

Integrating AI Planning with Natural Language Processing: A Combination of Explicit and Tacit Knowledge

KEBING JIN and HANKZ HANKUI ZHUO*, School of Computer Science and Engineering, Sun Yat-sen University, China

Natural language processing (NLP) aims at investigating the interactions between agents and humans, processing and analyzing large amounts of natural language data. Large-scale language models play an important role in current natural language processing. However, the challenges of explainability and complexity come along with the developments of language models. One way is to introduce logical relations and rules into natural language processing models, such as making use of Automated Planning. Automated planning (AI planning) focuses on building symbolic domain models and synthesizing plans to transit initial states to goals based on domain models. Recently, there have been plenty of works related to these two fields, which have the abilities to generate explicit knowledge, e.g., preconditions and effects of action models, and learn from tacit knowledge, e.g., neural models, respectively. Integrating AI planning and natural language processing effectively improves the communication between human and intelligent agents. This paper outlines the commons and relations between AI planning and natural language processing, argues that each of them can effectively impact on the other one by five areas: (1) planning-based text understanding, (2) planning-based natural language processing, (3) planning-based explainability, (4) text-based human-robot interaction, and (5) applications. We also explore some potential future issues between AI planning and natural language processing. To the best of our knowledge, this survey is the first work that addresses the deep connections between AI planning and Natural language processing.

CCS Concepts: • **Computing methodologies** → **Natural language processing; Planning and scheduling**; *Information extraction; Natural language generation.*

Additional Key Words and Phrases: AI planning, Natural language processing, Natural language understanding, Human-robot interaction, Explainability

ACM Reference Format:

Kebing Jin and Hankz Hankui Zhuo. 2022. Integrating AI Planning with Natural Language Processing: A Combination of Explicit and Tacit Knowledge. *ACM Trans. Intell. Syst. Technol.* 1, 1 (April 2022), 24 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 INTRODUCTION

Natural language processing (NLP) aims at investigating the interactions between agents and humans, processing and analyzing large amounts of natural language data. In recent years, for attaining better performance and handling large corpora, building large-scale language models is an inevitable trend in real applications [62, 78, 126]. Despite the success of language models in various domains, the explainability and complexity of language models have drawn intense research interests recently. In order to make models explainable and lightweight, integrating models with

*Corresponding author

Authors' address: Kebing Jin, jinkb@mail2.sysu.edu.cn; Hankz Hankui Zhuo, zhuohank@mail.sysu.edu.cn, School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China, 510006.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

2157-6904/2022/4-ART \$15.00

<https://doi.org/XXXXXXX.XXXXXXX>

symbolic planning has been demonstrated effective in various NLP tasks. Symbolic planning (AI planning) is a branch of artificial intelligence that focuses on building symbolic domain models and synthesizing plans to transit initial states to goals based on domain models. The plans are typically for execution by intelligent agents, autonomous robots, and unmanned vehicles. Different from classical control and classification problems, the solutions are complex and must be discovered and optimized in multidimensional space. Generally, those approaches are mostly based on structured data, which has a well-defined structure and logically explainable to humans.

Compared with structured data used in AI planning, natural language descriptions are often complicated by omissions, inverted order, etc., resulting in difficulties in reasoning about language descriptions. It is thus often hard to directly train neural models to generate available and correct solutions, although deep learning has been widely used to handle unstructured data. Deep learning methods do well in acquiring knowledge from data, capturing implied rules, and expressing them by mathematical and neural models, which are tacit and unable to be directly shared with other humans and agents. Different from deep learning methods that aim to learn tacit knowledge, planning-based methods are better at capturing changes, formalizing them by rules, and generating valid plans when handling structured data. Rules are already codified, namely explicit knowledge, which can be clearly expressed and easily shared with others. Therefore, AI planning is one of the considerable steps to understand implied rules and build domain models from large amount of texts in natural language processing [32, 72].

On the other hand, unstructured data in real world is not disorderly but often a sequence based on rules. As for a natural language description, there is a theme running through it, along with a series of relevant events and a coherent text unfolds. Each sentence relates to the preceding texts and influences following sentences, just like preconditions and effects of actions in AI planning. *For example, in a recipe about making a meatloaf shown in Figure 1(a), humans can easily understand it and capture the main information including verbs, e.g., “Heat”, and objects, e.g., “butter” and “skillet”. However, as for agents, when given a mass of data in the form of sentences, it is hard to directly build models to reason about the implied rules and predict next moves. If we extract these information and formalize them structurally, as shown in Figure 1(b), it is easier to construct models based on planning methods for guiding future unseen tasks.*

Besides using AI planning to help reason about implied rules in texts, the power of AI planning about capturing implied relations and computing valid solutions is another effective way to improve natural language processing, such as text summarization and machine translation. For example, there have been planning-based text generation methods [57, 119] extending a clear storyline ordered in advance. Those methods first compute sequences composed of keywords, key phrases, or contents as storylines, then use natural language processing techniques to extend storylines to coherent texts. *In the above-mentioned example, generating an available recipe in a correct order shown in Figure 1(a) is hard. However, given some rules, such as domain models about the operations of cooking, agents can compute plans toward achieving specified goals like a theme about making a meatloaf, as shown in Figure 1(b). Agents can easily extend the plan and gain a valid recipe.*

The integration of AI planning and natural language processing combines the best of tacit knowledge learning from sentences and explicit knowledge in the form of rules. As discussed in [52], it would be more effective to combine explicit and tacit knowledge rather than giving up explicit knowledge and learning everything from tacit knowledge, which is the current trend. Integrating AI planning and natural language processing allows human to communicate with agents in a more comfortable way, and enables intelligent agents to explain themselves to human in a human-understandable way. Natural language, as the most comfortable way to communicate with humans, establishes a relationship between humans and intelligent agents. In recent years, researchers have made efforts to connect with natural language and robots, such as by dialogue

Heat butter in a large **skillet**. **Add** the **onions**, **garlic**, **celery** and **carrot**. **Cook** the **mixture** over **medium heat** for **5 minutes**, **stirring** frequently. **Add** the **ketchup sauce** and the **salt** and **pepper**. **Cook** the **mixture** for another **minute**. **Combine** the **meat**, **bread crumbs**, **eggs** and **parsley** in a mixing **bowl**. **Add** the **cooked vegetables** from the **skillet**. **Mix everything** well together. **Push** the **mixture** into the **baking loaf pan**. Then **put** the **loaf** into the **oven** and **bake** for about **an hour**. When the **meatloaf** is done, allow it to **cool** slightly, then slowly **remove** the **loaf** from the baking pan by **cutting** along the **sides** with a butter **knife**. Gently **lift** the **meatloaf** from the **pan** and **place** on serving **plate**. **Cut** into slices of approximately **3/4 inch thick**.

(a) A recipe for making a meatloaf.

Heat(butter, skillet) → Add (onions, garlic, celery, carrot, skillet) → Cook(mixture, medium heat, 5 minutes) → stirring() → Add(ketchup sauce, salt, pepper, skillet) → Cook(mixture, medium heat, one minute) → Combine(meat, bread crumbs, eggs, parsley, bowl) → Add(cooked vegetables, bowl) → Mix() → Push(mixture, loaf pan) → Put(loaf, oven) → Bake(an hour) → Cool() → Remove(loaf) → Cut(sides, knife) → Lift(meatloaf, pan) → Place(meatloaf, plate) → Cut(meatloaf, 3/4 inch thick)

(b) A trace extracted from (a).

Fig. 1. An example textual recipe about making a meatloaf.

systems [82, 105] and natural language commands understanding [59, 108]. On the other hand, planning-based natural language models are based on structured data or implied rules, such as predicted storylines, which allows human to partly understand the principles of models.

In this paper, we first introduce some background knowledge in AI planning and natural language processing as well as their relations. Then we give a comprehensive overview of integrating AI planning and natural language processing by four aspects and their challenges: planning-based text understanding, planning-based natural language processing, planning-based explainability, and text-based human-robot interaction. Their relations are shown in Figure 2. Firstly, planning-based natural language understanding includes extracting actions from texts and learning domain models from texts. Secondly, we introduce planning-based natural language processing by three tasks integrated with AI planning, i.e., text generation, text summarization, and machine translation. Then we discuss planning-based explainability. Next, we introduce text-based human-robot interaction by extracting actions from natural language instructions, natural language command understanding, and dialogue generation. Finally, we present current applications, several future directions and conclude this paper. To the best of our knowledge, this survey is the first work that addresses the deep connections between AI planning and NLP.

2 PLANNING DOMAIN DESCRIPTION LANGUAGE AND NLP

In this section, we introduce modeling knowledge in AI planning, backgrounds in natural language processing (NLP), and relations between AI planning and NLP, including similarities, differences, and language model-based planning.

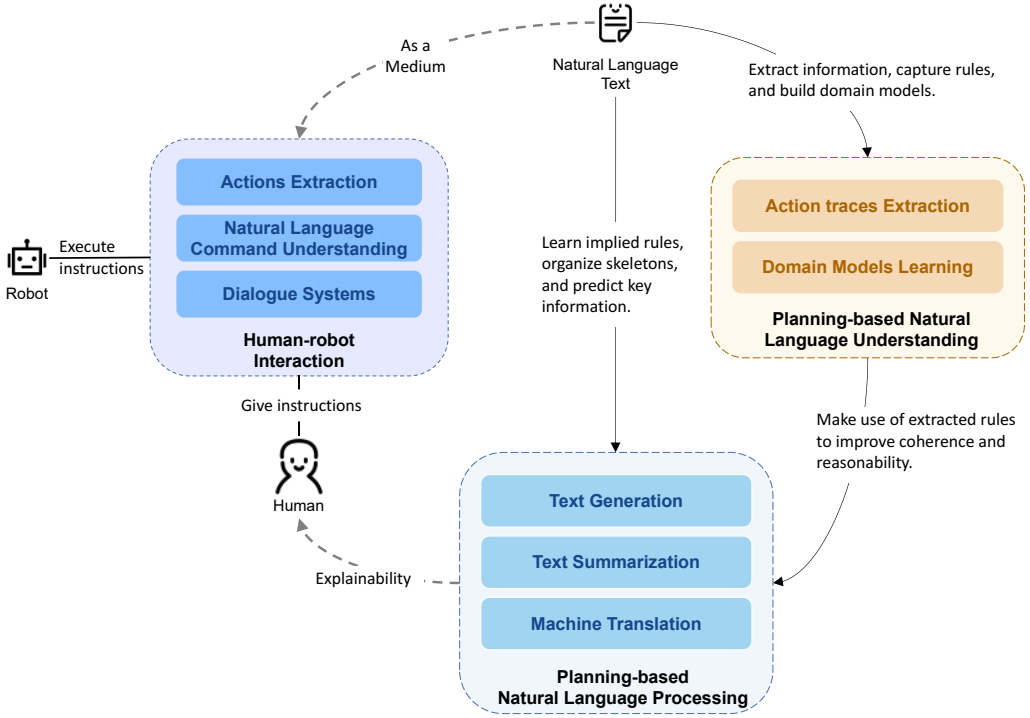


Fig. 2. Relations between AI planning and Natural language processing

2.1 Planning domain description language

A planning problem is composed of a planning domain \mathcal{D} and an instance p , defined by planning domain description language. With the development of AI planning, more and more extended planning domain description languages [34, 39, 92] have been proposed. Taking PDDL (Planning Domain Definition Language) [71] as an example, a planning domain \mathcal{D} is made up by several action models. An action model is defined by a tuple of $A = \langle a, pre(a), eff(a) \rangle$, where a is an action name with zero or more types of parameters. An action is a grounding of an action model, each of whose parameters is an object. $pre(a)$ is a set of preconditions requiring to be satisfied when executing a , each of which is a proposition or a numeric constraint. Similarly, $eff(a)$ is a set of effects, an effect can be a topical proposition, added into or deleted from the state after executing a , or a numeric updating, increasing or decreasing the value of variables according to specified functions. An instance is defined by $p = \langle s_0, g, \xi \rangle$, where s_0 is a set of initial assignments by propositions and variables, and g is a set of goals requiring to be achieved. ξ is an objective function guiding planner to compute for a minimum cost or maximum reward. A planning problem is to compute an available action sequence, which can transfer s_0 to a state containing desired goals g .

For example, parts of action models in the Rover domain are shown in Figure 3(a), where “(equipped_for_imaging ?r)” is a proposition asking that a rover “?r” should equip with a camera for taking image when executing action “Calibrate”. “(>= (energy ?r) 1)” is a numeric precondition requiring the energy of rover “?r” should be larger than 1. Figure 3(b) and (c) show an initial state and goals, respectively. An example valid plan, updating the initial state to a state achieving the goals, is shown in Figure 3(d), where each action is a grounding action model with parameters.

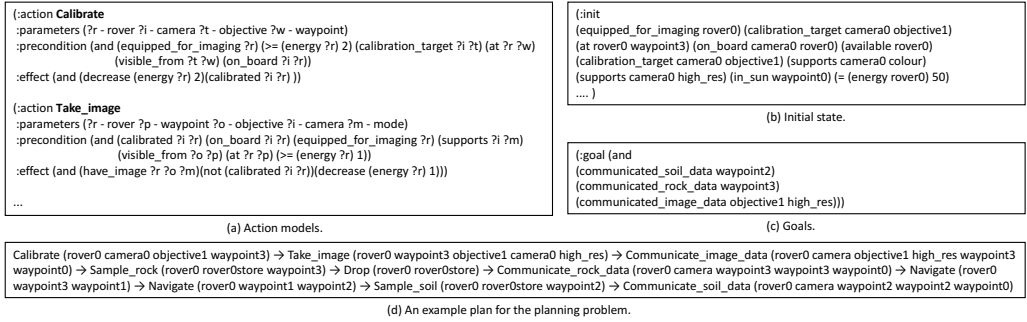


Fig. 3. An example in the Rover domain, including action models, initial state, goals, and an available plan.

2.2 Natural language processing

Recently, natural language processing (NLP) [17, 54, 114] has attracted lots of attention, and it builds a bridge between human and agents. Natural language processing is grand, including various fields, such as natural language understanding (NLU)[81, 107], natural language generation (NLG), [30, 37] machine translation[80], and spelling correction[43]. NLP has undergone several stages of rule-based models, statistic-based models, and neural network models. Rule-based NLP [97] is led by hand-crafted rule sets, whose main task is to understand natural language. It is, however, difficult and time-consuming to build all hand-crafted rules, lacking of scalability. Statistic-based NLP [65] makes use of probability distributions to generate proper words and sentences, promoting the application of statistical machine learning methods based on large-scale corpora in natural language processing. Nevertheless, statistic-based models is barely to capture long-term relations and use information included in contexts. Recently, deep learning has been widely used in NLP tasks[106], it is able to capture tacit knowledge implied in texts. However, deep learning is to fit neural networks and predict based on statistics, it can not “understand” the real meaning in natural language. In this paper, we focus on those NLP tasks related to AI planning. Compared with NLP approaches totally based on deep learning, planning-based NLP methods are more curious about implied logic and reasons of the solutions.

2.3 Relations between AI planning and natural language processing

In this section, we will sketch the relations between AI planning and natural language processing, which is mentioned by [38, 116, 122]. We will first introduce the similarities between AI planning and natural language processing, and then talk about their differences and deep relations.

We discuss the similarities between AI planning and natural language processing by two aspects. First of all, AI planning and natural language processing both revolve around observations and knowledge [38]. Planning tasks aim at either solving problems based on a current observation and goals along with priori knowledge like action models, or constructing knowledge such as transition functions based on sequential observations. Similarly, in natural language processing tasks, a text can be regarded as a sequence of observations, each observation is a sentence describing a partially observed state, where observations changes following implied rules. Secondly, they share two major common problems in planning tasks and natural language processing tasks: consistency and diversity. Both of plan traces and texts are cohesive descriptions and stringed by some rules and goals. As for planning, the rules are action models and transition functions, which update current states to next ones. The goals are sets of goal states or objective functions, guiding planners to attain optimal solutions. As for natural language processing, the rules are implied in the organization of

texts, such as transition connections or consequences. Their goals can be titles, themes, or topics. On the other hand, in natural language processing, given goals, we can generate lots of coherent texts composed of different events, similar to different plan traces computed for the same goal states. Moreover, a sequence of events can be written as various stylized texts. As shown in Table 1, we enumerate some concepts in AI planning and natural language processing, which can have some similarities. *For example, objects, such as “rover0” and “objective1” in Figure 3, in planning problems are similar to entities in texts, e.g., “skillet” and “onions” in Figure 1.*

Table 1. Similar concepts between AI planning and natural language processing

AI planning	Natural language texts
Objects	Entities
States	Sentences, sentiments, and intentions
Actions	Events
Domain models	Implied rules, transitions, and relations
Goals	Topics, and themes
Plan traces	Storylines, skeletons, and frameworks

Although AI planning and natural language processing have those commons, the difference between them is that AI planning is good at generating explicit knowledge, such as domain models [128–131], while natural language processing often learns tacit knowledge, such as training models from natural language data. [52] argues that AI systems should be able to know when to take advice and when to learn, to find a balance between explicit and tacit knowledge. Taking planning-based text generation as an example, although texts are required to be coherent and with correct logic, natural language processing is weak in computing available and valid events, which AI planning is good at. And the integration of both allows agents to generate coherent texts by first using AI planning to generate storylines and then learning text generator by natural language processing techniques.

In a word, there are close ties between AI planning and natural language processing, due to their different advantages of explicit and tacit knowledge. The combination allows each of them to effectively impact on the other one.

2.4 Language Model-based Planning

Although AI planning can effectively capture rules from action sequences, it is hard for humans without expert knowledge to construct structured plans. A natural and intuitive way is to use human natural language to describe plans, asking language model-based planners to be able to handle text sequences and predict the next moves [94, 132]. Prior works mostly make use of pre-trained language models (LMs) to understand abstract, high-level textual actions and learn actionable knowledge for guiding planning [47, 50, 61]. Specifically, Huang et al. [46] use large language models (LLMs) to generate natural language actions, they investigated actionable knowledge already contained pre-trained LLMs. On the other hand, some works map natural language instructions and high-level goals to actions and goals, and learn policy to make decisions [98]. For example, LID [61] uses policies initialized with pre-trained LMs and fine-tunes policies for predicting actions, validating that LMs are able to contain rich actionable knowledge. Language model-based planners are mostly based on templated textual actions datasets, rather than complex natural language instructions with various styles of descriptions. In the following sections, we introduce deep

connections between AI planning and natural language processing, to rise to the dual challenges from rules and natural language.

3 PLANNING-BASED NATURAL LANGUAGE UNDERSTANDING

In this section, we introduce a comprehension overview of natural language understanding based on AI planning. Natural language understanding aims at comprehending human language, including sentiments, relations in contexts, topics, etc. Compared with learning relations from structured state traces, learning relations between extracted events from texts are even more challenging. It asks agents to reason about contexts, capture themes of sentences by selecting words to represent them, compute causal relationships between the selected words. There are two major areas introduced in the following section, which are both based on AI planning, to understand the texts and learn causal relationships from texts. The first one is to extract action sequences from texts, which requires agents to understand complex contexts from action descriptions. Moreover, it requires agents to be capable of reasoning about connotations in texts, such as exclusion relations and optional relations between actions. The other one is to first select words to represent the main ideas of sentences, and then learn the rules implied in sentences and formalize them by readable domain models for guiding agents to solve future unseen tasks and helping agents and people understand logical relations between events.

3.1 Extracting actions from texts

There have been works on extracting action sequences from action descriptions [127]. The inputs of the task mostly include some texts describing some actions and procedures, the outputs are action sequences from the texts, each action is composed of a verb as action name and some objects. Figure 4 shows an example in [32], where an input text is in left part of Figure 4, extracting action traces are shown in the right part of Figure 4, the relations between actions are shown in the middle of Figure 4. This task does not only need extract a word standing for an action of the sentence, but also reason about contexts for completing omissions caused by pronouns. Early approaches [55, 70] mostly make use of specialized resources, such as semantic parsers and learned lexicons, to reason about natural language route instructions. For example, MARCO [68] was proposed to

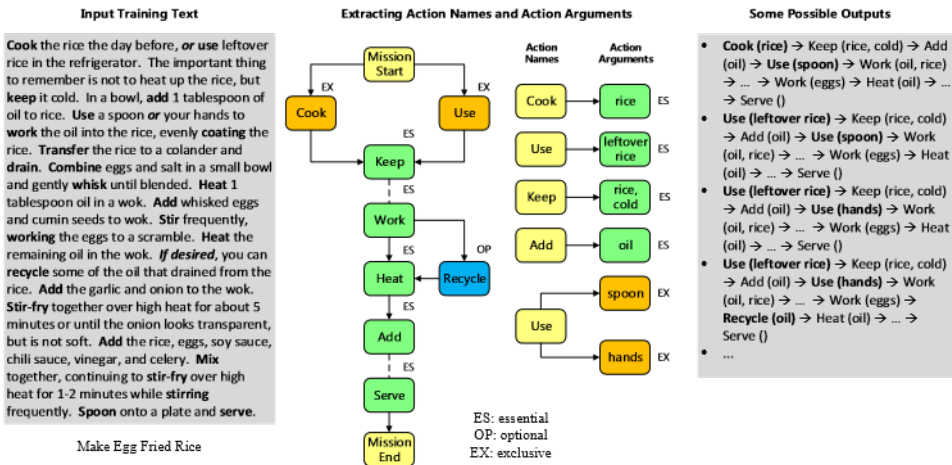


Fig. 4. An illustration of action sequence extraction problem in [32]

map free-form natural language route instructions to action sequences, arising great interest in natural language processing community. MARCO is able to model a sentence by an instruction, e.g., “Turn to face the green hallway” can be modeled by “Turn(until=(object=Path, appear=Green, side=Front, dist=0:))”. [22] presented a system along with a plan refinement algorithm to transform natural language navigation instructions into executable formal plans. Generally, methods with semantics parsers require high simplicity of the texts. Therefore, mostly approaches are based on instructional texts or similar texts following some templates.

In recent years, learning methods, such as reinforcement learning and LSTM, have been widely used in natural language processing, as well as extracting action sequences from natural language texts, with the rapid development of artificial intelligence. For example, [13] proposed a reinforcement learning approach for mapping natural language instructions in two domains, Windows troubleshooting guides and game tutorials, to sequences of executable actions. It uses a reward function to define the quality of the executed actions, and a policy gradient algorithm to estimate the parameters of a log-linear model for action selection. The learner repeatedly constructs action sequences for a set of documents, executes those actions, and observes the resulting reward. To handle free natural language without restricted templates, EASDRL [32] was presented to extract action sequences from texts, making use of deep reinforcement learning. It builds Q-networks to learn policies of extracting actions and extract plans from the labeled texts. EASDRL regards texts associated with actions as “states”, and associating words in texts with labels as “actions”. During capturing relations, EASDRL considers previously extracted actions as parts of states for deciding the choice of next operations. Therefore, EASDRL is able to reason about connotations in texts, such as exclusion relations and optional relations between actions. Except for reinforcement learning, there are more techniques used in extracting actions from texts. With the help of with long short-term memory (LSTM) recurrent neural networks, Mei et al. [72] proposed a neural sequence-to-sequence model to translate single-sentence natural language instructions to action sequences based upon a representation of the observable world state. LSTMs are applicable to a number of sequence learning problems, due to their ability to learn long-term dependencies, and they have been shown to be effective in tasks existing sequences. The LSTM framework allows agents to bidirectionally encode the navigational instruction sequence and decode the representation to an action sequence, based on a representation of the current state.

3.2 Learning domain models from texts

Besides extracting action sequences from texts, another way to understand text is to learn the implied relations from sentences. The input of the learning task is a set of texts, and the output is a planning domain model composed of action models describing the relations by propositional preconditions and effects following the syntax of planning domain description language, such as PDDL [71]. Preconditions and effects make use of propositions to describe conditions that must be satisfied when executing actions and results after executing them, respectively.

Learning domain models from instructional texts is a little different from narrative stories. Instructional texts are simpler than narrative stories, and the words are domain-dependent. Narrative stories are mostly third person synopses and they are always along with omission, which results in complexity when parsing sentences. A general way to construct a domain model is first to extract words and objects by parsing sentences for annotations (e.g., OpenNLP¹ and Stanford CoreNLP²), and then learn causal relationships between them and formalize the relations by action models.

¹<https://opennlp.apache.org/>

²<http://stanfordnlp.github.io/CoreNLP/>

To learn domain models from instructional texts, Sil and Yates [103] used text mining via a search method to identify documents that contain words that represent target verbs or events. Then they used inductive learning techniques to identify appropriate action preconditions and effects. The method relies on handcrafted Pointwise Mutual Information to learn a SVM-based classifier that scores preconditions for a given action. Branavan et al. [14] presented a reinforcement learning framework to extract precondition and effects relations implied by the text, and used these relations to compute action sequences for completing given tasks in the environment. Single argument predicates are extracted from the text as states, and regarded as sub-goals to construct hierarchical planning problems. Yordanova and Kirste [121] extracted verbs and objects from text instructions based on part of speech (POS) tagging module, and discovered causal relations on the basis of the order of appearance to build PDDL models. However, due to lacking of connections between texts and world states and analyses between variable texts with the same meaning, it is hard to directly construct domain models as learning action models from structured data. In this paper, domain models are constructed according to some templates after parsing sentences. For example, “If the apple is ripe, put the apple on the table.” indicates that “ripe”, a state of an apple, is a precondition of action “put”. Therefore, a precondition of action “put” is “(state-ripe)”. Although the model is readable for human, it is kind of redundant and not easy to understand for agents because of synonyms and polysemous words. Similarly, Lindsay et al. [64] assumed texts are in restricted templates when describing actions. They generated sequences of actions by constructing representations of sentences and cluster operators by computing similarity, and built PDDL domain models with the help of a domain model acquisition tool.

Considering the power of AI planning about offering correct causality and flexible narrative generation possibilities, constructing domain models has been used in narrative systems recently. Hayton et al. [42] proposed an approach taking natural language sentences which summarise the main elements of stories as inputs and generating action representations following PDDL, a narrative planning domain model. To overcome difficulties in parsing narrative stories, they presented two sets of rules to handle pronouns in stories. Then they used a template similar to [121] to construct planning domains. Another specific difficulty for planning-based narrative systems is that hand-crafted domain models require more narrative actions and types of narrative objects compared to generated planning domains. The plentiful actions and objects let generated plan traces and storyline be more interesting and they can be extended to enjoyable stories. To achieve it, Porteous et al. [87] tried to anticipate the consequences of plan failure and the remedial actions or objects needed, or described several potential alternatives. They extended narrative planning domains by two types of principled mechanisms to operationalize narrative action and object substitution during narrative plan authoring. An original domain model can be extended with the addition generated by two mechanisms alternately.

3.3 Challenges and future prospects

In AI planning, rules implied in states and actions are enforcedly constrained by preconditions and effects. Compared with AI planning, rules between events, such as concurrence, causality, and progression, in natural language processing are more flexible, which often intricately correlate with others. Moreover, the preconditions and effects are implied in natural language, which are abstract and hard to be modeled by propositions. In the other hand, in natural language text, a event can be written in different words, and a word owns various meanings. It therefore is difficult to directly distinguish them by only parsers. Those challenges make it hard to extract detailed logical relations implied in natural language, let along model implied relations by constructing structural domain models. It might be interesting to dig out clear logical relations implied in natural language

text, which can lay a foundation for natural language processing tasks, such as explainability and controllable text generation.

4 PLANNING-BASED NATURAL LANGUAGE PROCESSING

In this section, we introduce three natural language tasks integrated with AI planning, including text generation, text summarization, and machine translation. Planning-based natural language processing tasks concern about reasonability and coherence, making use of the power of AI planning about reasoning about rules and relations.

4.1 Planning-based text generation

One important field combined with AI planning is text generation, in which there have been significant advances, recently. Text generation asks models to generate coherent and interesting text based on preceding parts of the text, topics, titles, or themes, requiring agents to be capable of generating valid and clear logical frameworks. AI planning is one of crucial steps to guide models to generate well-organized long texts, owing to its power in learning domain models and computing solutions for goal-driven tasks. In this section, we introduce planning-based text generation methods with respect to the following two features:

- **Symbolic planning text generation** combines text generation with a classical planning framework, taking prior knowledge, e.g., domain models formalized by planning domain description language, as extra inputs.
- **Neural planning text generation** are neural generators combined learning with a skeleton planning, to make up for the difficulties in building hand-crafted domain models.

4.1.1 Symbolic planning text generation. In order to generate coherent text with correct logic, it is natural to give agents some prior knowledge about basic rules between events. On the other hand, early rule-based researches [9, 21, 83] about natural language processing explore constructing representations of texts and combine with hand-crafted rules. It, however, is hard and tedious to enumerate all rules. Therefore, making use of AI planning to capture implied rules in the form of domain models with symbolic representations and compute proper skeletons is a natural way, which can overcome the difficulties of manually constructing rules [36, 56, 66]. For example, Porteous et al. [86] proposed an approach injecting narrative control into plan generation through the use of PDDL [71] state trajectory constraints, to express narrative control information within the planning representation. They constructed constraint trees according to input domain models, and injected control into automatically generated narratives system. With the help of constraints, the approach decomposes problems into sets of smaller subproblems using the temporal orderings described by the constraints, and solves subproblems incrementally by a planner. Intentional Partial Order Causal Link (IPOCL) planning framework [90] is an extension of classical planning, it aims at finding a sound and believable sequence of character actions that transforms an initial state into a state arriving goals. IPOCL does not only create causally sound plot progression, but also reasons about character intentionality by identifying possible character goals that explain their actions and creating plans that explain why those characters commit to their goals. Compared to IPOCL, CPOCL [115] preserves the conflicting subplans without damaging the causal soundness of the overall story to generate interesting stories. CPOCL is an extension of IPOCL that explicitly captures how characters can thwart one another in pursuit of their goals, which is the essence of narrative conflicts. Making use of hand-crafted, well-defined domain models, symbolic planning text generation methods have the ability to produce impressive results in limited domains.

4.1.2 Neural planning text generation. However, it is often tedious or difficult to build domain models by hand due to the high requirements of manual efforts and domain knowledge. Automatically learning domains and constructing storylines have significantly attracted researchers' attention recently [88, 104].

To automatically learn domain models for helping generate coherent and valid stories, Li et al. [60] used a crowd-sourced corpus of stories to learn plot graphs that can then be used as constrained search spaces for sequences of story events, instead of relying on priori domain models. Specifically, the approach crowdsources a corpus of narrative examples of a new domain, automatically constructs domain models capturing different possible, non-contradictory story trajectories, and samples from the space of stories allowed by the domain model according to some story quality criteria. During the plot graph learning, learning mutual exclusion relations and optional events lets the generated story be coherent. C2PO [2] learns a branching story graph structure that can be searched, and introduces soft causal relations as causal relations inferred from commonsense reasoning. It creates a branching space of possible story continuations that bridge between plot points that are automatically extracted from existing natural language plot summaries.

Another way for constructing valid line of text is to construct storylines in advance, which can be skeletons, or sequences of keywords, key phrases, or contents. Xu et al. [118] generated skeletons composed of phrases learned by a reinforcement learning method, and then expanded skeletons to complete and fluent sentences. Fan et al. [31] proposed a novel approach which first generates plans in the form of predicate-argument structures, then generates stories with placeholder tokens to indicate entities, and finally replaces tokens by entities based on the global story contexts. The inputs of the task are short descriptions of scenes or events, and the approach outputs relevant narrative stories following the inputs.

Instead of generating skeletons with detailed prompts, some approaches first plan out storylines, which enable them to generate controllable stories with goals. [119] proposed a hierarchical generation framework that first planned a controllable storyline composed of keywords towards a goal (i.e., a title), and then generated a story based on the storyline. The RAKE algorithm [91] takes each sentence as an input and combines several word frequency based and graph-based metrics to weight the importance of the words. The approach regards the most important word as the keyword. The storyline is planned out based on the title, previously generated sentences, and the previous keywords in the storyline. In the experiments, they explore two strategies, dynamic schema and static schema. Results show the static schema performs better than the other one because it plans the storyline holistically, thus tends to generate more coherent and relevant stories. Similarly, [57] made use of a related framework, which first plans out storyline composed of a sequence of keywords and then generates the whole story, to handle stylized story generation. Stylized story generation is to generate stories with specified style given a leading context. Keywords are selected following some emotion-driven style, such as "fear", "anger", and "surprise". According to the stylized keywords, the approach can generate the whole stylized story with the guidance of the keywords. Yu et al. [123] followed Yao et al. [119] and used the RAKE algorithm to extract keywords to train a generation model for conducting keyword planning given story titles as inputs. Then the approach combines the story titles and the corresponding keywords of each story as the inputs of the graph module to automatically generate a graph for each story. According to the keywords, titles, story graphs, the approach encodes them into latent variables and further decodes them to generate the corresponding stories.

Except for generating stories according to prompts or goals, DYPLOC [44], a dynamic planning generation framework, takes a set of content items as inputs, each content item consists of a title, a set of entities, and a set of core concepts. To organize unordered content items, DYPLOC introduces

a plan scoring network, which learns to dynamically select and order contents based on what has been produced previously while generating the outputs.

4.1.3 Challenges and future prospects. Recently, text generation [49, 51, 124] has gained lots of attraction, aiming at letting intelligent agents express like humans. Numbers of methods to generate coherent texts have been proposed in recent years. Generating logical and controllable texts, however, is still a challenging task. AI planning is one of critical ways to enable agents to generate logical and controllable texts. Compared with symbolic planning text generation methods and neural planning text generation methods, the former is more capable of generating logical storylines with goals, and the texts generated by the latter are more diverse and coherent. Specially, symbolic planning methods can generate more explainable storylines, which is still challenging for deep learning methods. In general, it would be interesting to combine both for generating logical, controllable, and coherent text with diversity. Although there have been some approaches [31, 96, 109] proposed, based on the syntax of plans or representation of structure in AI planning, they use neural networks to predict unseen events instead of speculating based on logical relations implied. We hold the opinion that, compared with only learning blackbox neural models with implicit rules, appropriately combined with explicit logical relations would be a new attempt, which maybe contribute to natural language processing tasks.

4.2 Planning-based Text Summarization

Text summarization is to extract important information from texts and generate new texts based on those information in the form of summarizes. It requires agents to understand texts, filter information from abundant descriptions and organize them to form summarizes, which is one of the most researched areas among the NLP community. Text summarization can be categorized into extractive and abstractive techniques. Extractive summarization aims at selecting subsets of words or sentences from input articles to summarize them. Abstractive summarization takes articles as inputs, tries to understand the texts and generate summarizes. In recent years, some researchers try to integrate text summarization models with AI planning. The combination of AI planning and text summarization is mostly based on deep learning abstractive summarization, making use of content planning which describes or predicts skeletons of articles. Planning-based text summarization methods first plan out skeletons of summaries or compute probability distributions, and then generate the whole sentences based on the skeletons or predictions. For example, Narayan et al. [77] first computed plans in the form of entity chains, which are ordered sequences of entities, and then generated summaries conditioned on the plans. Marfurt et al. [69] proposed an abstractive summarization model implemented with a planning step, done by a hierarchical decoder, which first plans out an outline for the next sentence in the form of sentence representations and generates words according to the representations. Amplayo et al. [3] incorporated content planning in unsupervised summarization and datasets creation. They predicted aspect and sentiment probability distributions as content plans and generated sentences according to the predictions. During creating datasets, they made use of the distributions parametrized by the content planner to control the structures of created datasets.

In current planning-based text summarization approaches, content planning is based on deep learning which learns models and probability distributions to predict. However, only fitting deep learning models based on representations of sentences and words is hard to capture the relations implied. We believe that it would be interesting and challenging to let content planning modules understand relations implied in texts, which will allow content plans to be more coherent and controllable. On the other hand, although some mainstream text summarization methods are not based on AI planning, there are some commonalities between them. For example, tree-based and

graph-based text summarization approaches [4, 8, 58] first find the most important information from the text and then use trees and graphs to create summaries. Those structures aim at representing the relations between sentences, which is similar to the relations between actions in AI planning but at a more abstract level. Secondly, some text summarization methods try to obtain important words from sentences, such as verbs, objects, and subjects, to represent sentences semantically [1]. These forms have similarities with the structured representations in AI planning such as propositions and first-order predicates. Thirdly, ontology-based text summarization methods [74, 117] collect entities and their relationships, which reminds us of domain models in AI planning. We believe that there is a vast scope for researchers to combine AI planning and text summarization.

4.3 Planning-based Machine Translation

Machine translation aims at automatically translating content from source language to another target language, having a long history. One way to understand source texts and generate target texts is to combine neural language models with planning phases, i.e., first generating skeletons, and extending them by target languages. For example, Gülçehre et al. [41] integrated an auto-encoder with a planning mechanism, the auto-encoder first encodes texts by sequences of vector representations and decodes representations by generating target translation character-by-character. Specifically, they first created plans ahead in the form of action matrices, which are sequences of probability distributions, and made use of commitment plan vectors to govern whether to recompute plans or use them. Then they computed soft alignments based on the plan and generated texts in target language at each time-step. Shu and Nakayama [102] combined neural machine translation with a planning phase, which first generates planner codes to disambiguate uncertain information about the sentence structure and control the structure of output sentences. Bahdanau et al. [7] used actor-critic methods from reinforcement learning (RL) to generate sequences. They showed that sequences can be used in machine translation tasks, gaining better translation performance. Those approaches first produce big pictures of output texts by planning, then generate complete sentences conditioned on plans. They take advantages of capturing implied rules to generate more accurate and coherent target texts.

In natural language texts, the transitions between sentences imply relations and rules. Machine translation tasks do not only need to understand word-level structures of sentences, but also need to capture sentence-level relationships. Only relying on word-by-word text generation is hard and challenging to generate coherent and logical texts, especially when generating accurate transitions between sentences. Current approaches to capture implied rules and generate plans are mostly based on neural black-box models, lacking the explainability of making decisions. Previous researches about rule-based translation are explainable, they are based on hand-crafted rules. Although those rules are explicit and accurate, it is hard to manually write rules and the rules are not scalable. Producing rules are time-consuming and tedious. However, it would be interesting if we regard it as action models learning in planning community, which structurally formalize rules by preconditions and effects. We believe that combining rule-based machine translation with planning and neural machine translation may spark new ideas.

5 PLANNING-BASED EXPLAINABILITY

Nowadays, although deep learning approaches have been widely used in AI fields, human cannot understand practical meaning inside black-box neural models. Differently, AI planning is able to offer explicit knowledge, in the form of first-order logic, domain models, etc, implied in natural language. Therefore, the combination of natural language and AI planning may enable AI systems to be able to explain their reasoning to humans, which meets the needs for AI systems to work synergistically with humans. Those systems require agents to be aware of the intentions, capabilities

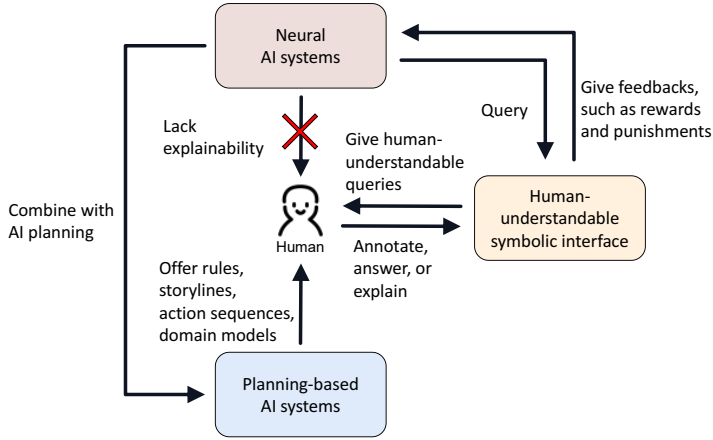


Fig. 5. Two types of making neural AI systems be explainable: (1) integrating AI systems with planning techniques; (2) making use of human-understandable interface.

and mental model of the human in the loop during its decision process. As shown in Figure 5, besides extracting action sequences [32, 68] and building domain models [14, 64] mentioned above, another way for making AI system explainable is to construct human-understandable symbolic interfaces (cf. [53]). A human-understandable symbolic interface is not only developed for its own computational efficiency, but also beneficial to humans. EXPAND system [40] and SERLfd framework [125], respectively. EXPAND system accelerates Human-in-the-Loop deep reinforcement learning by using human evaluative feedback and visual explanation. SERLfd uses self-explanation to recognize valuable high-level relational features as an interpretation of why a successful trajectory is successful, allowing SERLfd to guide itself and improve the efficiency.

6 TEXT-BASED HUMAN-ROBOT INTERACTION

The rapid developments of artificial intelligence let robots move out from industrial environments, and enter the daily life of humans, such as homes and hospitals. It requires robots to be able to respond quickly and effectively to rapidly-changing conditions and expectations. Language-based communication is the most natural method for humans to communicate with others, so natural language is a good candidate to be robot instruction for human-robot interaction. In this section, we will introduce text-based human-robot interaction from three aspects: extracting actions from natural language commands, natural language command understanding, and dialogue generation, as shown in the Figure 6.

6.1 Extracting actions from natural language command

One of important tasks in text-based human-robot interaction is to extract actions from natural language commands [48, 79, 85, 111]. Extracting actions from natural language commands is similar to action extraction in Section 3.1. Differently, extracting actions from natural language commands is not only to capture important words to indicate sentences, but also to understand implied rules and generate action sequences to guide robots to achieve tasks. Some approaches make use of with language descriptions, then build models that map language commands to action sequences. Tellex et al. [110] introduced a system, Generalized Grounding Graphs (G^3), taking a natural language command as input and outputting a plan for the robot. G^3 instantiates a

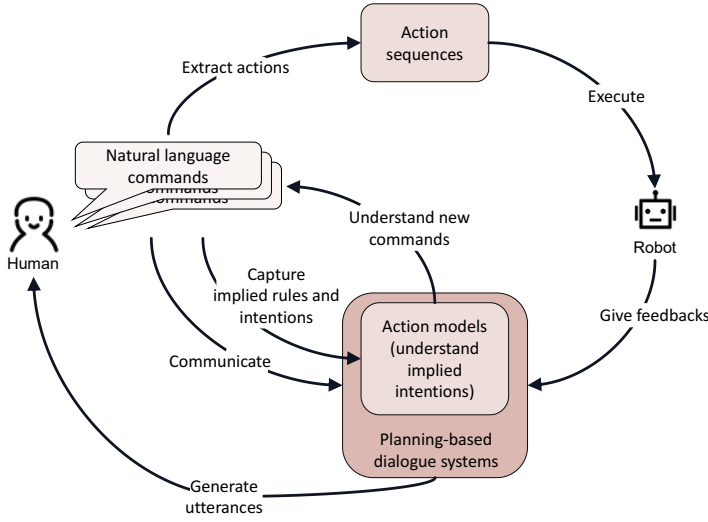


Fig. 6. Relations of researches in human-robot interaction.

probabilistic graphical model for a particular natural language command according to hierarchical and compositional semantic structure of the command. Cantrell et al. [20] presented a robotic architecture equipping with a planner that uses newly discovered information to produce new and updated plans, specifically information originating in spoken input produced by human operators. The robot can learn action sequences with defined preconditions and effects from natural language descriptions, and immediately apply this knowledge to improve planning. On the other hand, some approaches [16, 120] focus on temporal logic between natural language commands, aiming at handling semantic disambiguation of natural language.

6.2 Natural Language Command Understanding

Another important task is to enable agents to understand natural language commands given by humans, which requires agents to understand natural language commands and capture implied rules [10, 10, 12, 25, 25, 99], or learn new actions based on natural language commands and dialogues [35, 75, 95, 101]. To understand natural language commands, Thomason et al. [113] introduced a dialog agent to understand human natural language commands through semantic parsing, actively resolve ambiguities using a dialog manager, and incrementally learn from human-robot conversations. The agent employs incremental learning of a semantic parser from conversations on a mobile robot. It is implemented and tested both on a web interface with hundreds of users and on a mobile robot over several days, tasked with understanding navigation and delivery requests through natural language in an office environment. Brawer et al. [15] presented a framework for effectively grounding situated and natural language to action selection during human-robot interaction. It integrates verbal commands from a human partner with contextual information in the form of a task model. The approach is capable of acquiring and deploying new task representations from limited and natural language data sets, and without any prior domain knowledge of language or the task itself. Moreover, understanding action models and predicting next actions are widely researched for robotic tasks and navigation tasks with natural language commands [24, 26, 112].

On the other hand, another challenge in natural language command understanding tasks is to learn new actions when facing unknown natural language commands [6, 100]. Cantrell et al. [19] introduced an algorithm, to learn meanings of action verbs through dialogue-based natural language descriptions and integrated it in the robot's natural language subsystem. The algorithm allows robots to perform the actions associated with the learned verb meanings right away without any additional help or learning trials. Moreover, it allows human to interact with a robot to explain new action words in natural language, and lets the robot be able to perform the new action and store the procedural knowledge for future usage. Learning by Instruction Agent (LIA) [5] was proposed to learn new commands by natural language interaction with human. When facing a new natural language command that LIA does not understand, it prompts users to explain how to achieve the command through a sequence of natural language steps. LIA interprets commands using a semantic parser that maps each command to a logical form, which contains one or more of functions and predicates.

Undoubtedly, letting agents understand natural language commands and infer about next actions is an important but challenging task. An agent does not only need to capture rules implied in utterances, but also need to distinguish sentences described in different ways. We believe that the combination of planning-based natural language processing and human-robot interaction would create something interesting for natural language commands understanding tasks.

6.3 Dialogue Systems

Dialogue systems [23, 67] have been a bridge between human and robots, which interacts with human in natural language. AI planning is one of crucial mechanisms used in dialogue systems to recognize the intentions conveyed in dialogues [18, 76]. Planning-based dialogue systems take advantage of the power of capturing and expressing rules and use it to manage utterances or guide the generation. Rules, plans and intentions offer proper logical forms which derive appropriate communication acts in dialogue systems [28]. Plan-based model [27, 93] were proposed to manage the intentions and information implied in dialogues. Those models describe the common activities and relations between utterances, and can be used in the following generation processes. However, planning-based dialogue systems are still in early stages [28, 84], which facing the challenges of complex representations in open-domain, difficulties of manually constructing models and limitations of scalability of dialogue models. Nevertheless, we believe that planning could be a strong suit for dialogue systems by integrating with automatically domain models learning and searching strategies, especially for controlled dialogue generation. Moreover, we are interested in the explainability of planning-based dialogue systems, it would be interesting to know the reason for intentions generation.

7 APPLICATIONS

In this section, we introduce some reality applications based on combinations of AI planning and natural language processing. AI planning is widely used in reality management systems, such as logistics management, workshop schedule, and reservoir operation. Moreover, natural language processing (NLP) helps human communicate with agents, the combination of AI planning and NLP enables applications to be found in many fields, such as emergency managements [11, 29, 29, 33] and urban planning [63, 63, 89]. For example, the Urban Redevelopment Authority (URA) Centre in Singapore ³ deployed Robotic Process Automation and NLP to help conduct operations for resource optimization. The combination allows routine tasks to make use of AI planning and NLP, such as chatbots for public queries, which capture information from large datasets, analyze textual

³<https://www.ura.gov.sg/Corporate/Resources/Ideas-and-Trends/AI-in-Urban-Planning>

feedback, make planning decisions, and respond intelligently. On the other hand, the navigation systems in daily use, such as Baidu Maps ⁴ and Google Maps ⁵, combine making decisions and NLP techniques, which plan out routes with different objectives according to goals, and generate natural language suggestions to guide human. Moreover, agents learn from human commands and navigation datasets, helping agents understand human behaviors [45, 73].

8 CONCLUSION

In this paper, we consider that AI planning and natural language processing have strong ties, and we introduce recent works about four related tasks, i.e., planning-based text understanding, planning-based natural language processing, planning-based explainability, and text-based human-robot interaction. We first introduce backgrounds about AI planning and natural language processing and discuss commons between them, as well as their abilities to generate explicit knowledge, e.g., domain models, and learning from tacit knowledge, e.g., neural models. We then introduce methods of planning-based text understanding by extracting action sequences from texts and learning domain models from texts. Next, we give an overview of planning-based natural language processing about text generation, text summarization, and machine translation. Then, we introduce recent works in planning-based explainability and text-based human-robot interaction.

With this paper, we aim to provide a high-level view of AI planning and natural language processing for further studies, about integrating them for a combination of explicit and tacit knowledge. Combining learning from tacit knowledge and using explicit knowledge in a fully principled way is an open problem, although there are non-negligible relations between AI planning and natural language processing, allowing each of them can effectively impact the other one. However, there is not enough communication between these two fields. While many advances have been made in natural language processing by using AI planning algorithms, a significant amount of research is still required to understand the implied knowledge hidden in texts. Meanwhile, improving the ability to describe environments by domain models and solve large-scale planning problems is also beneficial to understanding texts and generating coherent and interesting texts. We believe that integrating AI planning and natural language processing, a complex combination of explicit and tacit knowledge, is a promising research area, which can improve the communication between human and intelligent agents.

REFERENCES

- [1] S Alshaina, Ansamma John, and Aneesh G Nath. 2017. Multi-document abstractive summarization based on predicate argument structure. In *2017 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*. IEEE, 1–6.
- [2] Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O. Riedl. 2021. Automated Storytelling via Causal, Commonsense Plot Ordering. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. 5859–5867. <https://ojs.aaai.org/index.php/AAAI/article/view/16733>
- [3] Reinald Kim Amplayo, Stefanos Angelidis, and Mirella Lapata. 2021. Unsupervised Opinion Summarization with Content Planning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. 12489–12497. <https://ojs.aaai.org/index.php/AAAI/article/view/17481>
- [4] Mozghan Nasr Azadani, Nasser Ghadiri, and Ensieh Davoodijam. 2018. Graph-based biomedical text summarization: An itemset mining and sentence clustering approach. *J. Biomed. Informatics* 84 (2018), 42–58. <https://doi.org/10.1016/j.jbi.2018.06.005>

⁴<https://map.baidu.com/>

⁵<https://www.google.de/maps>

- [5] Amos Azaria, Jayant Krishnamurthy, and Tom M. Mitchell. 2016. Instructable Intelligent Personal Agent. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, Dale Schuurmans and Michael P. Wellman (Eds.). 2681–2689. <http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12383>
- [6] Amos Azaria, Shashank Srivastava, Jayant Krishnamurthy, Igor Labutov, and Tom M. Mitchell. 2020. An agent for learning new natural language commands. *Auton. Agents Multi Agent Syst.* 34, 1 (2020), 6. <https://doi.org/10.1007/s10458-019-09425-x>
- [7] Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. 2017. An Actor-Critic Algorithm for Sequence Prediction. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. <https://openreview.net/forum?id=SJDaqvqev>
- [8] Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Multi-Document Abstractive Summarization Using ILP Based Multi-Sentence Compression. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, Qiang Yang and Michael J. Wooldridge (Eds.). 1208–1214. <http://ijcai.org/Abstract/15/174>
- [9] Robert Baud, Christian Lovis, Laurence Alpay, Anne-Marie Rassinaux, JR Scherrer, Anthony Nowlan, and Alan Rector. 1993. Modelling for natural language understanding.. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*. American Medical Informatics Association, 289.
- [10] Yonatan Bisk, Kevin J. Shih, Yejin Choi, and Daniel Marcu. 2018. Learning Interpretable Spatial Operations in a Rich 3D Blocks World. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). 5028–5036. <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17410>
- [11] Simon Blindheim, Sebastien Gros, and Tor Arne Johansen. 2020. Risk-based model predictive control for autonomous ship emergency management. *IFAC-PapersOnLine* 53, 2 (2020), 14524–14531.
- [12] Valts Blukis, Dipendra Kumar Misra, Ross A. Knepper, and Yoav Artzi. 2018. Mapping Navigation Instructions to Continuous Control Actions with Position-Visitation Prediction. In *2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings (Proceedings of Machine Learning Research, Vol. 87)*. 505–518. <http://proceedings.mlr.press/v87/blukis18a.html>
- [13] S. R. K. Branavan, Harr Chen, Luke S. Zettlemoyer, and Regina Barzilay. 2009. Reinforcement Learning for Mapping Instructions to Actions. In *ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore*, Keh-Yih Su, Jian Su, and Janyce Wiebe (Eds.). 82–90. <https://aclanthology.org/P09-1010/>
- [14] S. R. K. Branavan, Nate Kushman, Tao Lei, and Regina Barzilay. 2012. Learning High-Level Planning from Text. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*. 126–135. <https://aclanthology.org/P12-1014/>
- [15] Jake Brawer, Olivier Mangin, Alessandro Roncone, Sarah Widdler, and Brian Scassellati. 2018. Situated Human-Robot Collaboration: predicting intent from grounded natural language. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*. 827–833. <https://doi.org/10.1109/IROS.2018.8593942>
- [16] Igor Buzhinsky. 2019. Formalization of natural language requirements into temporal logics: a survey. In *17th IEEE International Conference on Industrial Informatics, INDIN 2019, Helsinki, Finland, July 22-25, 2019*. 400–406. <https://doi.org/10.1109/INDIN41052.2019.8972130>
- [17] Erik Cambria and Bebo White. 2014. Jumping NLP curves: A review of natural language processing research. *IEEE Computational intelligence magazine* 9, 2 (2014), 48–57.
- [18] Guy Camilleri. 2002. Dialogue systems and planning. In *International Conference on Text, Speech and Dialogue*. Springer, 429–436.
- [19] Rehj Cantrell, Paul W. Schermerhorn, and Matthias Scheutz. 2011. Learning actions from human-robot dialogues. In *20th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2011, Atlanta, Georgia, USA, July 31 - August 3, 2011*, Henrik I. Christensen (Ed.). 125–130. <https://doi.org/10.1109/ROMAN.2011.6005199>
- [20] Rehj Cantrell, Kartik Talamadupula, Paul W. Schermerhorn, J. Benton, Subbarao Kambhampati, and Matthias Scheutz. 2012. Tell me when and why to do it!: run-time planner model updates via natural language instruction. In *HRI, Holly A. Yanco, Aaron Steinfeld, Vanessa Evers, and Odest Chadwicke Jenkins (Eds.)*. 471–478. <https://doi.org/10.1145/2157689.2157840>
- [21] Nick Cercone and Gordon McCalla. 1986. Accessing knowledge through natural language. In *Advances in Computers*. Vol. 25. Elsevier, 1–99.
- [22] David L. Chen and Raymond J. Mooney. 2011. Learning to Interpret Natural Language Navigation Instructions from Observations. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco,*

- California, USA, August 7-11, 2011, Wolfram Burgard and Dan Roth (Eds.). <http://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3701>
- [23] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A Survey on Dialogue Systems: Recent Advances and New Frontiers. *SIGKDD Explor.* 19, 2 (2017), 25–35. <https://doi.org/10.1145/3166054.3166058>
 - [24] Howard Chen, Alane Suhr, Dipendra Misra, Noah Snaveley, and Yoav Artzi. 2019. TOUCHDOWN: Natural Language Navigation and Spatial Reasoning in Visual Street Environments. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. 12538–12547. http://openaccess.thecvf.com/content_CVPR_2019/html/Chen_TOUCHDOWN_Natural_Language_Navigation_and_Spatial_Reasoning_in_Visual_Street_CVPR_2019_paper.html
 - [25] Haonan Chen, Hao Tan, Alan Kuntz, Mohit Bansal, and Ron Alterovitz. 2020. Enabling Robots to Understand Incomplete Natural Language Instructions Using Commonsense Reasoning. In *2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020*. 1963–1969. <https://doi.org/10.1109/ICRA40945.2020.9197315>
 - [26] Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-Tür. 2020. Just Ask: An Interactive Learning Framework for Vision and Language Navigation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. 2459–2466. <https://ojs.aaai.org/index.php/AAAI/article/view/5627>
 - [27] Jennifer Chu-Carroll and Sandra Carberry. 1994. A plan-based model for response generation in collaborative task-oriented dialogues. *arXiv preprint cmp-lg/9405011* (1994).
 - [28] Philip R Cohen. 2020. Back to the future for dialogue research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 13514–13519.
 - [29] Daniel G Costa, João Paulo J Peixoto, Thiago C Jesus, Paulo Portugal, Francisco Vasques, Elivelton Rangel, and Maycon Peixoto. 2022. A Survey of Emergencies Management Systems in Smart Cities. *IEEE Access* (2022).
 - [30] Chenhe Dong, Yinghui Li, Haifan Gong, Miaoxin Chen, Junxin Li, Ying Shen, and Min Yang. 2021. A Survey of Natural Language Generation. *arXiv preprint arXiv:2112.11739* (2021).
 - [31] Angela Fan, Mike Lewis, and Yann N. Dauphin. 2019. Strategies for Structuring Story Generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). 2650–2660. <https://doi.org/10.18653/v1/p19-1254>
 - [32] Wenfeng Feng, Hankz Hankui Zhuo, and Subbarao Kambhampati. 2018. Extracting Action Sequences from Texts Based on Deep Reinforcement Learning. In *IJCAI*. 4064–4070.
 - [33] Daniela Fogli and Giovanni Guida. 2013. Knowledge-centered design of decision support systems for emergency management. *Decision Support Systems* 55, 1 (2013), 336–347.
 - [34] Maria Fox and Derek Long. 2002. PDDL+: Modeling continuous time dependent effects. In *Proceedings of the 3rd International NASA Workshop on Planning and Scheduling for Space*, Vol. 4. 34.
 - [35] Tyler M. Frasca, Bradley Oosterveld, Meia Chita-Tegmark, and Matthias Scheutz. 2021. Enabling Fast Instruction-Based Modification of Learned Robot Skills. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. 6075–6083. <https://ojs.aaai.org/index.php/AAAI/article/view/16757>
 - [36] Konstantina Garoufi. 2014. Planning-Based Models of Natural Language Generation. *Lang. Linguistics Compass* 8, 1 (2014), 1–10. <https://doi.org/10.1111/lnc3.12053>
 - [37] Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research* 61 (2018), 65–170.
 - [38] Christopher W. Geib and Mark Steedman. 2007. On Natural Language Processing and Plan Recognition. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, Manuela M. Veloso (Ed.). 1612–1617. <http://ijcai.org/Proceedings/07/Papers/260.pdf>
 - [39] Alfonso Gerevini and Derek Long. 2005. *Plan constraints and preferences in PDDL3*. Technical Report. Technical Report 2005-08-07, Department of Electronics for Automation . . .
 - [40] Lin Guan, Mudit Verma, Sihang Guo, Ruohan Zhang, and Subbarao Kambhampati. 2021. Widening the Pipeline in Human-Guided Reinforcement Learning with Explanation and Context-Aware Data Augmentation. *Advances in Neural Information Processing Systems* 34 (2021).
 - [41] Çağlar Gülçehre, Francis Dutil, Adam Trischler, and Yoshua Bengio. 2017. Plan, Attend, Generate: Character-Level Neural Machine Translation with Planning. In *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017*, Phil Blunsom, Antoine Bordes, Kyunghyun Cho, Shay B. Cohen, Chris Dyer, Edward Grefenstette, Karl Moritz Hermann, Laura Rimell, Jason Weston, and Scott Yih (Eds.).

- 228–234. <https://doi.org/10.18653/v1/w17-2627>
- [42] Thomas Hayton, Julie Porteous, João Fernando Ferreira, and Alan Lindsay. 2020. Narrative Planning Model Acquisition from Text Summaries and Descriptions. In *AAAI* 1709–1716.
 - [43] Daniel Hládek, Ján Staš, and Matúš Pleva. 2020. Survey of automatic spelling correction. *Electronics* 9, 10 (2020), 1670.
 - [44] Xinyu Hua, Ashwin Sreevatsa, and Lu Wang. 2021. DYPLOC: Dynamic Planning of Content Using Mixed Language Models for Text Generation. In *Proceedings of ACL*. 6408–6423.
 - [45] Jizhou Huang, Haifeng Wang, Miao Fan, An Zhuo, Yibo Sun, and Ying Li. 2020. Understanding the Impact of the COVID-19 Pandemic on Transportation-related Behaviors with Human Mobility Data. (2020), 3443–3450. <https://doi.org/10.1145/3394486.3412856>
 - [46] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. 2022. Language Models as Zero-Shot Planners: Extracting Actionable Knowledge for Embodied Agents. 162 (2022), 9118–9147. <https://proceedings.mlr.press/v162/huang22a.html>
 - [47] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. 2022. Inner Monologue: Embodied Reasoning through Planning with Language Models. *CoRR* abs/2207.05608 (2022). <https://doi.org/10.48550/arXiv.2207.05608> arXiv:2207.05608
 - [48] Pham Ngoc Hung and Takashi Yoshimi. 2016. Extracting actions from instruction manual and testing their execution in a robotic simulation. *ASEAN Engineering Journal* 6, 1 (2016), 47–58.
 - [49] Touseef Iqbal and Shaima Qureshi. 2020. The survey: Text generation models in deep learning. *Journal of King Saud University-Computer and Information Sciences* (2020).
 - [50] Peter A. Jansen. 2020. Visually-Grounded Planning without Vision: Language Models Infer Detailed Plans from High-level Instructions. *EMNLP 2020* (2020), 4412–4417. <https://doi.org/10.18653/v1/2020.findings-emnlp.395>
 - [51] HanQi Jin, Yue Cao, TianMing Wang, XinYu Xing, and XiaoJun Wan. 2020. Recent advances of neural text generation: Core tasks, datasets, models and challenges. *Science China Technological Sciences* 63, 10 (2020), 1990–2010.
 - [52] Subbarao Kambhampati. 2021. Polanyi’s revenge and AI’s new romance with tacit knowledge. *Commun. ACM* 64, 2 (2021), 31–32. <https://doi.org/10.1145/3446369>
 - [53] Subbarao Kambhampati, Sarath Sreedharan, Mudit Verma, Yantian Zha, and Lin Guan. 2021. Symbols as a Lingua Franca for Bridging Human-AI Chasm for Explainable and Advisable AI Systems. *CoRR* abs/2109.09904 (2021). arXiv:2109.09904 <https://arxiv.org/abs/2109.09904>
 - [54] Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. 2022. Natural language processing: State of the art, current trends and challenges. *Multimedia Tools and Applications* (2022), 1–32.
 - [55] Joohyun Kim and Raymond J. Mooney. 2012. Unsupervised PCFG Induction for Grounded Language Learning with Highly Ambiguous Supervision. In *EMNLP-CoNLL*, Jun’ichi Tsujii, James Henderson, and Marius Pasca (Eds.). 433–444. <https://aclanthology.org/D12-1040/>
 - [56] Alexander Koller and Jörg Hoffmann. 2010. Waking Up a Sleeping Rabbit: On Natural-Language Sentence Generation with FF. In *Proceedings of the 20th International Conference on Automated Planning and Scheduling, ICAPS 2010, Toronto, Ontario, Canada, May 12-16, 2010*, Ronen I. Brafman, Hector Geffner, Jörg Hoffmann, and Henry A. Kautz (Eds.). AAAI, 238–241. <http://www.aaai.org/ocs/index.php/ICAPS/ICAPS10/paper/view/1415>
 - [57] Xiangzhe Kong, Jialiang Huang, Ziquan Tung, Jian Guan, and Minlie Huang. 2021. Stylized Story Generation with Style-Guided Planning. In *ACL-IJCNLP*. 2430–2436.
 - [58] Litton J Kurisinkel, Yue Zhang, and Vasudeva Varma. 2017. Abstractive multi-document summarization by partial tree extraction, recombination and linearization. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 812–821.
 - [59] Anastassia Küstenmacher and Paul G Plöger. 2021. Improving the Reliability of Service Robots by Symbolic Representation of Execution Specific Knowledge. In *Robust and Reliable Autonomy in the Wild (R2AW)*.
 - [60] Boyang Li, Stephen Lee-Urban, George Johnston, and Mark Riedl. 2013. Story Generation with Crowdsourced Plot Graphs. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, July 14-18, 2013, Bellevue, Washington, USA*, Marie desJardins and Michael L. Littman (Eds.). <http://www.aaai.org/ocs/index.php/AAAI/AAAI13/paper/view/6399>
 - [61] Shuang Li, Xavier Puig, Chris Paxton, Yilun Du, Clinton Wang, Linxi Fan, Tao Chen, De-An Huang, Ekin Akyürek, Anima Anandkumar, Jacob Andreas, Igor Mordatch, Antonio Torralba, and Yuke Zhu. 2022. Pre-Trained Language Models for Interactive Decision-Making. *CoRR* abs/2202.01771 (2022). arXiv:2202.01771 <https://arxiv.org/abs/2202.01771>
 - [62] Zhuohan Li, Siyuan Zhuang, Shiyuan Guo, Danyang Zhuo, Hao Zhang, Dawn Song, and Ion Stoica. 2021. Terapipe: Token-level pipeline parallelism for training large-scale language models. In *International Conference on Machine Learning*. PMLR, 6543–6552.

- [63] Claudius Lieven, Bianca Lüders, Daniel Kulus, and Rosa Thoneick. 2021. Enabling digital co-creation in urban planning and development. (2021), 415–430.
- [64] Alan Lindsay, Jonathon Read, João F. Ferreira, Thomas Hayton, Julie Porteous, and Peter Gregory. 2017. Framers: Planning Models from Natural Language Action Descriptions. In *ICAPS*. 434–442.
- [65] Adam Lopez. 2008. Statistical Machine Translation. *ACM Comput. Surv.* 40, 3, Article 8 (aug 2008), 49 pages. <https://doi.org/10.1145/1380584.1380586>
- [66] Stephanie M Lukin and Marilyn A Walker. 2019. A narrative sentence planner and structurer for domain independent, parameterizable storytelling. *Dialogue & Discourse* 10, 1 (2019), 34–86.
- [67] Longxuan Ma, Mingda Li, Wei-Nan Zhang, Jiapeng Li, and Ting Liu. 2022. Unstructured Text Enhanced Open-Domain Dialogue System: A Systematic Survey. *ACM Trans. Inf. Syst.* 40, 1 (2022), 9:1–9:44. <https://doi.org/10.1145/3464377>
- [68] Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. 2006. Walk the Talk: Connecting Language, Knowledge, and Action in Route Instructions. In *AAAI*. 1475–1482. <http://www.aaai.org/Library/AAAI/2006/aaai06-232.php>
- [69] Andreas Marfurt and James Henderson. 2021. Sentence-level Planning for Especially Abstractive Summarization. In *Proceedings of the Third Workshop on New Frontiers in Summarization*. 1–14.
- [70] Cynthia Matuszek, Dieter Fox, and Karl Koscher. 2010. Following directions using statistical machine translation. In *HRI*, Pamela J. Hinds, Hiroshi Ishiguro, Takayuki Kanda, and Peter H. Kahn Jr. (Eds.). 251–258. <https://doi.org/10.1145/1734454.1734552>
- [71] Drew McDermott, Malik Ghallab, Adele E. Howe, Craig A. Knoblock, Ashwin Ram, Manuela M. Veloso, Daniel S. Weld, and David E. Wilkins. 1998. PDDL-the planning domain definition language.
- [72] Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2016. Listen, Attend, and Walk: Neural Mapping of Navigational Instructions to Action Sequences. In *Thirtieth AAAI Conference on Artificial Intelligence*, Dale Schuurmans and Michael P. Wellman (Eds.). 2772–2778. <http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12522>
- [73] Shah Jahan Miah, Huy Quan Vu, and Daminda Alahakoon. 2022. A social media analytics perspective for human-oriented smart city planning and management. *Journal of the Association for Information Science and Technology* 73, 1 (2022), 119–135.
- [74] M Jishma Mohan, C Sunitha, Amal Ganesh, and A Jaya. 2016. A study on ontology based abstractive summarization. *Procedia Computer Science* 87 (2016), 32–37.
- [75] Shiwali Mohan and John E. Laird. 2014. Learning Goal-Oriented Hierarchical Tasks from Situated Interactive Instruction. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27–31, 2014, Québec City, Québec, Canada*, Carla E. Brodley and Peter Stone (Eds.). 387–394. <http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8630>
- [76] Christian Muise, Tathagata Chakraborti, Shubham Agarwal, Ondrej Bajgar, Arunima Chaudhary, Luis A Lastras-Montano, Josef Ondrej, Miroslav Vodolan, and Charlie Wiecha. 2019. Planning for goal-oriented dialogue systems. *arXiv preprint arXiv:1910.08137* (2019).
- [77] Shashi Narayan, Yao Zhao, Joshua Maynez, Gonalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. *Transactions of the Association for Computational Linguistics* 9 (2021), 1475–1492.
- [78] Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Patwary, Vijay Korthikanti, Dmitri Vainbrand, Prethvi Kashinkunti, Julie Bernauer, Bryan Catanzaro, et al. 2021. Efficient large-scale language model training on gpu clusters using megatron-lm. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 1–15.
- [79] Daniel Nyga and Michael Beetz. 2012. Everything robots always wanted to know about housework (but were afraid to ask). In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 243–250.
- [80] MD Okpor. 2014. Machine translation approaches: issues and challenges. *International Journal of Computer Science Issues (IJCSI)* 11, 5 (2014), 159.
- [81] Daniel W Otter, Julian R Medina, and Jugul K Kalita. 2020. A survey of the usages of deep learning for natural language processing. *IEEE transactions on neural networks and learning systems* 32, 2 (2020), 604–624.
- [82] Vishal Pallagani and Biplav Srivastava. 2021. A Generic Dialog Agent for Information Retrieval Based on Automated Planning Within a Reinforcement Learning Platform. *Bridging the Gap Between AI Planning and Reinforcement Learning (PRL)* (2021).
- [83] C. Raymond Perrault and James F. Allen. 1980. A Plan-Based Analysis of Indirect Speech Acts. *Am. J. Comput. Linguistics* 6, 3-4 (1980), 167–182.
- [84] Ronald PA Petrick and Mary Ellen Foster. 2016. Using general-purpose planning for action selection in human-robot interaction. In *2016 AAAI Fall Symposium Series*.
- [85] Ngoc Hung Pham and Takashi Yoshimi. 2015. Extraction of actions and objects from instruction manual for executable robot planning. In *2015 15th International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 881–885.

- [86] Julie Porteous and Marc Cavazza. 2009. Controlling Narrative Generation with Planning Trajectories: The Role of Constraints. In *ICIDS 2009 (Lecture Notes in Computer Science, Vol. 5915)*, Ido Iurgel, Nelson Zagalo, and Paolo Petta (Eds.), 234–245. https://doi.org/10.1007/978-3-642-10643-9_28
- [87] Julie Porteous, João F. Ferreira, Alan Lindsay, and Marc Cavazza. 2021. Automated narrative planning model extension. *Auton. Agents Multi Agent Syst.* 35, 2 (2021), 19. <https://doi.org/10.1007/s10458-021-09501-1>
- [88] Julie Porteous, João F Ferreira, Alan Lindsay, and Marc Cavazza. 2021. Automated narrative planning model extension. *Autonomous Agents and Multi-Agent Systems* 35, 2 (2021), 1–29.
- [89] Bernd Resch, Anja Summa, Peter Zeile, and Michael Strube. 2016. Citizen-centric urban planning through extracting emotion information from twitter in an interdisciplinary space-time-linguistics algorithm. *Urban Planning* 1, 2 (2016), 114–127.
- [90] Mark O. Riedl and Robert Michael Young. 2010. Narrative Planning: Balancing Plot and Character. *J. Artif. Intell. Res.* 39 (2010), 217–268. <https://doi.org/10.1613/jair.2989>
- [91] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic keyword extraction from individual documents. *Text mining: applications and theory* 1 (2010), 1–20.
- [92] Scott Sanner et al. 2010. Relational dynamic influence diagram language (rddl): Language description. *Unpublished ms. Australian National University* 32 (2010), 27.
- [93] Milene Santos Teixeira and Mauro Dragoni. 2022. A Review of Plan-Based Approaches for Dialogue Management. *Cognitive Computation* (2022), 1–20.
- [94] Buser Say. 2021. A Unified Framework for Planning with Learned Neural Network Transition Models. (2021), 5016–5024. <https://ojs.aaai.org/index.php/AAAI/article/view/16635>
- [95] Matthias Scheutz, Evan A. Krause, Bradley Oosterveld, Tyler M. Frasca, and Robert Platt Jr. 2017. Spoken Instruction-Based One-Shot Object and Action Learning in a Cognitive Robotic Architecture. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017*, Kate Larson, Michael Winikoff, Sanmay Das, and Edmund H. Durfee (Eds.), 1378–1386. <http://dl.acm.org/citation.cfm?id=3091315>
- [96] Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-Planning Neural Text Generation From Structured Data. In *AAAI, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.)*, 5414–5421.
- [97] Khaled Shaalan et al. 2010. Rule-based approach in Arabic natural language processing. *The International Journal on Information and Communication Technologies (IJICT)* 3, 3 (2010), 11–19.
- [98] Pratyusha Sharma, Antonio Torralba, and Jacob Andreas. 2022. Skill Induction and Planning with Latent Language. (2022), 1713–1726. <https://doi.org/10.18653/v1/2022.acl-long.120>
- [99] Lanbo She and Joyce Yue Chai. 2017. Interactive Learning of Grounded Verb Semantics towards Human-Robot Communication. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, Regina Barzilay and Min-Yen Kan (Eds.), 1634–1644. <https://doi.org/10.18653/v1/P17-1150>
- [100] Lanbo She, Yu Cheng, Joyce Yue Chai, Yunyi Jia, Shaohua Yang, and Ning Xi. 2014. Teaching Robots New Actions through Natural Language Instructions. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication, IEEE RO-MAN 2014, Edinburgh, UK, August 25-29, 2014*, 868–873. <https://doi.org/10.1109/ROMAN.2014.6926362>
- [101] Lanbo She, Shaohua Yang, Yu Cheng, Yunyi Jia, Joyce Yue Chai, and Ning Xi. 2014. Back to the Blocks World: Learning New Actions through Situated Human-Robot Dialogue. In *Proceedings of the SIGDIAL 2014 Conference, The 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 18-20 June 2014, Philadelphia, PA, USA*, 89–97. <https://doi.org/10.3115/v1/w14-4313>
- [102] Raphael Shu and Hideki Nakayama. 2018. Discrete Structural Planning for Neural Machine Translation. *CoRR abs/1808.04525* (2018). arXiv:1808.04525 <http://arxiv.org/abs/1808.04525>
- [103] Avirup Sil and Alexander Yates. 2011. Extracting STRIPS Representations of Actions and Events. In *RANLP*, 1–8.
- [104] Nisha Simon and Christian Muise. 2022. TattleTale: Storytelling with Planning and Large Language Models. (2022).
- [105] Sarath Sreedharan, Tathagata Chakraborti, Christian Muise, Yasaman Khazaeni, and Subbarao Kambhampati. 2020. -d3wa+-a case study of xaip in a model acquisition task for dialogue planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 30, 488–497.
- [106] Felix Stahlberg. 2020. Neural Machine Translation: A Review. *J. Artif. Intell. Res.* 69 (2020), 343–418. <https://doi.org/10.1613/jair.1.12007>
- [107] Shane Storks, Qiaozhi Gao, and Joyce Y Chai. 2019. Commonsense reasoning for natural language understanding: A survey of benchmarks, resources, and approaches. *arXiv preprint arXiv:1904.01172* (2019), 1–60.
- [108] Gavin Suddrey, Ben Talbot, and Frederic Maire. 2022. Learning and executing re-usable behaviour trees from natural language instruction. *IEEE Robotics and Automation Letters* (2022).
- [109] Pradyumna Tambwekar, Murtaza Dhuliawala, Lara J. Martin, Animesh Mehta, Brent Harrison, and Mark O. Riedl. 2019. Controllable Neural Story Plot Generation via Reward Shaping. In *Proceedings of the Twenty-Eighth International*

- Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, Sarit Kraus (Ed.). 5982–5988. <https://doi.org/10.24963/ijcai.2019/829>
- [110] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R. Walter, Ashis Gopal Banerjee, Seth J. Teller, and Nicholas Roy. 2011. Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation. In *AAAI*, Wolfram Burgard and Dan Roth (Eds.). <http://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3623>
 - [111] Moritz Tenorth, Daniel Nyga, and Michael Beetz. 2010. Understanding and executing instructions for everyday manipulation tasks from the World Wide Web. In *IEEE International Conference on Robotics and Automation, ICRA 2010, Anchorage, Alaska, USA, 3-7 May 2010*. 1486–1491. <https://doi.org/10.1109/ROBOT.2010.5509955>
 - [112] Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. 2019. Vision-and-Dialog Navigation. In *3rd Annual Conference on Robot Learning, CoRL 2019, Osaka, Japan, October 30 - November 1, 2019, Proceedings (Proceedings of Machine Learning Research, Vol. 100)*, Leslie Pack Kaelbling, Danica Kragic, and Komei Sugiura (Eds.). 394–406. <http://proceedings.mlr.press/v100/thomason20a.html>
 - [113] Jesse Thomason, Shiqi Zhang, Raymond J. Mooney, and Peter Stone. 2015. Learning to Interpret Natural Language Commands through Human-Robot Dialog. In *IJCAI 2015*, Qiang Yang and Michael J. Wooldridge (Eds.). 1923–1929. <http://ijcai.org/Abstract/15/273>
 - [114] AmirSina Torfi, Rouzbeh A Shirvani, Yaser Keneshloo, Nader Tavaf, and Edward A Fox. 2020. Natural language processing advancements by deep learning: A survey. *arXiv preprint arXiv:2003.01200* (2020).
 - [115] Stephen G. Ware and Robert Michael Young. 2011. CPOCL: A Narrative Planner Supporting Conflict. In *AIIDE*, Vadim Bulitko and Mark O. Riedl (Eds.). <http://www.aaai.org/ocs/index.php/AIIDE/AIIDE11/paper/view/4058>
 - [116] Robert Wilensky. 1981. Meta-Planning: Representing and Using Knowledge About Planning in Problem Solving and Natural Language Understanding. *Cogn. Sci.* 5, 3 (1981), 197–233. https://doi.org/10.1207/s15516709cog0503_2
 - [117] Chungheng Xiang, Tingsong Jiang, Baobao Chang, and Zhifang Sui. 2015. ERSOM: A Structural Ontology Matching Approach Using Automatically Learned Entity Representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton (Eds.). 2419–2429. <https://doi.org/10.18653/v1/d15-1289>
 - [118] Jingjing Xu, Xuancheng Ren, Yi Zhang, Qi Zeng, Xiaoyan Cai, and Xu Sun. 2018. A Skeleton-Based Model for Promoting Coherence Among Sentences in Narrative Story Generation. In *EMNLP*. 4306–4315. <https://aclanthology.org/D18-1462/>
 - [119] Lili Yao, Nanyun Peng, Ralph M. Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-Write: Towards Better Automatic Storytelling. In *AAAI*. 7378–7385.
 - [120] Rongguang Ye, Qingchuan Xu, Jie Liu, Yang Hong, Chengfeng Sun, Wenzheng Chi, and Lining Sun. 2021. A Natural Language Instruction Disambiguation Method for Robot Grasping. In *IEEE International Conference on Robotics and Biomimetics, ROBIO 2021, Sanya, China, December 27-31, 2021*. 601–606. <https://doi.org/10.1109/ROBIO54168.2021.9739456>
 - [121] Kristina Y. Yordanova and Thomas Kirste. 2016. Learning Models of Human Behaviour from Textual Instructions. In *ICAART*. 415–422.
 - [122] R Michael Young, Stephen G Ware, Brad A Cassell, and Justus Robertson. 2013. Plans and planning in narrative generation: a review of plan-based approaches to the generation of story, discourse and interactivity in narratives. *Sprache und Datenverarbeitung, Special Issue on Formal and Computational Models of Narrative* 37, 1-2 (2013), 41–64.
 - [123] Meng-Hsuan Yu, Juntao Li, Zhangming Chan, Rui Yan, and Dongyan Zhao. 2021. Content Learning with Structure-Aware Writing: A Graph-Infused Dual Conditional Variational Autoencoder for Automatic Storytelling. In *AAAI*. 6021–6029.
 - [124] Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2022. A survey of knowledge-enhanced text generation. *ACM Computing Surveys (CSUR)* (2022).
 - [125] Yantian Zha, Lin Guan, and Subbarao Kambhampati. 2021. Learning from Ambiguous Demonstrations with Self-Explanation Guided Reinforcement Learning. *CoRR* abs/2110.05286 (2021). arXiv:2110.05286 <https://arxiv.org/abs/2110.05286>
 - [126] Zhengyan Zhang, Yuxian Gu, Xu Han, Shengqi Chen, Chaojun Xiao, Zhenbo Sun, Yuan Yao, Fanchao Qi, Jian Guan, Pei Ke, et al. 2021. Cpm-2: Large-scale cost-effective pre-trained language models. *AI Open* 2 (2021), 216–224.
 - [127] Fengda Zhao, Zhikai Yang, Xianshan Li, Dingding Guo, and Haitao Li. 2021. Extract Executable Action Sequences from Natural Language Instructions Based on DQN for Medical Service Robots. *Int. J. Comput. Commun. Control* 16, 2 (2021). <https://doi.org/10.15837/ijccc.2021.2.4115>
 - [128] Hankz Hankui Zhuo and Subbarao Kambhampati. 2013. Action-Model Acquisition from Noisy Plan Traces. In *IJCAI*. 2444–2450.
 - [129] Hankz Hankui Zhuo and Subbarao Kambhampati. 2017. Model-lite planning: Case-based vs. model-based approaches. *Artif. Intell.* 246 (2017), 1–21. <https://doi.org/10.1016/j.artint.2017.01.004>

- [130] Hankz Hankui Zhuo, Héctor Muñoz-Avila, and Qiang Yang. 2014. Learning hierarchical task network domains from partially observed plan traces. *Artif. Intell.* 212 (2014), 134–157.
- [131] Hankz Hankui Zhuo and Qiang Yang. 2014. Action-model acquisition for planning via transfer learning. *Artif. Intell.* 212 (2014), 80–103. <https://doi.org/10.1016/j.artint.2014.03.004>
- [132] Hankz Hankui Zhuo, Yantian Zha, Subbarao Kambhampati, and Xin Tian. 2020. Discovering Underlying Plans Based on Shallow Models. *ACM Trans. Intell. Syst. Technol.* 11, 2 (2020), 18:1–18:30. <https://doi.org/10.1145/3368270>

Received xxxx; revised xxxx; accepted xxxx