Pragmatic Communication: Bridging Neural Networks for Distributed Agents

Tianhao Guo†‡
†Department of Physical and Electronic Engineering, Shanxi University, Taiyuan, China
‡National Key Laboratory of Integrated Services Networks (ISN), Xidian University, Xi'an, China
E-mails: tianhao_guo@sxu.edu.cn

IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) | 978-1-6654-9427-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/INFOCOMWKSHPS57453.2023.10226074

***Abstract*—In this paper, an intelligence-to-intelligence communication design with a language generation scheme is studied. The concepts and features of pragmatics and pragmatic communication are first discussed and defined from a linguistic point of view: intelligence-to-intelligence communication in a certain environment, using task performance as the evaluation criterion, with the inputs of the goal and the construction of the environment, and the output of task completion. Then, we propose the "glue neural layer" (GNL) design to bridge two intelligence to form a deeper neural network for effective and efficient communication training. Based on the design of GNL, we shed light on the thoughts about the relationship between the structure of languages and neural networks. Furthermore, a neuromorphic framework of pragmatic communication is proposed to find a base for further discussion. Experiments show that GNL design can dramatically change performance. Finally, the advantage of pragmatic and several open research problems are discussed.**

I. Introduction

In device-to-device communication and resource offloading, the mobility of devices causes short timescale variability. To make the most of every communication chance in opportunistic fog networking, compression for information exchange is necessary. Thanks to the development of deep neural networks, the end-to-end deep learning enabled semantic communication (DeepSC) system was proposed [1] to compress the messages by transmitting "high-level" information (from passing symbols without considering the meaning to "reaching meaning"). DeepSC splits different layers of neural networks on both sides of devices and the messages are replaced by forwarding parameters.

The source-channel coding and encoding are replaced by different layers and are jointly optimized by end-to-end backpropagation. Moreover, semantic communication uses a variety of metrics, such as sentence similarity of natural language information and distortion of picture information, and only requires the matching of semantic meaning, not the error-free matching of bit sequences like traditional communication systems, which relaxes the error requirements. In this way, DeepSC extracts and compresses bit information with higher-level semantic information, providing redundancy when sending the same bits. Along with joint source-channel coding (JSCC), semantic communication is anticipated to theoretically surpass the Shannon limit.

However, Shannon and Weaver [2], [3] also envisioned the future direction by dividing communication into three levels: 1, passing symbols (bits); 2, "reaching meaning"; 3, communication having an effect on behavior and thus "reaching effectiveness ". In this paper, we propose pragmatic communication, focusing on the "third-level" information (from "reaching meaning" to "reaching effectiveness").

To reach effectiveness, pragmatic communication must return to the ultimate goal: influence the communication subjects and make a difference in actions, thus eventually increasing performance. At the receiver side of DeepSC, adding one or more layers of neural networks to choose actions depending on the semantics is a simple and intuitive approach to put pragmatic communication into practice. This approach is equivalent to classification and it is restricted to simple action selection, which is no different with semantic communication.

In order to achieve complex and sequential actions to reach the goal after communication, the action subject should be part of the communication system. Naturally, the communication system should also include the communication subject. In this paper, **the action subject and communication subject are regarded as one agent** for both communication and execution.

A. From Semantic Communication to Pragmatic Communication

In the existing semantic communication research, the source information remains the external input of the communication system, which is generated by the communication subject. If the communication subject is introduced into the communication system and placed at both the sending and receiving ends, the first problem arises is **how the communication subject generates the source information (language generation)?** The second problem is: **since communication is not an action that can interact with the environment to get loss function from the reward, how to train the communication ability of agents?**

- To answer the question of language generation between machines, we have to return to the original goal of communication: collaboration under survival pressure. In this paper, we choose a partially observed environment with multiple fully cooperated (with a teamwork reward) agents [4] and enable agents to send messages.
- To answer the question of training to communicate, the communication between agents is not simulated as the action since there is no way to train. In this paper, we propose a glue neural layer (GNL) to link two distributed agents into a deeper neural network for forwarding (the contents of the messages) and backpropagation in the training phase and turn the activation function into step functions in the execution phase for compression of the message (lightness of pragmatic communication) and simulation of the "communication action" [4]. Another question arises: **how to design the activation function to transfer to step function without loss of functionality smoothly?**

TABLE I: List of Terminology in Pragmatic Communication.

Concepts	Meanings
M2M language learning	agents are trained to generate source information for the pragmatic communication system
Bandwidth($\|m\|$)	The size of the message (bits) transmitted between agents
Glue neural layer (GNL)	The layer that bridges two agents so that forward and back propagation are possible
Channel width	The dimension of the glue neural layer
Word	The one-hot vector.
Dictionary (codebook)	A message with the size of $\|m\|$ can express $2^{\|m\|}$ words. $2^{\|m\|}$ is the dictionary size
Sentence	Dictionary can be reduced to $2^{\|m\|/N}$, while sentence length increase to N (parallel one-hot vectors)
Semantic base (state)	the meaning of the word (one-hot vector) understood by machines

- To answer the question of activation function design for GNL, the Logistic function with a random value with Gaussian distribution is used [4] so that the weights of the GNL will be close to 0 or 1 at the end of the training phase, therefore it can smoothly transfer to execution (inference) phase. Now GNL, as a channel for communication between two agents, is built to bridge two split neural networks of distributed devices, but the channel width is still narrow. We further propose a question, **how to design the GNL to build a better channel for boosting the training of pragmatic communication?**
- To answer the question of increasing the channel width for machine-to-machine language generation, we change the Logistic function to the Softmax function with Gumbel distribution. The input of the Softmax function is with no limits (message size $|m| > 1$ bit), and the output of the Softmax function is a one-hot vector with higher dimension (from $|m|$ to $2^{|m|}$). During the backpropagation of the training phase, without increasing the message size $|m|$, it is easier to train the communication function of the agent through the sparsed GNL, as easy as the classification task. Same to human language acquisition and generation, it is easier to use and learn languages by words, not by popping out letters. However, when $|m|$ gets larger, the machine will face the curse of dimensionality. Just like natural language created by humans, it is hard to communicate if the language has a very large dictionary.
- To solve the curse of dimensionality to enable pragmatic communication in light-weight devices in fog networks, we propose the concept of the sentence for machine language generation. Similar to the natural language process (NLP), it is desirable to resemble a word with a one-hot vector and a sentence with several such vectors, which means parallel one-hot vector channels. The dimension decrease from $2^{|m|}$ to $2^{|m|/N}$. It is well known that most of the popular natural languages have almost the same sentence size (no more than one hundred words) and dictionary size (no more than one million words). The reason is obvious that the neural networks in our brains have limited dimension and depth, which leads to limited long-term memory and short-term memory.

B. Linguistic Proofs of Pragmatic Communication

According to Morris's triadic theory of signs [5], there are three relations: the relationship between the sign and its referent, the relationship between the sign and the person, and the relationship between the sign and another sign, belonging to semantics, pragmatics, and syntax, respectively.

According to Grice's theoretical framework [6], semantics studies "what is said," and pragmatics studies "what is implicated." It is widely believed that the semantic meaning of a sentence, i.e., "what is said," is completely **context-free** , while "what is implicated," is **dependent on the context and changes with time**. Once semantics cannot fully explain the meaning of a sign, then the meaning has to be attributed to pragmatics. In some cases, "what is implicated" is more important because it is the ultimate goal of communication. But semantics is more general.

Different from the semantics paradigm, where languages are all universal natural languages, according to pragmatics, language can be derived from rules only defined by both sides of the conversation, without taking into consideration external factors. Moreover, the more tacit agreement there is, the fewer words are needed. As a result, trained pragmatic information should be light and hard to understand by outsiders.

Both semantics and pragmatics study meanings. Semantics deals with meaning in a binary relationship, i.e., "what" and "X" in "What does X mean?"; while pragmatics deals with meaning in a ternary relation, i.e., "what," "you," and "X" in "What did **you** mean by X?" The pragmatics definition of meaning is relative to the language users, whereas the semantics definition of meaning is a property of a linguistic expression.

Pragmatics also studies the mixture and transition of language and actions. For example, Bob want Alice to turn off the air conditioning. In addition to talking, Bob can also shiver or sneeze. That is, the combination of language and action (behavior), together with training and learning.

C. The Features and Definition of Pragmatic Communication

Correspondingly, the features of pragmatic communication can be concluded as follows:

- Pragmatic communication is based on a certain context or environment, not as general as semantic communication for any pictures or texts.
- Pragmatic involves communication subjects so the source information is not an outside input (but the goal of communication or the task) and the output is also measured by the performance of the subjects.
- Pragmatic communication is based on tacit agreements (or language generation) between machine-to-machine only for certain tasks, so it is more confidential and light-weighted.
- Pragmatic communication is not purely transmitting messages, it is based on machines cooperating for a common goal of a task. Therefore, communication is trained with actions together in a certain environment with a certain task.

The concept of semantics is widely used in computer vision and natural language processing, and research in semantic

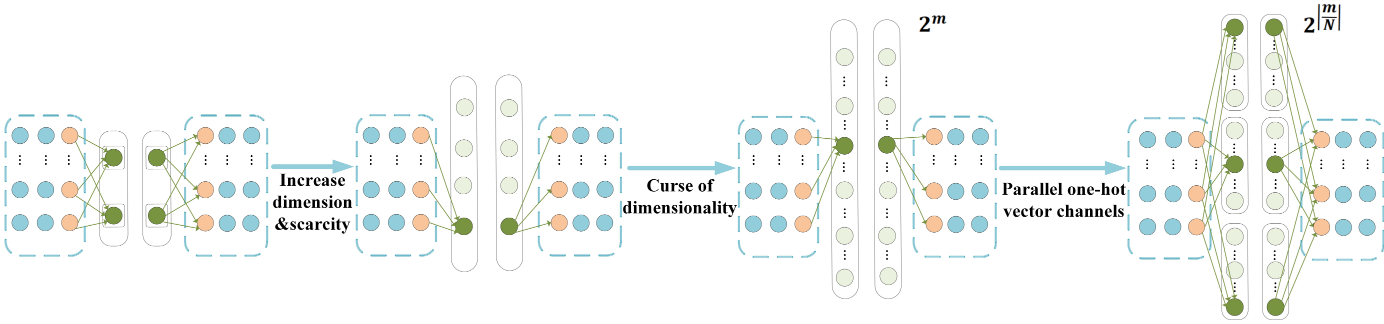

Fig. 1: The design of glue neural layer from increasing the dimension to one-hot vector, to meeting the curse of dimensionality, and finally to parallel one-hot vector channels.

communication systems also focuses on cases where the inputs (source information) are texts and videos. However, pragmatics, which is not well known by the academia in engineering and computer science, specializes more in the generation, understanding, and use of languages in a certain situation for a certain task. Therefore we have the conclusion that pragmatic communication is suited for applications in reinforcement learning scenarios where distributed agents communicate with each other.

Above all, the pragmatic communication is defined as intelligence-to-intelligence communication in a certain environment based on tacit agreements (or language generation) for certain tasks, with the inputs of the goal and the construction of the environment, and the output of task completion, using performance of actions as the evaluation criterion.

II. The Framework of Pragmatic Communication

A. The Glue Neural Layer Design

In this paper, we call the layer between two agents (the sender and the receiver) the glue neural layer (GNL) since it serves as an intermediary between both agents for forwarding and gradient backpropagation, bonding neural networks of two agents to a single deeper one. Logistic function with Gaussian distribution reparameterization is first used to transfer to step function without loss of functionality smoothly. However, the Logistic function is a single-input single-output activation function, restricting its suitability for pragmatic communication with a large channel bandwidth (message size). Inspired by the features of natural languages, we propose the GNL design for effective and efficient pragmatic communication.

B. Increase the Dimension and Sparsity of the Glue Neural Layer

To increase the channel of pragmatic communication, the Softmax function is used as the activation function of the GNL. However, the output of a Softmax function is a probability vector (exaggerating the difference, good for classification but bad for intermediate results) instead of a one-hot vector reflecting the true probabilistic situation required by the latter part of the NN. We adopt a well-known reparameterization trick called Gumbel Softmax. Instead of directly sampling from the distributed output by the Softmax function, we can add in advance a random vector independently sampled from a standard Gumbel distribution. For the Softmax function, the study in [7] proves the random re-parameter selection of the Gumbel distribution is the best choice. The probability vector output by the Softmax function shows the original meaning of probability and can accurately pass the parameters to the receiver agent. Jang et al. [7] also show that as training proceeds, the combination of the Softmax function with the temperature parameter can approximate the Argmax function, thus ensuring a smooth transition to the execution phase.

In the training phase, for incorrect initial values, the gradient simply reduces the value of that neural node. Therefore, the one-hot vector generated by the Softmax function reduces the correlation of different symbols because of sparsity and uses all possible semantics bases to directly participate in the training of neural networks across both transceivers and transmitters. As a result, the use of the Softmax function increases the error correction capability of cross-neural network training by expanding the dimension of GNL, thus speeding up the training, improving the accuracy of communication, and making the message easier to be understood by the agent with the same message size (the amount of information remains the same). The left part of Fig. 1 shows the transition from Logistic-based GNL to Softmax-based GNL with message size $|m| = 2$.

The message of multiple bits corresponds to a single semantic state (semantics bases) to bridge two agents. Although the message itself has the same amount of information, it is easier to be received by the human brain with a word as the unit rather than a letter, and the same is true for machine language design.

C. The Equilibrium Point of the Vocabulary and the Sentence Length for Machine Language

Assuming that the message size required for a particular communication scenario is large, the dimension of the GNL can be much larger than the neural networks of the distributed hardware, as shown in Fig. 1. In addition to the lightness, when the dimension of the output layer of the sender agent is too large, **the full traversal is not feasible, and it is difficult to train sufficiently** for stable performance.

It is difficult to learn a language if the vocabulary is too large, so we speak sentences rather than popping out a single word for communication. Inspired by natural languages, we propose parallel one-hot vector channels for GNL design, as shown in the right part of Fig. 1. Similar to NLP, it is desirable to resemble a word with a one-hot vector and a sentence with several such vectors. The message is divided by N (sentence length), regarded as the source encoding of pragmatic communication since there is no need to compress the message. The channel width is decreased from $2^{|m|}$ to $N * 2^{|m/N|}$.

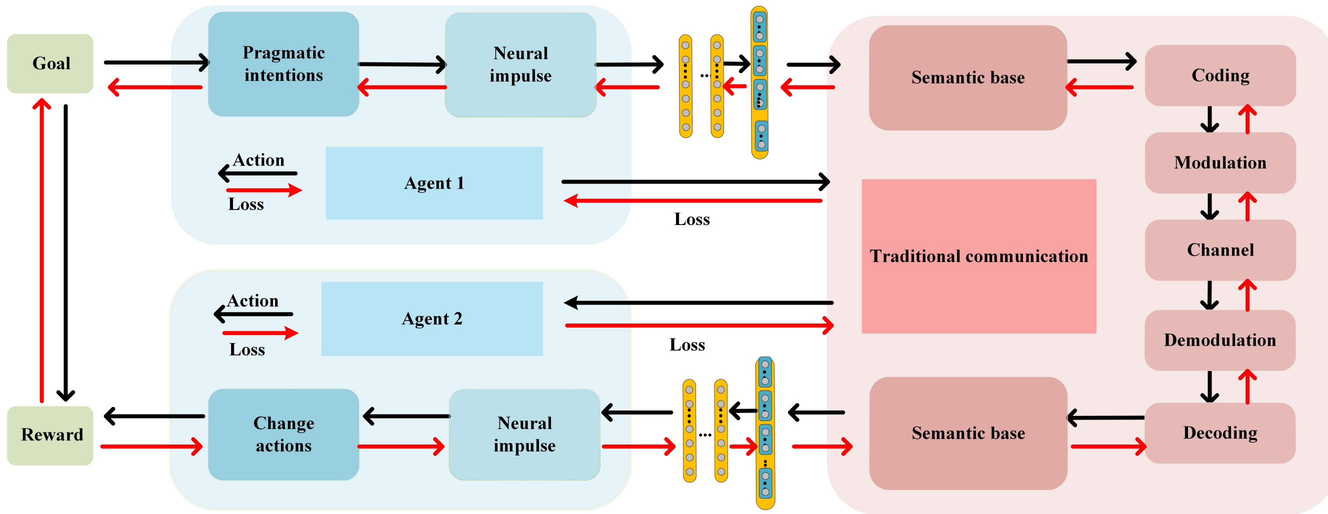

Fig. 2: Framework of pragmatic communication.

D. The Framework of Pragmatic Communication

The procedure of pragmatic communication is shown in Fig. 2.

- In the initial stage, the neural networks of the sender generate random neural impulses, which are mapped to semantic states (semantic bases) in the GNL.
- Then the semantic states are encoded into a message of $|m+r|$ bits (m for information bits, r is redundant bits used in channel encoding for error detection, correction, and confidentiality) and sent to the channel after modulation.
- After demodulation and decoding, the semantic bases are linked to the input layer of the receiver agent (the input layer can also include other information such as observations).
- Influenced by the message, the input layer generates neural impulses, which are forwarded through the neural networks to the output layer of the receiver agent, leading to the action selection.
- The environment gives a reward in response to the actions. The loss function is generated from the reward and is propagated backward from the receiver all the way to the sender (red arrow in Fig. 2) so that the sender and receiver gradually correct the semantic intention and comprehension respectively.

As we can see from the framework, pragmatic communication is a plug-in functionality which is trained by both sides of the communication agents with the help of GNL. The channel coding and modulation can be the same with the traditional communication, and can also be the same with split learning-based semantic communication, split neural networks in both side to train channel coding. The pragmatic communication system has not yet considered channel noise as [8] did because channel encoding and decoding are transparent in this paper.

III. Evaluation

In the game of prisoners and one interrogation room [4], [9], the dimension and the activation function of the GNL are changed to study the performance of pragmatic communication and influence of GNL.

The Fig. 3 shows the GNL with the activation function of Logistic with Gaussian distribution. When the message size $|m|$ increases from 1 bit to 2 bits, the speed of convergence gets slower but the rewards can eventually reach 1. When $|m|$ increases from 2 bits to 3 and 4 bits, the speed of convergence gets even slower at the beginning, but rewards can reach 1 earlier. The results show that with the increase of the dimension of GNL, it takes more time for training, but the communication bandwidth between agents has a direct influence on the ability of pragmatic communication, thus influencing the performance of the cooperation task.

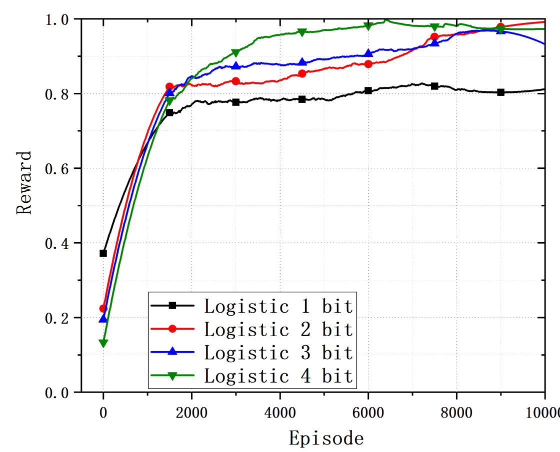

Fig. 3: Influence of message size for four agents.

As shown in Fig. 4, for the GNL with the activation function of Softmax with and without Gumbel distribution, the message size $|m|$ is fixed as 2 bits but the dimension of the GNL is the same as the Logistic case with $|m| = 4$. However, because the scarcity of the GNL brought by a one-hot vector that can boost training, the speed of convergence is much faster. The curve of Softmax with Gumbel distribution is the first one to reach 1, followed by the curve of Softmax without Gumbel distribution.

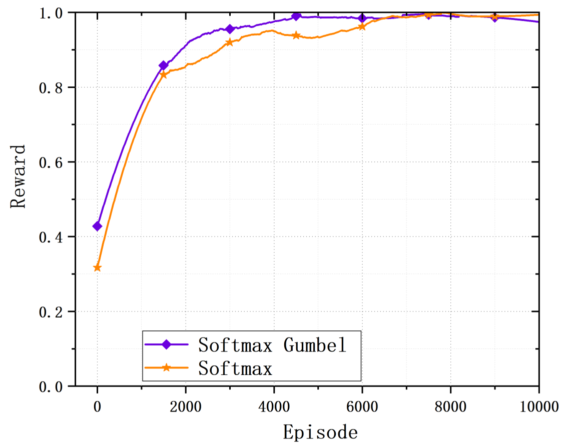

Fig. 4: Influence of Softmax-Gumbel for four agents.

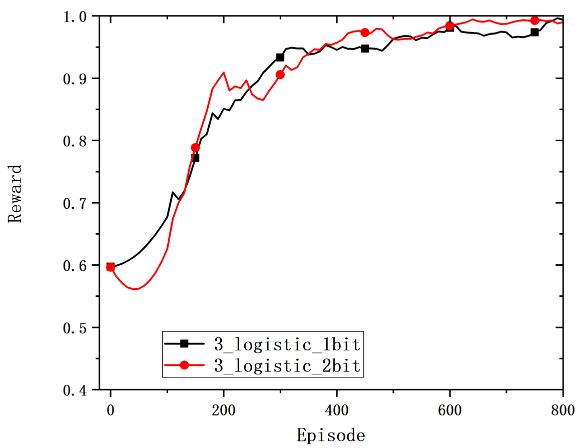

Fig. 5: Influence of message size for three agents.

In this game, the complexity of collaboration is directly related to the number of prisoners. If there are 4 prisoners, the reward with 1-bit communication cannot reach 100% (70%) but 2-bit can significantly reduce the training time and the task success rate can reach 100% and is stable. The message with 2 bits is sufficient to describe all four situations (semantic bases) of the game (00, 01, 10, and 11) for GNL with the Logistic function or (0001, 0010, 0100, 1000) for GNL with Softmax function, which also reveals the lightness of pragmatic communication.

Moreover, the pragmatic communication can be further compressed. In the scenario with three prisoners, as shown in Fig. 5, the collaboration with a message size of 1 bit of communication can also achieve a 100% task success rate [4], but needs more training time (slower than 2 bit case). The reason is that pragmatic communication is trained including agents with the memory.

IV. Discussion

In this section, two advantages of pragmatic communication are discussed. Based on that, we list several promising applications.

A. *Lightness of Pragmatic Communication*

The lightness of pragmatic communication refers to the following three aspects.

- lightness of communication. Compared with traditional communication and semantic communication, the pragmatic message is more condensed because of tacit agreements and memory of machines. Therefore, pragmatic communication is suitable for distributed scenarios with limited bandwidth and poor channel conditions, especially military scenarios.
- Lightness of devices. The reduced dimensional design of GNL is suitable for distributed devices with limited dimensionality and depth of neural networks.
- Lightness of the training. The purpose of GNL design is to reduce the training overhead and improve the expression and comprehension ability of the agents.

B. *Confidentiality of Pragmatic Communication*

From the perspective of confidential communication, the eavesdropper agent cannot participate in the training and can only train the neural networks by eavesdropping on the message with the binary-type data, which cannot be derived to train its networks. Even if the transmission is continuous floating-point data that can be derived, understanding the floating-point message requires: first, eavesdropper needs to guess the dimension, depth, and structure of the neural networks of our agents and build an eavesdropping agent with the same neural network, the same behavior space, and the same observation of the environment; second, eavesdropper needs to correspond the message to $2^{|m|}$ different states (semantic bases); third, the glue layer with $2^{|m|}$ dimensions should be trained with fully connected weights with the input layer. The difficulty is similar to learning a language from scratch and completing the "Turing test" in a machine language environment, and in the process, it requires long periods of eavesdropping, observation, and behavior without being detected by us. We can also add redundancy to our messages before they are sent by encoding a state into a bit message, which can be used for channel encoding to improve the success rate of sending messages in a harsh wireless environment and to further enhance security so that eavesdroppers cannot correspond a bit message to a semantic base (state).

C. *Applications of Pragmatic Communication*

Pragmatic communication has the advantages of lightness and confidentiality and the disadvantage of context dependence. Therefore, pragmatic communication is useful in distributed system with security requirements but limited hardware resources such as bandwidth, computation, energy, and storage.

For example, the GNL design may be useful for light split machine learning or federated learning. Moreover, pragmatic communication is promising in IoT, mobile edge computing, and robots. Especially, pragmatic communication is useful in cases with extremely narrow Shannon channel bandwidth, extremely high security requirements, and extreme insufficiency of resources, for example, military communication and disaster relief.

V. Future Directions and Conclusion

In summary, we propose a framework of pragmatic communication with GNL design inspired by the usage of nature languages. Then we find proofs that “pragmatic” is suitable to describe intelligence-to-intelligence communication from linguistic point of view. Moreover, the features of pragmatic communication perfectly corresponds the features of “pragmatics”. Preliminary experiments validate the effectiveness design of GNL.

However, the equilibrium point of scarcity and lightness of GNL have not been found. Here we conclude some works for future study.

- A mathematical way to find the equilibrium point of the vocabulary and the sentence length (scarcity and lightness) of GNL is meaningful because it sheds light on the relationship between languages and neural networks. Otherwise, extensive experiments are needed to realize the equilibrium point just for agents with certain structures of neural networks.
- More environments are needed to test pragmatic communication and more algorithms such as proximal policy optimization (PPO) should be added to both sides of agents to increase the stability of the pragmatic communication, so that to find the equilibrium point.
- The framework in this paper is not perfect, but it is the base for future discussion. For example, it is worth trying to replace the Softmax-Gumbel combination with a constellation diagram with Rice or Rayleigh distribution for the activation function of the GNL, which means that GNL might also be implemented at the modulation layer with the consideration of the noise as [8] did.
- The agents communicate with each other in this work have the same neural networks. However, communication is also necessary and possible between subjects with different neural structures. For example, dogs can also communicate with humans, and dogs can even understand simple words. However, the “channel” of human-to-dog communication is not as wide as the “channel” of human-to-human communication, and might not be as wide as the “channel” of dog-to-dog communication. Inspired by this phenomenon, extensive experiments are needed for the investigation on pragmatic communication between agents with different dimensions and depths of neural networks. Moreover, agents can also have different roles in the coordination task. For example, in the one hundred prisoners with one light bulb game, agents with different roles perform better [9].
- Pragmatic communication is goal-oriented or task-oriented communication, so a literature review and a clear classification of tasks are necessary. Since loss is the foundation of backpropagation in training and the loss function is based on reward, how to assign rewards or credits to communication is important in pragmatic communication. For games such as referential games [8] or the correctness of bidding in the bridge game [10], the task is all about communication and the content of the message is the belief or the guess, so the reward can be regarded as direct ACK for pragmatic communication and communication itself can be modeled as an action. However, for many tasks, communication is not the ultimate goal and the ultimate goal of communication is to influence the action subjects for cooperation improvement. If communication itself cannot directly get the reward from the environment, it is hard to assign contributions from the actions that really interact with the environment and the GNL design in this paper might be the solution.

References

[1] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, “Deep learning enabled semantic communication systems,” *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.

[2] C. E. Shannon, *The mathematical theory of communication*. University of illinois Press Champaign, IL, USA, 1949.

[3] W. Weaver, “Recent contributions to the mathematical theory of communication,” *ETC: a review of general semantics*, pp. 261–281, 1953.

[4] J. Foerster, I. A. Assael, N. De Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” *Advances in neural information processing systems*, vol. 29, 2016.

[5] C. W. Morris, *Foundations of the Theory of Signs*. University of Chicago, 1938.

[6] P. Grice, *Studies in the Way of Words*. Harvard University Press, 1989.

[7] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with gumbel-softmax,” *arXiv preprint arXiv:1611.01144*, 2016.

[8] T. Tung, S. Kobus, J. P. Roig, and D. Gündüz, “Effective communications: A joint learning and communication framework for multi-agent reinforcement learning over noisy channels,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2590–2603, 2021.

[9] H. van Ditmarsch and B. Kooi, *One Hundred Prisoners and a Light Bulb*. Springer International Publishing, 2015, pp. 83–94.

[10] Z. Tian, S. Zou, I. Davies, T. Warr, L. Wu, H. B. Ammar, and J. Wang, “Learning to communicate implicitly by actions,” in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, no. 05, 2020, pp. 7261–7268.