NL2Bash: A Corpus and Semantic Parser for Natural Language Interface to the Linux Operating System

Xi Victoria Lin[1]* Chenglong Wang[2] Luke Zettlemoyer[2] Michael D. Ernst[2]

*Work done at the University of Washington

[1] Salesforce Research [2]University of Washington

[1] xilin@salesforce.com [2]{clwang, lsz, mernst}@cs.washington.edu

Overview & Motivation

We present new data and semantic parsing methods for the problem of mapping natural language sentences to Bash commands (NL2Bash). Our corpus consists of ~10,000 one-line Bash commands scraped from the web, paired with expert-written natural language descriptions. It is by far the largest NL-to-code corpus built from practical code snippets and expert-written natural language.

Goal: To enable any end user to perform complex but otherwise repetitive tasks on a computer by simply stating their goals in English.

Example NL-Bash Pairs

1. "Remove all the pdfs in my current directory."

```
> find . -name "*.pdfs" -exec rm {} \;
```

2. "View remaining disk space."

```
> df -h
```

In-scope Bash Syntax

1. Single command that consists of the utility, option flags and arguments
2. Logical connectives: &&, ||, (), etc.
3. Nested commands constructed using pipeline |, command substitution $() and process substitution <(), where the input argument of one command is another command('s output)

Evaluation Methods & Results

Manual Evaluation

- Three programmers judged the correctness of the translation output and we took their majority vote.
- We compute both the full command accuracy and the command structure accuracy, i.e. ignoring errors in the entity strings.

Table 4: Performance of the Baseline Systems on 100 randomly sampled dev set examples.

Model		Acc_F^1	Acc_F^3	Acc_T^1	Acc_T^3
Seq2Seq	Char	0.24	0.27	0.35	0.38
	Token	0.10	0.12	**0.53**	0.59
	Sub-token	0.19	0.27	0.41	0.53
CopyNet	Char	0.25	0.31	0.34	0.41
	Token	0.21	0.34	0.47	**0.61**
	Sub-token	**0.31**	**0.40**	0.44	0.53
Tellina		0.29	0.32	0.51	0.58

- Acc_F^k - top-k full command accuracy
- Acc_T^k - top-k command structure accuracy
- We use manual evaluation since an NL description may have multiple translations and not all of them are in the corpus.

Table 5: Full test set performance of the two best systems, ST-CopyNet and Tellina.

Model	Acc_F^1	Acc_F^3	Acc_T^1	Acc_T^3
ST-CopyNet	**0.36**	**0.45**	0.49	0.61
Tellina	0.27	0.32	0.53	0.62

Corpus Construction

Challenges

- Code snippets paired with natural language descriptions are rare.
- Collecting/Generating this type of data requires expert knowledge.

Data Collection

Programmers collected Bash one-liners from programming help websites and copied/wrote a natural language description for each of them.

Natural language diversity: Multiple programmers may annotate the same command (in the same or different webpages).

Data Filtering & Cleanning

✓Filter out non-grammatical commands & commands with out-of-scope syntax.

✓Filter out commands that contain non-Bash program interpreters, such as python, c++ and emacs.

✓Fix spelling errors.

Data Statistics

We collected 12,609 NL-Bash pairs in total (9,305 pairs after filtering). The Bash commands cover 102 unique utilities and 206 option flags. The distribution is long-tailed.

Table 1: Natural Language Statistics

# sent.	# word	# words per sent.		# sent. per word	
		avg.	median	avg.	median
8,559	7,790	11.7	11	14.0	1

Table 2: Bash Command Statistics

# cmd	# temp	# token	# tokens / cmd		# cmds / token	
			avg.	median	avg.	median
7,587	4,602	6,234	7.7	7	11.5	1

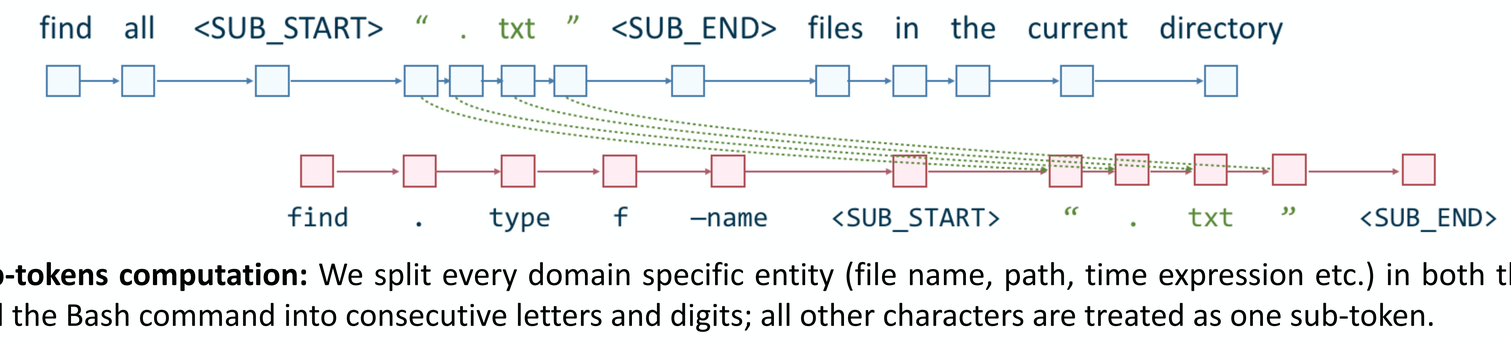
# utility	# flag	# reserv. token	# cmds / util.		# cmds / flag	
			avg.	median	avg.	median
102	206	15	155.0	38	101.7	7.5

Table 3: Data Split Statistics

	Train	Dev	Test
# pairs	8,090	609	606
# unique nls	7,340	549	547

Baseline Approaches

Sub-Token Sequence-to-Sequence Translation with Copying



Sub-tokens computation: We split every domain specific entity (file name, path, time expression etc.) in both the NL and the Bash command into consecutive letters and digits; all other characters are treated as one sub-token.

[1] Incorporating Copying Mechanism in Sequence-to-Sequence Learning. Gu et. al. ACL 2016.

[2] Sequence to Sequence Learning with Neural Networks. Sutskever et. al. NIPS 2014.

Resources

Please visit the project website to test our pre-trained NL2Bash translator:
http://tellina.rocks

The code and data for this paper is released at:
https://github.com/TellinaTool/nl2bash

Our paper is on arXiv:
https://arxiv.org/abs/1802.08979

Please email us any questions or comments.

Scan me