribution suitable for an academic workshop. We are mindful of balancing simplicity (for practicality) and complexity (for meaningful results), and our plan reflects a path to achieve both. The end result will be a comprehensive project report and possibly publishable findings that highlight the power of combining LLM priors with MARL communication and the potential of agents developing general co-working styles that span tasks.

References

[1] Abhishek Das, Théo Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Michael Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In *International Conference on Machine Learning*, 2019.

[2] Amir Eftekhar, Kuan-Hui Zeng, Jiajun Duan, Ali Farhadi, Aniruddha Kembhavi, and Ranjay Krishna. Selective visual representations improve convergence and generalization for embodied ai. *arXiv preprint arXiv:2311.04193*, 2023.

[3] Jakob N Foerster, Yannis M Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, 2016.

[4] Hengyuan Hu, Adam Lerer, Alexander Peysakhovich, and Jakob Foerster. Other-play for zero-shot coordination. In *International Conference on Machine Learning*, 2020.

[5] Jiechao Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *Advances in Neural Information Processing Systems*, 2018.

[6] Samuel Karten, Margo Tucker, Hao Li, Shashank Kailas, Michael Lewis, and Katia Sycara. Interpretable learned emergent communication for human–agent teams. *IEEE Transactions on Cognitive and Developmental Systems*, 2023. Early access.

[7] Jason Lee, Kyunghyun Cho, Jason Weston, and Douwe Kiela. Emergent translation in multi-agent communication. In *International Conference on Learning Representations Workshop Track*, 2017.

[8] Ido Levy, Paul Woudenberg, Douwe Kiela, and Angeliki Lazaridou. Unsupervised translation of emergent communication. In *AAAI Conference on Artificial Intelligence*, 2025.

[9] Hao Li, Hadi N. Mahjoub, Bardia Chalaki, Vamshi Tadiparthi, Kyungjae Lee, Ehsan Moradi-Pari, Corey M. Lewis, and Katia Sycara. Language grounded multi-agent reinforcement learning with human-interpretable communication. In *NeurIPS*, 2024.

[10] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *AAAI Conference on Artificial Intelligence*, 2018.

[11] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, 2016.

[12] Andrew Szot, Max Schwarzer, Harsh Agrawal, Bogdan Mazoure, Ryan Metcalf, William Talbott, and Alexander Toshev. Large language models as generalizable policies for embodied tasks. In *The Twelfth International Conference on Learning Representations*, 2024.