Research Statement

He He

The past several years have seen a growing need for intelligent systems that work in real time and interact with humans, such as self-driving cars and personal assistants. Most existing learning methods are passive: we feed in the data and request a prediction. These methods are often not sufficient for interactive systems due to their assumption of a fixed data distribution and the lack of a communication channel. My research builds upon the insight that a model is always situated in a dynamic environment and that information acquisition is an integral part of learning. Two themes of my work to date are: **prediction with adaptive information** and **interacting with humans**. I have developed models that make dynamic predictions by deciding when and what information is needed for a particular instance, as well as agents that collaborate with humans and communicate in natural language.

More broadly, my goal is to build intelligent agents that work with people in a dynamic environment. Such interactive applications require us to rethink the learning paradigm. My research approach emphasizes on integrating predictions with decisions about exchanging information with the environment and the users. To develop effective interactive learning paradigms, one must combine innovations from machine learning, natural language processing, and human-computer interaction. As a result, my work spans from principled learning algorithms [1, 2] to live interactive platforms [3, 4].

Dynamic Information Acquisition for Efficient Prediction

Information acquisition is a crucial part for any intelligent systems. While it might be possible to gather all available information, efficiency becomes an issue because information usually comes at some expense, ranging from computational cost such as processing large text files to monetary cost such as conducting a medical exam. As machine learning models are increasingly applied to real-time, data-intensive applications, there is a need for dynamic models that choose useful information from various sources on its own. Gathering all information is often neither possible nor necessary. For example, when doctors diagnose diseases, they start with basic data (e.g. age, blood pressure, body temperature); only when these observations are not sufficient to make a decision do they move on to costly tests such as MRI or X-ray. The goal is a good trade-off between information cost and prediction accuracy. I formalized this problem as a sequential decision-making process: for each instance, the model uses a *policy* to query features sequentially based on past predictions until it is confident enough to make the final prediction. Thus it learns which features are most cost-effective on an example-by-example basis. I applied this framework to both multiclass classification [1] and structured prediction (dependency parsing [5]). The latter is one of the first few works that study feature computation cost in natural language processing, and our parser gains up to 8x speedup without hurting the accuracy.

At an abstract level, we learn a policy to search a space—subset of features in the above setting. The initial success in feature selection motivated me to think about algorithms that learn to search the solution space of combinatorial optimization problems. Can we exploit common problem structures across instances to learn promising search paths? Can we predict the future cost of a node without expanding it? I designed an algorithm that learns to prune and prioritize nodes on a branch-and-bound tree for solving Integer Linear Programming (ILP) problems [2]. Given the same amount of budget (time or number of nodes explored), our solver consistently finds better solutions compared to a highly-optimized commercial solver (Gurobi) across multiple problems.

My approach to the above problems is based on imitation learning: at training time, a teacher with knowledge of the ground truth demonstrates the correct behavior, so that the learner can learn to mimic the teacher's action in various situations. The key challenge here is that the teacher's action selection policy may not be within the learner's capability. To address this issue, I proposed an adaptive learning objective that interpolates between the teacher's demonstrated action and the learner's action to provide a more accessible learning curriculum [1]. Our algorithm enjoys the same theoretical guarantees (the learner can reach the teacher's performance in the end) and better empirical performance compared to directly learning from the teacher.

Incremental Language Understanding

The dynamic information acquisition setting I described above assumes that all information is available *upon request*, which may not be true for real-time applications. What if the needed information has not arrived yet? In a dynamic environment, the model must anticipate what will happen in the future and react to changes in a timely manner. Moving toward this setting, I started to explore scenarios where the information needed for a prediction comes in a stream. My particular focus in the language domain is simultaneous interpretation—a challenging task even for human translators. Unlike translating written text, the interpreter must translate *while* the speaker is talking under stringent time constraints. The key challenge in this setting is that the critical information needed for a translation may not be available until some time in far future, making it hard to reduce translation delay without significantly hurting the translation quality. For example, when translating from verb-final languages (e.g. Japanese) to verb-medial languages (e.g. English), a naïve translator often has to wait until the end of a sentence for the verb to produce a sensible translation. In [6], we find that compared to translators, professional simultaneous interpreters use a unique set of skills to balance speed and accuracy, e.g. break long sentences into smaller, independent units, summarizing content, reordering words according to the source language grammar. Inspired by these observations, I developed the first framework [7] that unifies machine translation and interpretation techniques in a reinforcement learning setting, enabling later work on end-to-end neural interpretation systems [8, 9]. In addition to sentence segmentation which is widely used in prior work, predicting future words [7] and reordering words [10] produces better translations sooner. Our work also opens the door to collaborative interpretation: The machine provides intermediate results such as segmented sentences and predicted future words, and the human interpreter organizes the pieces together. This would significantly reduce the cognitive load of an interpreter and potentially lower the required expertise for this task.

Interacting with Humans

Learning in dynamic environments often involves interacting with humans. There is an increasing trend of systems communicating with users to better understand their needs, e.g. virtual assistants and smart homes. A natural language communication interface will open up endless possibilities for interacting with humans: Imagine scientists collaborate with computers to navigate through the literature on a research problem, or children chat with the computer to learn new things. My work [4] designed a new symmetric collaborative dialogue setting, where two agents, each with private knowledge, must communicate to achieve a common goal. Consider two people chatting about their friends to find the mutual one, or more generally, exchanging ideas to solve a puzzle. A typical dialogue in this setting consists of information exchange, question answering, coordination, and inference. I developed a neural network model that bridges two main approaches to dialogue: semantic parsing that grounds utterances to logical forms and encoder-decoder models that embed utterances to a vector space. Our model combines structured knowledge and unstructured conversation context by connecting utterance embeddings to a knowledge graph. Our dialogue agent can successfully complete the task with humans and is considered collaborative and humanlike by their human partners.

Communication is easier if we know who we are talking to. Humans tailor their communication strategy a lot depending on the other person's intention and habits. To tackle this problem, I focused on an online game setting,[1] where players compete with each other to answer trivia questions as fast as possible. The goal is to design an agent that can play with humans and adapt to different playing strategies (e.g. cautious, aggressive). I proposed a deep reinforcement learning model which learns a latent distribution of player strategies as it plays with multiple users repeatedly [11]. The model then adjusts its strategy depending on what type of player it is competing against. I have shown that by explicitly modeling different user behavior, the agent is able to recognize different types of strategy and exploit weak players.

[1] `http://protobowl.com`

Future Work

My prior work has laid the foundation for prediction with adaptive information and interacting with humans. Moving toward the future, I plan to address new challenges arising from the interactive setting, such as robust interaction interface and goal-oriented natural language communication.

My ongoing work focuses on **robust dialogue systems**. One common failure mode of current data-driven dialogue models is that once the conversation deviates from the training data (i.e. the model no longer fully understands the dialogue context), the model falls into some disastrous state (e.g. repeating itself, generating generic responses). Such brittleness prevents us from deploying dialogue models, which thus prevents learning through user interaction. Currently, I am exploring two key ideas to tackle this problem. First, we often have strong prior knowledge about the user intention given a dialogue domain (e.g. booking tickets, chatting about movies), which allows us to map utterances to structured slots and values with high precision. Second, the dialogue agent should know what it does not know, and avoid the unknown regions during the conversation by asking questions or diverting the topic. With these guidelines, our hybrid rule and data-driven system has achieved better performance on multiple tasks compared to more complex neural models. Looking forward, the end goal of this project is to build dialogue agents that improve themselves through chatting with people, much as modern search engines learn from user click feedback. This requires significant advancement in language generation, reinforcement learning, and human-in-the-loop learning, e.g. how to refine an utterance toward a certain goal, how to do safe exploration and solicit feedback in a conversation, and how to interleave with human responses to keep the conversation on the right track.

Modern intelligent agents need to interact with a population of users who have different preferences and habits. Expanding my work on opponent modeling [11], I am interested in developing **personalized, adaptive systems** that automatically adjust to different individuals and occasions. One example is *text style transfer*. People change their vocabulary and sentence structure depending on whom they are addressing and where they are speaking, e.g. talking to their child vs talking to their boss, tweeting vs writing an email. I am currently working on extracting idiosyncratic patterns from text and building models that apply these patterns to another piece of text. When interacting with a population of users to learn strategies targeting at each individual, another challenge is to balance exploration and exploitation across multiple subjects. We do not want to repeatedly discover common knowledge about a community of users. Imagine a tutoring system working with many learners: While each learner may prefer different amount of time on each topic, most of them would follow a common syllabus. To scale an interactive system to a large number of users with diverse needs, I will explore several research questions: How can we discover a shared "basis" of user behavior? How can we transfer knowledge among individuals and quickly adapt to new users or new scenarios? How to capture changing user behavior when they are also adapting to the system?

The success of an interactive system hinges on the quality of the feedback it receives. Asking for human demonstrations at each step (supervised learning) is too expensive. Fortunately, reinforcement learning provides a way to guide the model through numerical rewards. In many real-world applications, however, the reward is a single number (e.g. user satisfaction score) and is insufficient to point the model in the right direction. On the other hand, much richer information is carried in language, which is easy to obtain during interaction. To address the problem of sparse reward, I am interested in **eliciting and learning from rich supervision signals in language**. When humans communicate with each other, a lot of feedback is implicit in their responses. For example, short, generic responses can indicate losing interest, such as "okay", "fair enough", and "I guess so". In more goal-oriented settings, they may also directly express negative feedback, e.g. "I don't want that.", "No, I was asking...". To take advantage of these signals, we need an empathetic component in the system that "senses" where things go wrong, thus alleviate the credit assignment problem. Furthermore, the agent can proactively ask users or a human teacher for stronger supervision. This can involve demonstration of a task, instruction in natural language, or paraphrasing a request. In this human-in-the-loop setting, we must also balance learning and the cost of rich supervision, which connects to my previous work on dynamic information acquisition.

Overall, I am interested in developing robust, efficient interactive agents, with natural language as the key communication channel. By innovating in ways an agent gathers and exchanges information in a dynamic environment, I hope to open up possibilities for what an intelligent agent can work with and learn from.

References

[1] He He, Hal Daumé III, and Jason Eisner. Imitation learning by coaching. In *Neural Information Processing Systems (NIPS)*, 2012.

[2] He He, Hal Daumé III, and Jason Eisner. Learning to search in branch and bound algorithms. In *Neural Information Processing Systems (NIPS)*, 2014.

[3] Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daumé III. Besting the quiz master: Crowdsourcing incremental classification games. In *Empirical Methods in Natural Language Processing*, 2012.

[4] He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In *Association for Computational Linguistics (ACL)*, 2017.

[5] He He, Hal Daumé III, and Jason Eisner. Dynamic feature selection for dependency parsing. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2013.

[6] He He, Jordan Boyd-Graber, and Hal Daumé III. Interpretese vs. translationese: The uniqueness of human strategies in simultaneous interpretation. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2016.

[7] Alvin C. Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. Don't until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[8] Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O. K. Li. Learning to translate in real-time with neural machine translation. In *EACL*, 2017.

[9] Khanh Nguyen, Hal Daumé, and Jordan L. Boyd-Graber. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *EMNLP*, 2017.

[10] He He, Alvin Grissom II, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. Syntax-based rewriting for simultaneous machine translation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*, 2015.

[11] He He, Jordan Boyd-Graber, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *Proceedings of the International Conference of Machine Learning*, 2016.