A Survey of Reinforcement Learning Informed by Natural Language

Jelena Luketina[1*], **Nantas Nardelli**[1,2], **Gregory Farquhar**[1,2], **Jakob Foerster**[2], **Jacob Andreas**[3], **Edward Grefenstette**[2,4], **Shimon Whiteson**[1], **Tim Rocktäschel**[2,4*]

[1]University of Oxford, [2]Facebook AI Research

[3]Massachusetts Institute of Technology, [4]University College London

{jelena.luketina, gregory.farquhar, shimon.whiteson}@cs.ox.ac.uk,

{nantas, jnf, egrefen, rockt}@fb.com, jda@mit.edu

Abstract

To be successful in real-world tasks, Reinforcement Learning (RL) needs to exploit the compositional, relational, and hierarchical structure of the world, and learn to transfer it to new environments. Recent advances in representation learning for language make it possible to build models that acquire world knowledge from text corpora and integrate this knowledge into downstream decision making problems. We thus argue that the time is right to investigate a tight integration of natural language understanding into RL in particular. We survey the state of the field, including work on instruction following, text games, and learning from textual domain knowledge. Finally, we call for the development of new environments as well as more investigation into the potential uses of recent Natural Language Processing (NLP) methods for such tasks.

1 Introduction

Languages, whether natural or formal, allow us to encode abstractions, to generalize, to communicate plans, intentions, and requirements, both to other parties *and to ourselves* [Gopnik and Meltzoff, 1987]. These are fundamentally desirable capabilities of artificial agents. However, agents trained with traditional approaches within dominant paradigms such as Reinforcement Learning (RL) and Imitation Learning (IL) typically lack such capabilities, and struggle to efficiently learn from interactions with rich and diverse environments. In this paper, we argue that the time has come for natural language to become a first-class citizen of solutions to sequential decision making problems (*i.e.* those often approached with RL[1]). We survey recent work and tools that are beginning to make this shift possible, and outline next research steps.

Humans are able to learn quickly in new environments due to a rich set of commonsense priors about the world, some of which are reflected in natural language [Shusterman *et al.*, 2011]. It is thus natural to question whether agents can learn not only from rewards or demonstrations, but also from language, to improve generalization and sample efficiency in sequential decision making problems. While some environments require agents to process language by design, for instance if language is part of the observation space [Branavan *et al.*, 2009], a growing body of work suggests that language in RL has broader applications. In this survey, we are particularly interested in sequential decision making problems with the following properties: (a) the use of RL methods is severely constrained by data efficiency due to limited or expensive environment interactions, and (b) human priors that would help to solve the task are or can be expressed easily in natural language.

Information that could be useful for RL tasks is contained in both general and task-specific large textual corpora. For instance, consider pre-training neural representations of words and sentences from large general textual corpora. Such pre-trained representations have shown great success in transferring syntactic and, to some extent, semantic information to downstream tasks [Peters *et al.*, 2018b; Goldberg, 2019]. Cross-domain transfer from self-supervision on language data might similarly help to initialize RL agents. In addition, applying machine reading techniques [Banko *et al.*, 2007] to task-specific corpora like game manuals and wikis has the promise to inform agents of valuable starting policies [Branavan *et al.*, 2012] or task-specific environmental dynamics and reward structure [Narasimhan *et al.*, 2018].

Previous attempts at using language for RL tasks have mostly been limited to relatively small [Janner *et al.*, 2018] or synthetic language corpora [Hermann *et al.*, 2017]. We argue that recent significant advances in pre-training and representation learning [Peters *et al.*, 2018a; Radford *et al.*, 2019] make it worth revisiting this research agenda with a much more ambitious scope for the integration of natural language and RL. While the problem of grounding language (*i.e.* learning the correspondence between text and environment features) remains a significant research challenge, past work has shown that high-quality linguistic representations can improve other types of cross-modal transfer (*e.g.* zero-shot transfer in image classification [Frome *et al.*, 2013]) and might be similarly applied in RL.

This survey focuses on the current state of the field of integrating natural language into RL agents and environments. We first provide background on RL and techniques for self-

*Contact Authors

[1]We write RL for brevity and to focus on a general case, but our arguments are relevant for many sequential decision making approaches, including IL and planning.

supervision and transfer in natural language (§2). We then review prior work, considering settings where interaction with language is necessary (§3.1) and where language can optionally be used to facilitate learning (§3.2). In the former category we review instruction following, induction of reward from language, and environments with text in the action or observation space, all of which have language in the problem formulation itself. In the latter, we review work that has used language to facilitate RL by transfer from domain-specific textual resources, or as a means of representing policies.

We conclude by identifying what we believe are the most important challenges for integrating natural language in RL (§4). Inspired by gaps in the existing literature, we advocate for the development of new research environments utilizing domain knowledge in natural language, as well as a wider use of NLP methods like pre-trained language models and parsers to inform RL agents about the structure of the world.

2 Background

2.1 Reinforcement and Imitation Learning

Reinforcement Learning [Sutton and Barto, 2018] is a framework that enables agents to reason about sequential decision making as an optimization process. Problems are formulated as Markov Decision Processes (MDPs), tuples $\langle S, A, T, R, \gamma \rangle$ where S is the set of states, A the set of actions, T the transition function $T : S \times A \rightarrow S$, R the reward function $T : S \times A \times S \rightarrow \Re$, and $\gamma \in [0, 1)$ is a discount factor, typically set by either the environment or the agent designer. Given this setup, the goal of the optimization process is to find a policy $\pi(a|s) = p(A = a|S = s)$ that maximises the expected discounted cumulative return $\sum_{k=0}^{\infty} \gamma^k r_{k+1}$. Partially observable MDPs (POMDPs) are used to model settings in which the agent cannot access the Markov state S, and must instead rely only on noisy or ambiguous observations.

Since their inception, RL algorithms have been successful in applications such as continuous control [White and Sofge, 1992], dialogue systems [Singh *et al.*, 2002], and board games [Tesauro, 1995]. Recent improvements in function approximation and pattern recognition made possible by deep learning have allowed RL to scale to problems with high dimensional input spaces such as videogames [Torrado *et al.*, 2018] and complex planning problems such as Go [Silver *et al.*, 2017]. For a review on such recent algorithmic developments see [Arulkumaran *et al.*, 2017]. State-of-the-art methods are often sample inefficient, requiring millions or billions of interactions, and often generalize poorly to tasks even marginally different to those seen during training.

Imitation learning typically employs the same formalism as RL but no rewards are observed. Instead, the learning algorithm has access to demonstrations of optimal or near-optimal policies. Supervised learning methods can then be used to find an approximately optimal policy. IL can be combined with RL by providing a favorable policy initialisation or an auxiliary objective.

2.2 Transfer from Natural Language

NLP has seen a recent surge of models that transfer syntactic and semantic knowledge to various downstream tasks. Current NLP systems commonly employ deep learning models and represent (sequences of) words using dense vector representations that are either pre-trained from large textual corpora or that are learned from scratch for the task at hand [Deerwester *et al.*, 1990; Mikolov *et al.*, 2013; Peters *et al.*, 2018a]. These models are motivated by Firth's distributional hypothesis: "You shall know a word by the company it keeps" [Firth, 1957]. Hence, the learned vector representation of a word like *scorpion* should be similar to *spider* if the corresponding words appear in similar contexts, *e.g.*, if they can both be found around other words like *venomous* or *exoskeleton*. Such word representations can transfer knowledge to downstream language processing tasks [Peters *et al.*, 2018b] (*e.g.* in text classification problems, the knowledge that documents containing the word *scorpion* are likely about animals even if only *spider* appears in the training data).

It is thus natural to ask whether knowledge about the world, communicated in the form of language, might also prove useful for other decision making problems. This communication may take the form of explicit goals (*go to the door on the far side of the room*), constraints on policies (*avoid the scorpion*), or generic information about the reward or transition function (*scorpions are fast*). In the following section, we survey existing literature on using natural language to aid goal specification, sample efficiency and generalization in RL.

3 Current Use of Natural Language in RL

In reviewing efforts that integrate language in RL we highlight work that develops tools, approaches, or insights that we believe may be particularly valuable for improving the generalization or sample efficiency of learning agents through their use of natural language. We separate the literature into **language-conditional** RL (in which interaction with language is necessitated by the problem formulation itself) and **language-assisted** RL (in which language is used to facilitate learning). The two categories are not exclusive, in that for some language-conditional RL tasks, NLP methods or additional textual corpora are used to assist learning [Bahdanau *et al.*, 2019; Goyal *et al.*, 2019].

3.1 Language-conditional RL

We first review literature for tasks in which integrating natural language is unavoidable, *i.e.*, when the task itself is to interpret and execute instruction given in natural language, or natural language is part of the state and action space. Approaches to such tasks, we argue in (§4.1), can also be improved by developing methods that enable transfer from general and task-specific textual corpora. Methods developed for language-conditional tasks are relevant for language-assisted RL as they both deal with the problem of grounding natural language sentences in the context of RL. Moreover, in cases such as instruction following of sequences, the full instructions are often not necessary to solve the underlying RL problem but they assist learning by structuring the policy [Andreas *et al.*, 2017] or by providing auxiliary rewards [Goyal *et al.*, 2019].

Instruction Following

Instruction following agents are presented with tasks defined by high-level (sequences of) instructions. We focus on in-

structions that are represented by (at least somewhat natural) language, and may take the form of formal specifications of appropriate/inappropriate actions, of goal states (or goals in general), or of desired policies. Effective instruction following agents execute the low level actions corresponding to the optimal policy or reach the goal specified by their instructions, and can generalize to unseen instructions during testing.

In a typical instruction following problem, the agent is given a description of the goal state or of a preferred policy as a proxy for a description of the task [MacMahon *et al.*, 2006]. Some work in this area focuses on simple object manipulation tasks [Wang *et al.*, 2016; Bahdanau *et al.*, 2019], while other work focuses on 2D or 3D navigation tasks where the goal is to reach a specific entity. Entities might be described by predicates ("Go to the red hat") [Hermann *et al.*, 2017] or in relation to other entities ("Reach the cell above the westernmost rock.") [Janner *et al.*, 2018]. Some approaches use object-level representation and relational modeling to exploit the structure of the instruction in relation to world entities [Chen and Mooney, 2011], or embed both the instruction and observation to condition the policy directly [Mei *et al.*, 2016]. The use of human-generated instructions, as apposed to synthetic language, is not a standard in the field [Hermann *et al.*, 2017], but natural language instructions are for example used in [MacMahon *et al.*, 2006; Misra *et al.*, 2017].

This line of work has strong ties to Hierarchical RL [Barto and Mahadevan, 2003], with individual sentences or clauses from instructions corresponding to subtasks [Branavan *et al.*, 2010]. When the vocabulary of instructions is sufficiently simple, an explicit options policy can be constructed that associates each task description with its own modular sub-policy [Andreas *et al.*, 2017]. A more flexible approach is to use a single policy that conditions on the currently executed instruction, allowing some generalization to unseen instructions [Mei *et al.*, 2016; Oh *et al.*, 2017]. Current approaches of this form, however, require first pre-training the policy to interpret each of the primitives in a single-sentence instruction following setting.

Rewards from Instructions

Another use of instructions is to induce a reward function for RL agents or planners to optimize. This is relevant when the environment reward is not available to the agent at test time, but is either given during training [Tellex *et al.*, 2011] or can be inferred from expert trajectories. The work addressing this setting is influenced by methods from the inverse reinforcement learning (IRL) literature [Ziebart *et al.*, 2008; Ho and Ermon, 2016]. A common architecture consists of a reward-learning module that learns to ground an instruction to a (sub-)goal state or trajectory segment, and is used to generate a reward for a policy-learning module or planner.

When the transition function is known and full demonstrations are available, the reward function can be learned using standard IRL methods like MaxEnt IRL [Ziebart *et al.*, 2008] as in [Fu *et al.*, 2019]. Otherwise, given a dataset of goal-instruction pairs, as in [Bahdanau *et al.*, 2019], the reward function is learned through an adversarial process similar to that of [Ho and Ermon, 2016]. For a given instruction, the reward-learning module aims to discriminate goal states from the states visited by the policy (assumed non-goal), while the agent is rewarded for visiting states the discriminator cannot distinguish from the goal states.

When environment rewards are available but are very sparse, instructions may still be used to generate auxiliary rewards to help learn efficiently. In this setting, [Goyal *et al.*, 2019] and [Wang *et al.*, 2019] use auxiliary reward-learning modules trained offline to predict whether trajectory segments correspond to natural language annotations of expert trajectories. [Agarwal *et al.*, 2019] perform a meta-optimisation to learn auxiliary rewards conditioned on features extracted from instructions: the auxiliary rewards are learned so as to increase performance on the true objective after being used for a policy update. As some environment rewards are available, these settings are closer to language-assisted RL.

Language in the Observation and Action Space

Environments that use natural language as a first-class citizen for driving the interaction with the agent present a strong challenge for RL algorithms. Natural language heavily exploits ambiguity and common bias to cheaply encode information. Furthermore, linguistic observation and action spaces grow combinatorially as the size of the vocabulary and the complexity of the grammar increase, and grow compositionally when interaction relies on multiple sentences, paragraphs, or longer text. Examples of settings that include these problems are *dialogue systems*, *question answering* (Q&A), and *text games*. The former two are largely areas of historical focus in NLP research, and they have been extensively reviewed by [Chen *et al.*, 2017] and [Bouziane *et al.*, 2015] respectively. Recently, however, work has emerged around visual and embodied Q&A, in which agents are tasked with performing multimodal visual and language-based reasoning [Antol *et al.*, 2015; Johnson *et al.*, 2017], or act in a 3D environment while answering queries as part of the task [Das *et al.*, 2017; Chen *et al.*, 2018]. These new lines of research attempt to introduce more elements of decision making with respect to both tasks and algorithms.

Text games are easily framed as RL environments and make a good testbed for structure learning, knowledge extraction, and transfer across tasks [Branavan *et al.*, 2012]. [Narasimhan *et al.*, 2015] also observe that when the action space of the text game is constrained to verb-object pairs, decomposing the Q-function as separate parts for verb and object provides enough structure to make learning more tractable, although they don't propose a way to scale this method to action-sentences of arbitrary length. To facilitate the development of a consistent set of benchmarks in this problem space, [Côté *et al.*, 2018] propose *TextWorld*, a framework that allows the generation of instances of text games that behave as RL environments. They note that existing work on word-level embedding models for text games (e.g. [Kostka *et al.*, 2017]) only achieve acceptable performance on easy tasks.

3.2 Language-assisted RL

In this section, we consider work that explores how knowledge about the structure of the world can be transferred from natural language corpora and methods into RL tasks, in cases where

language itself is not essential to the task. Textual information can assist learning by specifying informative features, annotating states or entities in the environment, or describing shared subtasks in a multitask setting. In most cases covered here, the textual information is task-specific, with a few cases of using task-independent information through language parsers [Branavan *et al.*, 2012] and sentence embeddings [Goyal *et al.*, 2019].

Language for Communicating Domain Knowledge

In a more general setting than instruction following, any kind of text containing potentially task-relevant information could be available. Such text may contain advice regarding the policy an agent should follow or information about the environment dynamics. Such unstructured and descriptive (in contrast to instructive) textual information is more abundant and can be found in wikis, manuals, books, or the web. However, using such information requires (i) retrieving useful information for a given context and (ii) grounding that information with respect to observations.

[Eisenstein *et al.*, 2009] learn abstractions in the form of conjunctions of predicate-argument structures that can reconstruct sentences and syntax in task-relevant documents using a generative model. These abstractions are used to obtain a feature space that improves imitation learning outcomes. [Branavan *et al.*, 2012] develop an agent that learns a policy for the first few moves in Civilization II, a turn-based strategy game, while accessing the game's natural language manual. The agent is trained using Monte Carlo rollouts to simultaneously estimate Q-values, select relevant sentences from the manual, and classify whether words in the manual relate to the state, action, or neither. This task is simplified by hand-engineered features that help match states and actions to relevant words and sentences. More recently, [Narasimhan *et al.*, 2018] investigate planning in a 2D game environment where properties of entities in the environment are annotated by natural language (*e.g.* the 'spider' and 'scorpion' entities might be annotated with the descriptions "randomly moving enemy" and "an enemy who chases you", respectively). Descriptive annotations facilitate transfer by learning a mapping between the annotations and the transition dynamics of the environment.

Language for Structuring Policies

One use of natural language is communicating information about the state and/or dynamics of an environment. As such, it is a great candidate for building priors on the structure or representations used by an agents' models. This can include shaping representations towards more generalizable abstractions, making the representation space more interpretable to humans, or structuring the computations within a model.

[Andreas *et al.*, 2016] propose a neural architecture that is dynamically composed of a collection of jointly-trained neural modules, based on the parse tree of a natural language prompt. While originally developed for visual question answering, [Das *et al.*, 2018] and [Bahdanau *et al.*, 2019] successfully apply variants of this idea to RL tasks. [Andreas *et al.*, 2018] explores the idea of natural language descriptions as a policy parametrization in a 2D navigation task adapted from [Janner *et al.*, 2018]. In a pre-training phase, the agent learns to imitate expert trajectories conditional on instructions. The agent is then adapted to the new task, in which instructions and expert trajectories are not available, by optimizing in the space of natural language instructions.

Compositionality and hierarchical structure of natural language make it a particularly good candidate for representing policies in hierarchical RL. [Shu *et al.*, 2018] and [Andreas *et al.*, 2018] can be viewed as using language (rather than logical or learned representations) as policy specifications for a hierarchical agent.

4 Future of Natural Language in RL

The preceding sections surveyed the literature exploring how natural language can be integrated with RL. Several trends are evident: (i) studies for language-conditional RL are more numerous than for language-assisted RL, (ii) learning from task-dependent text is more common than learning from task-independent text, (iii) within work studying transfer from task-dependent text, only a handful of papers study how to use unstructured and descriptive text, (iv) there are only a few papers exploring methods for structuring internal plans and building compositional representations using the structure of language, and finally (v) natural language, as opposed to synthetic languages, is still not the standard in research on instruction following.

To advance the field, we argue that more research effort should be spent on learning from naturally occurring text corpora in contrast to instruction following. While learning from unstructured and descriptive text is particularly difficult, it has a much greater application range and potential for impact. Moreover, we argue for development of more diverse environments with real-world semantics. The tasks used so far use small and synthetic language corpora, and are too artificial to significantly benefit from transfer from real-world textual corpora. In addition, we emphasize the importance of developing standardized environments and evaluations for comparing and measuring progress of models that integrate natural language into RL agents.

We believe that there are several factors that make focusing such efforts worthwhile now: (i) recent progress in pre-training language models, (ii) general advances in representation learning, as well as (iii) development of tools that make construction of environments for RL agents easier. Some significant work, especially in language-assisted RL, has been done prior to the surge of deep learning methods [Eisenstein *et al.*, 2009; Branavan *et al.*, 2012], and is worth revisiting. In addition, we encourage the reuse of software infrastructure, e.g. [Chevalier-Boisvert *et al.*, 2018; Bahdanau *et al.*, 2019; Côté *et al.*, 2018] for constructing environments and standardized tests.

4.1 Learning from Text Corpora in the Wild

The web contains abundant textual resources that provide instructions and how-to's.[2] For many games, detailed walkthroughs and strategy guides exist. We believe that transfer from task-independent corpora could also enable agents to

[2] *e.g.* https://www.wikihow.com/ or https://stackexchange.com/

better utilize such task-dependent corpora. Preliminary results that demonstrate zero-shot capabilities [Radford *et al.*, 2019] suggest that a relatively small dataset of instructions or descriptions could suffice to ground and consequently utilize task-dependent information for better sample efficiency and generalization of RL agents.

Task-independent Corpora

Natural language mirrors how humans think and communicate about the world. For instance, a state-of-the-art pre-trained language model would assign a higher probability to "get the green apple from the tree behind the house" than to "get the green tree from the apple behind the house". Harnessing such implicit commonsense knowledge captured by statistical language models could allow us to transfer such knowledge to RL agents.

In the short-term, we anticipate more use of pre-trained word and sentence representations for research on language-conditional RL. For example, consider instruction following with natural language annotations. Without transfer from a language model (or another language grounding task as in [Yu *et al.*, 2018]), instruction following systems cannot generalize to instructions containing unseen synonyms or paraphrases (e.g. "fetch a stick", "return with a stick", "grab a stick and come back"). While pre-trained word and sentence representations alone will not solve the problem of grounding an unseen object or action, they do help with generalization to instructions with similar meaning but unseen words and phrases. In addition, we believe that learning representations for transferring knowledge about analogies, going beyond using analogies as an auxiliary tasks [Oh *et al.*, 2017] will play an import role in generalizing to unseen intsructions.

As pre-trained language models and automated question answering become more capable, one interesting long-term direction are studies on agents that can query knowledge more explicitly. For example, during the process of planning in natural language, an agent that has a pre-trained language model as sub-component could let the latter complete "to open the door, I need to..." with "turn the handle". Such an approach could be expected to learn more rapidly than *tabula rasa* reinforcement learning. However, such agents would need to be capable of reasoning and planning in natural language, which is a related line of work (see Language for Structuring Policies in §3.2).

Task-dependent Corpora

Research on transfer from descriptive task-dependent corpora is promising due to its wide application potential. It also requires development of new environments, as early research may require access to relatively structured and partially grounded forms of descriptive language similarly to [Narasimhan *et al.*, 2018]. One avenue for early research are environments with relatively complex but still synthetic languages, providing information about environmental dynamics or advice about good strategies. For example, in works studying transfer from descriptive task-dependent language corpora [Janner *et al.*, 2018; Narasimhan *et al.*, 2018], natural language sentences could be embedded using representations from pre-trained language models. Integration of pre-trained machine reading systems with RL agents trained to query them could help in extracting useful information from unstructured task-specific language corpora such as the game manual used in [Branavan *et al.*, 2012].

4.2 Towards Diverse Environments with Real-World Semantics

One of the central promises of language in RL is the ability to rapidly specify and help agents adapt to new goals, reward functions, and environment dynamics. This capability is not exercised at all by standard RL benchmarks like strategy games (which typically evaluate agents against a single or small number of fixed reward functions). It is evaluated in only a limited way by existing instruction following benchmarks, which operate in closed task domains (navigation, object manipulation, etc.) and closed worlds. The simplicity of these tasks is often reflected in the simplicity of the language that describes them, with small vocabulary sizes and multiple pieces of independent evidence for the grounding of each word.

Real natural language has important statistical properties, such as the power-law distribution of word frequencies, that do not appear in environments with synthetically generated language and small numbers of entities [Zipf, 1949]. Without environments that encourage humans to talk about (and force agents to learn from) complex composition and the "long tail" of lexicon entries, we cannot hope that RL methods that incorporate language in *closed-world tasks* will generalize at all outside of such tasks.

A natural starting point is provided by open-world video games like Minecraft [Johnson *et al.*, 2016], in which users are free to assemble complex structures from simple parts, and thus have an essentially unlimited universe of possible objects to describe and goals involving those objects. Looking ahead, as core machine learning tools for learning from feedback and demonstrations become sample-efficient enough to use in the real world, we anticipate that tools combining language and RL will find applications as wide-ranging as autonomous vehicles, virtual assistants and household robots.

5 Conclusion

The currently predominant way RL agents are trained restricts their use to environments where all information about the policy can be gathered from directly acting in and receiving reward from the environment. This *tabula rasa* learning results in low sample efficiency and poor performance when transferring to other environments. Utilizing natural language in RL agents could drastically change this by transferring knowledge from natural language corpora to RL tasks, as well as between tasks, consequently unlocking RL for more diverse and real-world tasks. While there is a growing body of papers that incorporate language into RL, most of the research effort has been focused on simple RL tasks and synthetic languages, with highly structured and instructive text.

To realize the potential of language in RL, we advocate for more research into learning from unstructured or descriptive language corpora, with a greater use of NLP tools like pre-trained language models. Such research also requires development of more challenging environments that reflect the semantics and diversity of the real world.

References

[Agarwal *et al.*, 2019] Rishabh Agarwal, Chen Liang, Dale Schuurmans, and Mohammad Norouzi. Learning to Generalize from Sparse and Underspecified Rewards. *arXiv:1902.07198 [cs, stat]*, 2019.

[Andreas *et al.*, 2016] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *CVPR*, 2016.

[Andreas *et al.*, 2017] Jacob Andreas, Dan Klein, and Sergey Levine. Modular Multitask Reinforcement Learning with Policy Sketches. In *ICML*, 2017.

[Andreas *et al.*, 2018] Jacob Andreas, Dan Klein, and Sergey Levine. Learning with latent language. In *NAACL-HLT*, 2018.

[Antol *et al.*, 2015] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *ICCV*, 2015.

[Arulkumaran *et al.*, 2017] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. A Brief Survey of Deep Reinforcement Learning. *IEEE Signal Proc. Magazine*, 2017.

[Bahdanau *et al.*, 2019] Dzmitry Bahdanau, Felix Hill, Jan Leike, Edward Hughes, Arian Hosseini, Pushmeet Kohli, and Edward Grefenstette. Learning to Understand Goal Specifications by Modelling Reward. In *ICLR*, 2019.

[Banko *et al.*, 2007] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open information extraction from the web. In *IJCAI*, 2007.

[Barto and Mahadevan, 2003] Andrew G Barto and Sridhar Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete event dynamic systems*, 13(1-2):41–77, 2003.

[Bouziane *et al.*, 2015] Abdelghani Bouziane, Djelloul Bouchiha, Noureddine Doumi, and Mimoun Malki. Question answering systems: survey and trends. *Procedia Computer Science*, 73:366–375, 2015.

[Branavan *et al.*, 2009] S. R. K. Branavan, Harr Chen, Luke S. Zettlemoyer, and Regina Barzilay. Reinforcement Learning for Mapping Instructions to Actions. In *ACL*, 2009.

[Branavan *et al.*, 2010] S. R. K. Branavan, Luke S Zettlemoyer, and Regina Barzilay. Reading between the lines: Learning to map high-level instructions to commands. In *ACL*, 2010.

[Branavan *et al.*, 2012] S. R. K. Branavan, David Silver, and Regina Barzilay. Learning to Win by Reading Manuals in a Monte-Carlo Framework. *JAIR*, 2012.

[Chen and Mooney, 2011] David L Chen and Raymond J Mooney. Learning to interpret natural language navigation instructions from observations. In *AAAI*, 2011.

[Chen *et al.*, 2017] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. A survey on dialogue systems: Recent advances and new frontiers. *ACM SIGKDD Explorations Newsletter*, 2017.

[Chen *et al.*, 2018] Howard Chen, Alane Shur, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. *arXiv preprint arXiv:1811.12354*, 2018.

[Chevalier-Boisvert *et al.*, 2018] Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. Minimalistic gridworld environment for openai gym. https://github.com/maximecb/gym-minigrid, 2018.

[Côté *et al.*, 2018] Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. TextWorld: A Learning Environment for Text-based Games. *arXiv:1806.11532 [cs, stat]*, 2018.

[Das *et al.*, 2017] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied Question Answering. *CVPR*, 2017.

[Das *et al.*, 2018] Abhishek Das, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Neural Modular Control for Embodied Question Answering. *CoRL*, 2018.

[Deerwester *et al.*, 1990] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[Eisenstein *et al.*, 2009] Jacob Eisenstein, James Clarke, Dan Goldwasser, and Dan Roth. Reading to learn: constructing features from semantic abstracts. In *ACL*, 2009.

[Firth, 1957] John R Firth. A synopsis of linguistic theory, 1957.

[Frome *et al.*, 2013] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc Aurelio Ranzato, and Tomas Mikolov. DeViSE: A Deep Visual-Semantic Embedding Model. In *NIPS*, 2013.

[Fu *et al.*, 2019] Justin Fu, Anoop Korattikara, Sergey Levine, and Sergio Guadarrama. From Language to Goals: Inverse Reinforcement Learning for Vision-Based Instruction Following. In *ICLR*, 2019.

[Goldberg, 2019] Yoav Goldberg. Assessing BERT's Syntactic Abilities. *CoRR*, abs/1901.05287, 2019.

[Gopnik and Meltzoff, 1987] Alison Gopnik and Andrew Meltzoff. The development of categorization in the second year and its relation to other cognitive and linguistic developments. *Child development*, 1987.

[Goyal *et al.*, 2019] Prasoon Goyal, Scott Niekum, and Raymond J. Mooney. Using Natural Language for Reward Shaping in Reinforcement Learning. *arXiv:1903.02020 [cs, stat]*, 2019.

[Hermann *et al.*, 2017] Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, Marcus Wainwright, Chris Apps, Demis Hassabis, and Phil Blunsom. Grounded Language Learning in a Simulated 3d World. *arXiv:1706.06551 [cs, stat]*, 2017.

[Ho and Ermon, 2016] Jonathan Ho and Stefano Ermon. Generative Adversarial Imitation Learning. In *NIPS*, 2016.

[Janner *et al.*, 2018] Michael Janner, Karthik Narasimhan, and Regina Barzilay. Representation learning for grounded spatial reasoning. *TACL*, 2018.

[Johnson *et al.*, 2016] Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The malmo platform for artificial intelligence experimentation. In *IJCAI*, pages 4246–4247, 2016.

[Johnson *et al.*, 2017] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017.

[Kostka *et al.*, 2017] B. Kostka, J. Kwiecieli, J. Kowalski, and P. Rychlikowski. Text-based adventures of the golovin AI agent. In *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, 2017.

[MacMahon *et al.*, 2006] Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. Walk the talk: Connecting language, knowledge, and action in route instructions. In *AAAI*, 2006.

[Mei *et al.*, 2016] Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. Listen, Attend, and Walk: Neural Mapping of Navigational Instructions to Action Sequences. *AAAI*, 2016.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. *NIPS*, 2013.

[Misra *et al.*, 2017] Dipendra Misra, John Langford, and Yoav Artzi. Mapping Instructions and Visual Observations to Actions with Reinforcement Learning. *EMNLP*, 2017.

[Narasimhan *et al.*, 2015] Karthik Narasimhan, Tejas D. Kulkarni, and Regina Barzilay. Language understanding for text-based games using deep reinforcement learning. In *EMNLP*, 2015.

[Narasimhan *et al.*, 2018] Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. Grounding Language for Transfer in Deep Reinforcement Learning. *JAIR*, 2018.

[Oh *et al.*, 2017] Junhyuk Oh, Satinder P. Singh, Honglak Lee, and Pushmeet Kohli. Zero-shot task generalization with multi-task deep reinforcement learning. In *ICML*, 2017.

[Peters *et al.*, 2018a] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *NAACL*, 2018.

[Peters *et al.*, 2018b] Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. Dissecting contextual word embeddings: Architecture and representation. In *EMNLP*, 2018.

[Radford *et al.*, 2019] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[Shu *et al.*, 2018] Tianmin Shu, Caiming Xiong, and Richard Socher. Hierarchical and Interpretable Skill Acquisition in Multi-task Reinforcement Learning. *ICLR*, 2018.

[Shusterman *et al.*, 2011] Anna Shusterman, Sang Ah Lee, and Elizabeth Spelke. Cognitive effects of language on human navigation. *Cognition*, 2011.

[Silver *et al.*, 2017] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 2017.

[Singh *et al.*, 2002] Satinder Singh, Diane Litman, Michael Kearns, and Marilyn Walker. Optimizing dialogue management with reinforcement learning: Experiments with the njfun system. *JAIR*, 2002.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Tellex *et al.*, 2011] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI*, 2011.

[Tesauro, 1995] Gerald Tesauro. Temporal Difference Learning and TD-Gammon. *Communications of the ACM*, 1995.

[Torrado *et al.*, 2018] Ruben Rodriguez Torrado, Philip Bontrager, Julian Togelius, Jialin Liu, and Diego Perez-Liebana. Deep reinforcement learning for general video game ai. In *CIG*. IEEE, 2018.

[Wang *et al.*, 2016] Sida I Wang, Percy Liang, and Christopher D Manning. Learning Language Games through Interaction. In *ACL*, 2016.

[Wang *et al.*, 2019] Xin Wang, Qiuyuan Huang, Asli Çelikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *CVPR*, 2019.

[White and Sofge, 1992] David Ashley White and Donald A Sofge. *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptative Approaches*. Van Nostrand Reinhold Company, 1992.

[Yu *et al.*, 2018] Haonan Yu, Haichao Zhang, and Wei Xu. Interactive Grounded Language Acquisition and Generalization in a 2d World. *ICLR*, 2018.

[Ziebart *et al.*, 2008] Brian D Ziebart, Andrew Maas, J Andrew Bagnell, and Anind K Dey. Maximum Entropy Inverse Reinforcement Learning. In *AAAI*, 2008.

[Zipf, 1949] George Kingsley Zipf. Human behavior and the principle of least effort. 1949.