ATPT: Automate Typhoon Contingency Plan Generation from Text

(Demonstration Track)

Yifeng Zeng[1], Zhangrui Yao[2], Yinghui Pan[3*], Wanqing Chen[3], Junxin Zhou[3], Junhan Chen[2], Biyang Ma[1] and Zhong Ming[3]
[1]Department of Computer and Information Sciences, Northumbria University, UK
[2]Department of Automation, Xiamen University, China
[3]College of Computer Science and Software Engineering, Shenzhen University, China
Email:panyinghui@szu.edu.cn(*CorrespondingAuthor)

ABSTRACT

Artificial intelligence (AI) planning models play an important role in decision support systems for disaster management e.g. typhoon contingency plan development. However, constructing an AI planning model always requires significant amount of manual effort, which becomes a bottleneck to emergency response in a time-critical situation. In this demonstration, we present a framework of automating a domain model of planning domain definition language from natural language input through deep learning techniques. We implement this framework in a typhoon response system and demonstrate automatic generation of typhoon contingency plan from official typhoon plan documents.

KEYWORDS

Planning Domain Definition Language; Natural Language Process; Typhoon Contingency Plan

ACM Reference Format:

Yifeng Zeng[1], Zhangrui Yao[2], Yinghui Pan[3*], Wanqing Chen[3], Junxin Zhou[3], Junhan Chen[2], Biyang Ma[1] and Zhong Ming[3]. 2021. ATPT: Automate Typhoon Contingency Plan Generation from Text: (Demonstration Track). In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

Typhoon response systems demand time-critical decision making, which expects a planning model to be implemented in an emergency mode. In particular, many inputs to build an AI planning model in order to generate a situation-aware typhoon contingency plan are presented in the form of text e.g. typhoon response manuals, a situational report or even a collection of messages from social media platforms. This needs an intelligent agent that is able to read natural language inputs and build an AI planning model accordingly. It challenges a fundamental research issue of automating an AI planning model, most of which focuses on learning the planning model from available data [13].

In this research application, we choose the classical AI planning model - Planning Domain Definition Language (PDDL) [4, 9], which has been well studied and enjoys wide applications in many fields [6], as the typhoon contingency planning model in the demonstration. Another benefit of using PDDL arises from its clear semantics that facilitates the learning process particularly involving a free text. A PDDL plan model contains two basic files: a domain file and a problem file. A problem file specifies a plan goal, initial states and a collection of objects involving in a planning problem, and it reflects decision makers' needs in the plan. In contrast, a domain file encodes important properties of the planning models, namely *Predicates* and *Actions*, that compose the planning dynamics under uncertainty. In principle, the domain file is the key element of a PDDL model and decides the planning quality when the PDDL model is executed. The domain file specification requires a large amount of domain knowledge and can not be built by a general user in a straightforward way. Hence, the automatic construction of a PDDL model lies in learning both *Predicates* and *Actions* in a domain file, namely domain model learning, from domain inputs.

There has seen growing interest in building domain models of PDDL from natural language inputs [7, 12, 14]. The model learning is either conducted through the standard natural language processing (NLP) techniques [2], e.g. OpenNLP [3] and Stanford CoreNLP [8], or developed using a deep reinforcement learning method [15]. However, most of the previous work still demands to annotate the inputs of the natural language sentences and treats the sentences in a separated way. It leads to more duplicate actions than what are needed in a PDDL domain model. In this research, we consider a more natural input where a text document or a series of messages are presented to an autonomous planning system. This is a general setting in a typhoon response system since a typhoon contingency plan development mainly depends on both official documents prescribing response strategies and a steam of live information in an emergency situation.

2 FROM TEXT TO PLAN MODELS

We are building an autonomous typhoon response system (ATRS) for the purpose of generating typhoon tracking charts and developing typhoon contingency plans in practice. The backbone is an operational planning model that can be constructed from available information including data and text inputs. In this demonstration, we show the PDDL domain model learning from text.

We present the end-to-end PDDL planning model development framework in Fig. 1. This framework aims to generate both predicates and actions for a domain model in PDDL. Instead of dealing

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

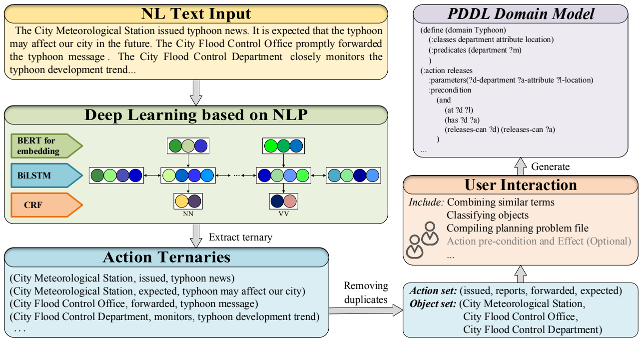

Figure 1: A deep learning based framework for automating a PDDL domain model.

with the inputs of individual sentences, we receive an entire paragraph of text and convert the action extraction into a sequence labelling problem. To identify a set of correct words (*Predicates* and *Actions*), we resort to deep learning based NLP techniques, e.g. bidirectional encoder representations from transformers (BERT) [1], to process the inputs and use a bidirectional long-short term memory-conditional random field (BiLSTM- CRF) model [11] to learn an action representation with a triplet (*subject, predicate, object*) from the processed text. The BERT model functions as a segmentation mechanism to divide the input paragraphs and subsequently transforms the sentences and associated words into a low-dimension semantic representation in a hidden feature space. The BiLSTM-CRF model deals with the sequence labelling problem and identifies a proper set of predicates and actions from the vectors that embed the features of the paragraphs. In our text, we add the constraints of typhoon states and relevant control strategies in the CRF model so that it could learn the triplets in a more accurate manner.

As the learning process may still mis-understand some terminologies in the typhoon context, we implement a user interaction component where users have chance to refine the learned triplets. The refined triplets become the inputs to learning PDDL domain models. By using the triplets, the system fills in the domain models following the PDDL structure. In addition, a problem file is specified in the user interaction and is compiled together with the domain model to build a PDDL model. By doing this, we can develop a typhoon contingency plan based on the learned PDDL model. As we use pre-trained BERT models, the actual execution of building the PDDL model is rather efficient. In addition, we could transfer the knowledge in the built PDDL model to cope with a typhoon contingency plan in different cities. It builds up a timely and consistent response in ATRS.

3 AUTONOMOUS TYPHOON RESPONSE SYSTEM

We implement a web-based autonomous typhoon response system (ATRS) that has the main functionalities of generating a typhoon contingency plan, plotting typhoon tracks and maintaining a knowledge graph of typhoon response systems in Fig. 2. Potential end-users include decision makers for typhoon emergency response, academic researchers in disaster management and the general public.

The system first asks for a text document to be uploaded and then processes the document to retrieve the corresponding triplets using the deep learning framework. The identified triplets are presented to users who could add more fields into the list. Subsequently, the users can upload a problem file that joins with the learned domain file to compose the final PDDL plan model. The model is to be downloaded and run by some well developed PDDL solvers [10], which provides a typhoon contingency plan to the users.

In addition, we provide the comparison between our techniques and the latest domain model learning methods that is used for the purpose of generating a storyline [5]. The users can choose either of them to learn the domain model and expect similar interactions to refine the learned outputs. In the current format, our new framework has better performance than others, and can deal with general text inputs. We intend to accommodate more PDDL model learning techniques in the future, and improve all the ATRS functionalities to publish the external link for a general use. The link of our demonstration video could be found below [16]. We add elaborative text at the bottom of every screen in the video.

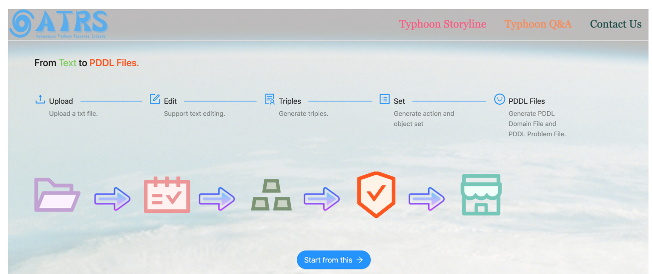

Figure 2: Web-based autonomous typhoon response system (ATRS) provides an automatic generation of typhoon contingency plan through the PDDL model learning framework.

4 CONCLUSION

Our new learning framework makes a further step of widening the type of inputs for learning the PDDL model when a text document is provided to a typhoon response system. We have implemented it to generate a proper typhoon contingency plan in the autonomous typhoon response system. This work contributes into an automatic generation of planning models for intelligent agent planning and scheduling in intelligent systems. Learning PDDL domain models still faces many challenges and user interaction is needed to remedy the learning process in order to generate a precise PDDL model. We are continuously improving the PDDL model learning and providing more friendly user interactions in ATRS.

ACKNOWLEDGMENTS

Professor Yifeng Zeng and Dr. Biyang Ma would like to thank the supports from the EPSRC project (Grant No. EP/S011609/1) in the UK. Others are supported in part by the National Natural Science Foundation of China (Grants No.61772442, 61836005 and 61806089).

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[2] Hankz Hankui Zhuo Feng, Wenfeng and Subbarao Kambhampati. 2018. Extracting Action Sequences from Texts based on Deep Reinforcement Learning. *arXiv preprint arXiv:1803.02632* (2018).

[3] Apache Software Foundation. 2004. OpenNLP. website. https://opennlp.apache.org/docs/

[4] Maria Fox and Derek Long. 2003. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *Journal of Artificial Intelligence Research* 20 (2003), 61–124.

[5] Thomas Hayton, Julie Porteous, Joao Ferreira, Alan Lindsay, and Jonathon Read. 2017. Storyframer: From input stories to output planning models. In *Workshop on Knowledge Engineering for Planning and Scheduling. The 27th International Conference on Automated Planning and Scheduling*. 1–9.

[6] Jorg Hoffmann, Holger Hermanns, Michaela Klauck, Marcel Steinmetz, Erez Karpas, and Daniele Magazzeni. 2020. let's Learn Their Language? A Case for Planning with Automata-Network Languages from Model Checking. 34 (2020), 13569–13575.

[7] Yongfeng Huo, Jing Tang, Yinghui Pan, Yifeng Zeng, and Langcai Cao. 2020. Learning a Planning Domain Model From Natural Language Process Manuals. *IEEE Access* 8 (2020), 143219–143232.

[8] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*. 55–60. http://www.aclweb.org/anthology/P/P14/P14-5010

[9] Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins. 1998. PDDL - The Planning Domain Definition Language. (1998). http://www.cs.yale.edu/homes/dvm.

[10] Christian Muise, Hanbin Li, Qianyu Zhang, Mau Magnaguagno, Nir Lipovetzky, Florian GeiÃ§er, Steve Levine, and JÃűrg Hoffmann. 2015. PDDL editor. website. http://editor.planning.domains

[11] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* (2018).

[12] Julie Porteous, João F Ferreira, Alan Lindsay, and Marc Cavazza. 2020. Extending narrative planning domains with linguistic resources. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems.

[13] Adithya Raam Sankar, Prashant Doshi, and Adam Goodie. 2020. Evacuate or Not? A POMDP Model of the Decision Making of Individuals in Hurricane Evacuation Zones. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference (Proceedings of Machine Learning Research, Vol. 115)*, Ryan P. Adams and Vibhav Gogate (Eds.). 669–678.

[14] Miglani Shivam. 2020. NL to PDDL: One-Shot Learning of Planning Domains from Natural Language Process Manuals. *Delft University of Technology Repositories* (2020).

[15] Avirup Sil, Fei Huang, and Alexander Yates. 2010. Extracting action and event semantics from web text. In *2010 AAAI Fall Symposium Series*.

[16] Yifeng Zeng, Yinghui Pan Zhangrui Yao, Wanqing Chen, Junhan Chen Junxin Zhou, Biyang Ma, and Zhong Min. 2021. ATPT: Automate Typhoon Contingency Plan Generation from Text. Video. Retrieved January 19, 2021 from https://youtu.be/5BnT-RGULbI