A Novel Network Configuration Generation Scheme from Network Operator's Intent Described by Natural Language

Masaya SUZUKI[†a)], ***Student Member***, **Kimihiro MIZUTANI**[†,††], **Satoru KOBAYASHI**[†††], **Kensuke FUKUDA**[†††], ***and*** **Osamu AKASHI**[†††], ***Members***

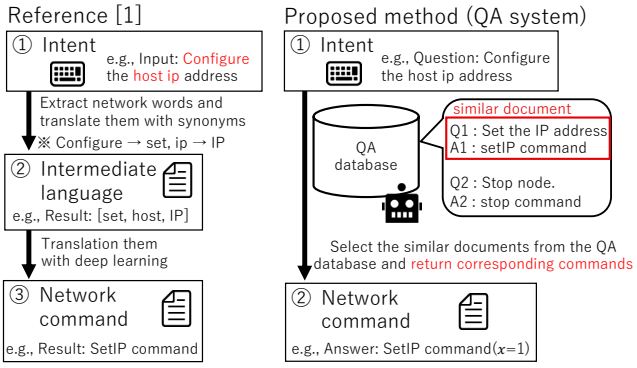

Fig. 1 Abstraction of the reference and our IBN scheme

1. Introduction

Currently, network operators take much time to learn network configuration commands for their equipment. To support the learning, Intent-based networking (IBN) is proposed, which aims to translate the network operator's intent described by natural language to real network configuration. Ref. [1] is the latest IBN scheme, which can translate the operator's intent into the network commands. However, as shown in the left of Fig.1, it must pass the intermediate language translation step, which needs lots of translation rules. To save this cost, we propose a new IBN scheme without the intermediate language translation. Our IBN scheme directly finds corresponding network commands for an operator's input by using deep learning. In evaluation, we confirmed our IBN scheme can extract network operator's intents and generate configurations with a high probability.

2. Proposed scheme

We regard our IBN scheme as a question-answering system (QA system), which returns the corresponding network command from the QA database when the operator inputs a question (i.e., intent) described by natural language (Right of Fig.1). The QA database stores the pairs of a network command and the corresponding document. Receiving the operator's intent (e.g., Configure the host IP address), QA system searches the similar document from QA database. Here, we apply SBERT [2] as the search function, which calculates similarities between the intent and each document and outputs the corresponding network command candidates taking Top-k highest similarity. Receiving the output, the operator selects the appropriate one (e.g., SetIP command) from the candidates, and applies it to a network setting.

3. Evaluation

We aim to build a network environment which consists of a switch and hosts connecting to the switch by using our QA system on Mininet [3]. In detail, we use a building scenario (i.e., config) with some noise as the inputs and check the corresponding returns from the QA system whether the output, which is the corresponding network command candidates taking Top-k highest similarity ($k = 1, 3, 5$), includes the correct command. Notes that the noise indicates the replacement of words with their synonyms and the addition of a noisy word. With these settings, against the building scenario with 100 noise patterns, we evaluate the success probability that the correct answer is included in the return. As a result, the success probabilities were 12%($k = 1$), 89%($k = 3$), and 96%($k = 5$). This result showed that network operators probably select the correct network command that meets their intent for $k \geq 5$. Thus, we conclude that our IBN scheme can generate configurations with a high probability.

Acknowledgement

This work is partially supported by JSPS Kakenhi 20H04185.

References

[1] J.Arthur, et al. "Refining Network Intents for Self-Driving Networks." Computer Communication Review, vol. 48, no. 5, pp. 55–63, 2019.
[2] R. Nils, et al. "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks." in Proc. the 2019 Conference on Empirical Methods in Natural Language Processing, pp. 3980–3990, 2019.
[3] Mininet Python API Reference Manual, `Mininet:http://mininet.org/api/index.html`

[†]Graduate School of Science and Engineering, Kindai University 3-4-1 Kowakae, Higashi Osaka, Osaka, Japan
[††]Cyber Informatics Research Institute, Kindai University, Same location as Kindai University.
[†††]National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo.
a) E-mail: suzuki.masaya@kindai.ac.jp