

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

find system log files
older than a month

```
find / -name "*.log"  
-mtime +30
```

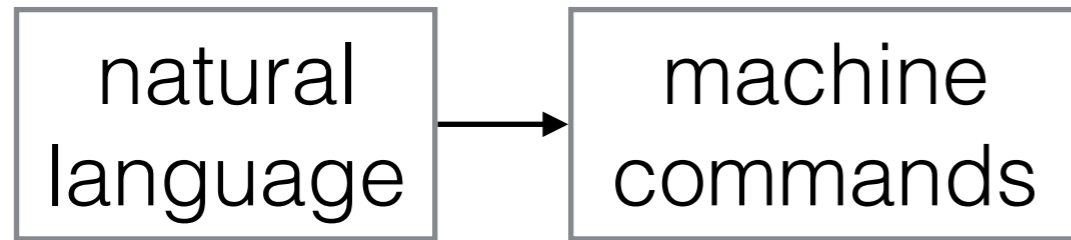
Victoria Lin[☂] Chenglong Wang[☂] Luke Zettlemoyer[☂]

Michael D. Ernst[☂]

{xilin,clwang,lsz,mernst}@cs.washington.edu

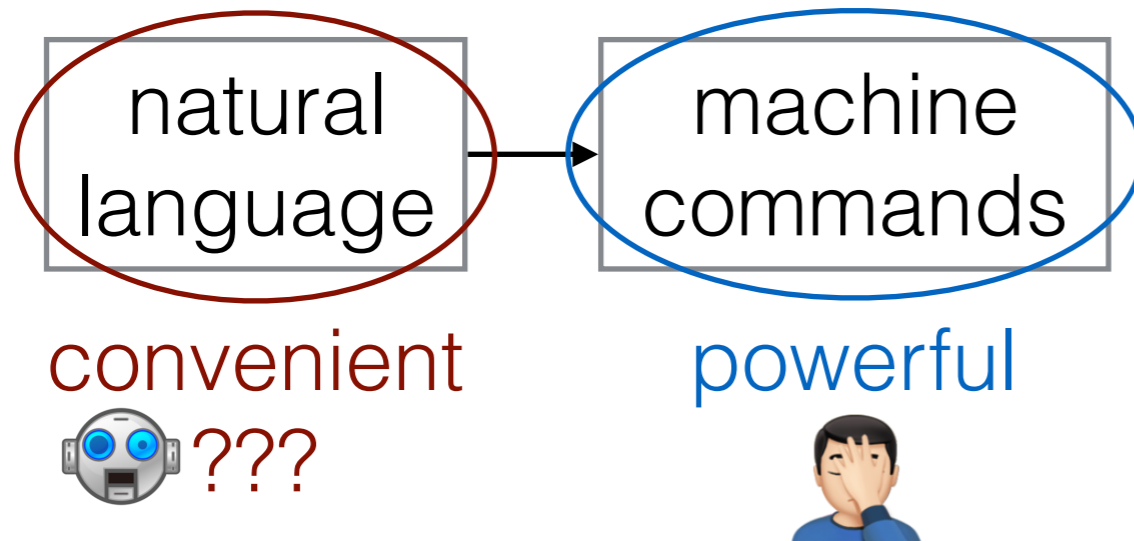
OUTLINE

Problem Definition



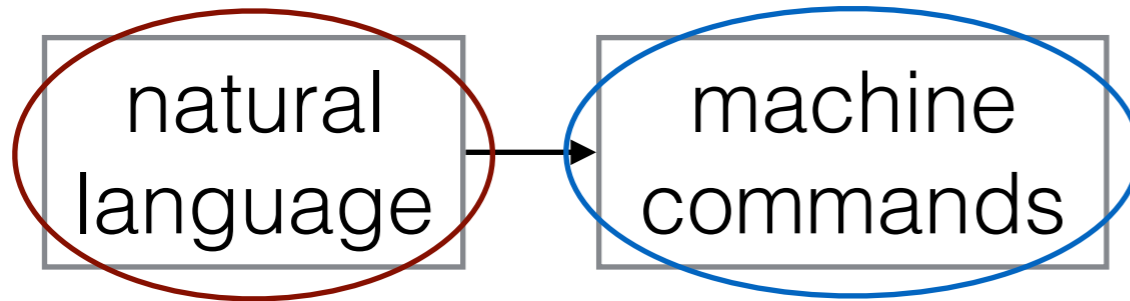
OUTLINE

Problem Definition



OUTLINE

Problem Definition



convenient



Domain

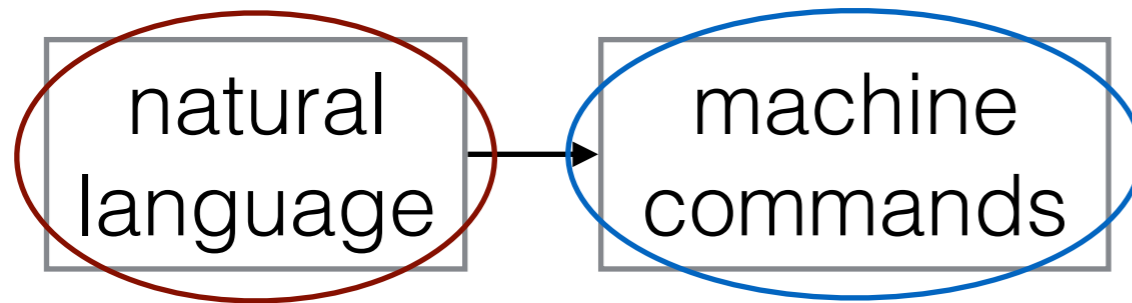
powerful



```
Victoria-MacBook-Pro-2:Projects xllin$ ls -l | display all files
total 0
drew--x-x 22 xllin staff 748 Jun 8 12:43 helper
drew--x-x 9 xllin staff 386 Apr 21 2016 opoloye
drew--x-x 7 xllin staff 236 Jun 9 2016 opoloye_kbp
drew--x-x 5 xllin staff 170 Dec 17 2015 pigwidgeon
drew--x-x 13 xllin staff 442 Mar 15 22:47 reflexnet
drew--x-x 12 xllin staff 486 Jun 6 14:14 resume
drew--x-x 26 xllin staff 884 Dec 18 2015 skim
drew--x-x 22 xllin staff 748 Jun 9 2016 toplocc
drew--x-x 31 xllin staff 1894 Feb 21 14:15 task_platform
drew--x-x 16 xllin staff 544 Mar 29 18:18 tellino
drew--x-x 52 xllin staff 1768 May 10 20:44 tellino_fsm
drew--x-x 11 xllin staff 374 Mar 21 21:02 todoo3_github.io
drew--x-x 62 xllin staff 2186 Oct 9 2015 tutor
drew-----@ 16 xllin staff 544 Feb 12 12:13 xllin-cog
Victoria-MacBook-Pro-2:Projects xllin$ find -name beam_search.py | find all files named "beam_search.py"
./helper/encoder_decoder/beam_search.py
./reflexnet/net/beam_search.py
./tellino/tellino_learning_module/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name beam_search.py -mtime -1 | find all files named "beam_search.py" that was modified today
./helper/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$
```

OUTLINE

Problem Definition



convenient



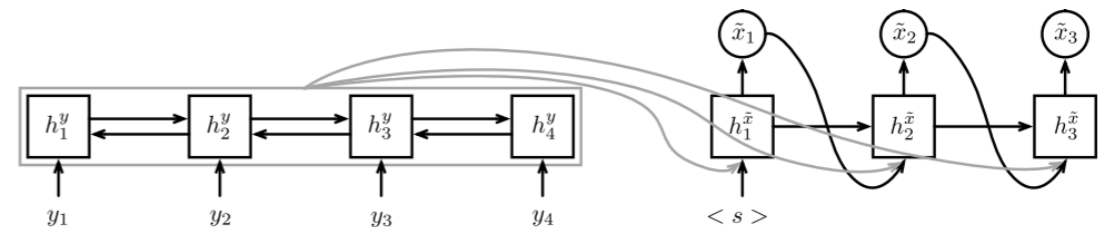
powerful



Domain

```
Victoria-MacBook-Pro-2:Projects xllin$ ls -l
total 0
drwxr-xr-x  22 xllin  staff  748 Jun  8 12:43 helper
drwxr-xr-x   9 xllin  staff  288 Apr 21 2016 opoloye
drwxr-xr-x   7 xllin  staff  236 Jun  9 2016 opoloye_kbp
drwxr-xr-x   5 xllin  staff  170 Dec 17 2015 pigwidgeon
drwxr-xr-x  13 xllin  staff  442 Mar 15 22:47 reflexnet
drwxr-xr-x  12 xllin  staff  486 Jun  6 14:14 resume
drwxr-xr-x  26 xllin  staff  884 Dec 18 2015 skim
drwxr-xr-x  22 xllin  staff  748 Jun  9 2016 toploc
drwxr-xr-x  31 xllin  staff 1894 Feb 21 14:13 task_platform
drwxr-xr-x  16 xllin  staff  544 Mar 29 18:18 tellino
drwxr-xr-x  32 xllin  staff 1768 May 10 20:44 tellino_fsm
drwxr-xr-x  11 xllin  staff  374 Mar 31 21:02 todoo3_github.io
drwxr-xr-x@ 62 xllin  staff 2186 Oct  9 2015 tutor
dme-----@ 16 xllin  staff  544 Feb 12 12:13 xpsa-cop
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py"
./reflexnet/encoder_decoder/beam_search.py
./reflexnet/net/beam_search.py
./tellino/tellino_learning_module/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py" -mtime -1
./reflexnet/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py"
that was modified today
```

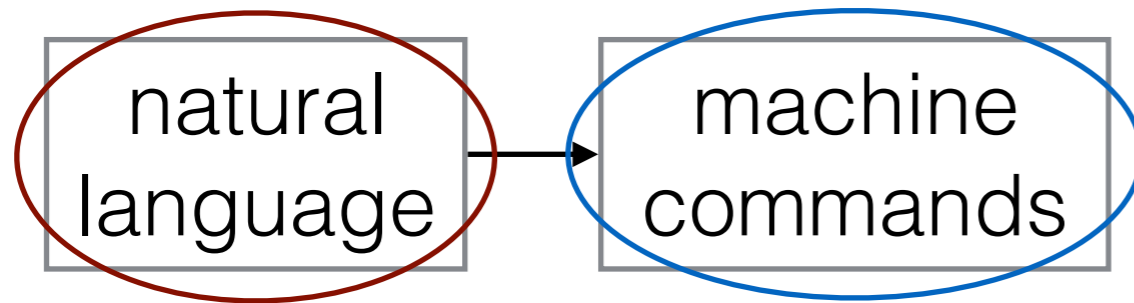
Data-Driven Approaches



Adaptions from state-of-the-art neural machine translation models

OUTLINE

Problem Definition



convenient



powerful



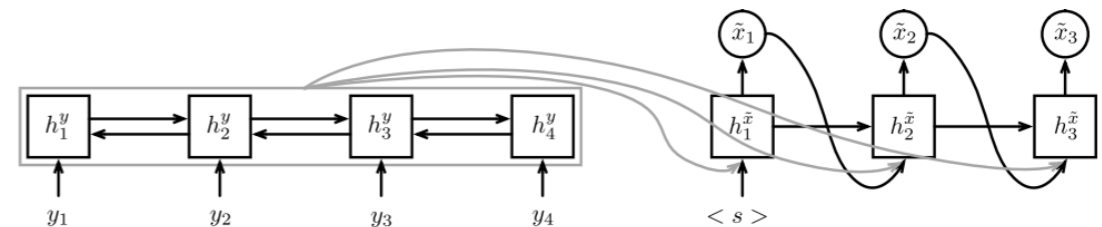
Corpus Construction



Domain

```
Victoria-MacBook-Pro-2:Projects xllin$ ls -l
total 0
drwxr-xr-x  22 xllin  staff  748 Jun  8 12:43 helper
drwxr-xr-x   9 xllin  staff  288 Apr 21 2016 opoloye
drwxr-xr-x   7 xllin  staff  236 Jun  9 2016 opoloye_kbp
drwxr-xr-x   5 xllin  staff  170 Dec 17 2015 pigwidgeon
drwxr-xr-x  13 xllin  staff  442 Mar 15 22:47 reflexnet
drwxr-xr-x  12 xllin  staff  486 Jun  6 14:14 resume
drwxr-xr-x  26 xllin  staff  884 Dec 18 2015 skim
drwxr-xr-x  22 xllin  staff  748 Jun  9 2016 toplocq
drwxr-xr-x  31 xllin  staff 1894 Feb 21 14:13 task_platform
drwxr-xr-x  16 xllin  staff  544 Mar 29 18:18 tellino
drwxr-xr-x  32 xllin  staff 1768 May 10 20:44 tellino_fsm
drwxr-xr-x  11 xllin  staff  374 Mar 31 21:02 todoo3_github.io
drwxr-xr-x@ 62 xllin  staff 2186 Oct  9 2015 tutor
dme-----@ 16 xllin  staff  544 Feb 12 12:13 xpsa-csg
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py"
./helper/encoder_decoder/beam_search.py
./reflexnet/net/beam_search.py
./tellino/tellino_learning_module/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py" -mtime -1
./helper/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py"
that was modified today
```

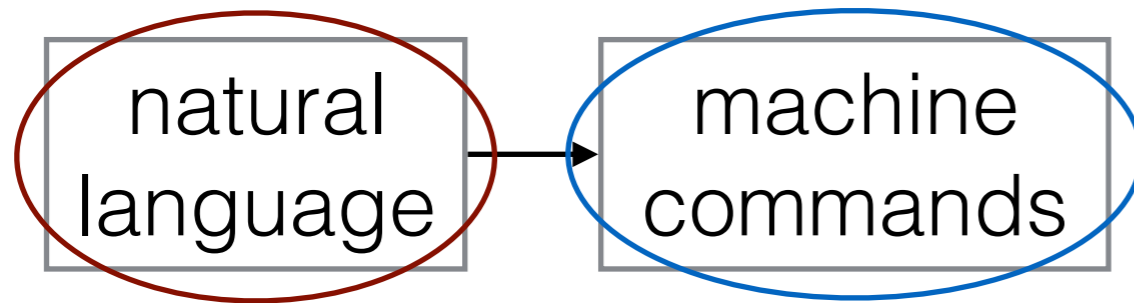
Data-Driven Approaches



Adaptions from state-of-the-art neural machine translation models

OUTLINE

Problem Definition



convenient



powerful



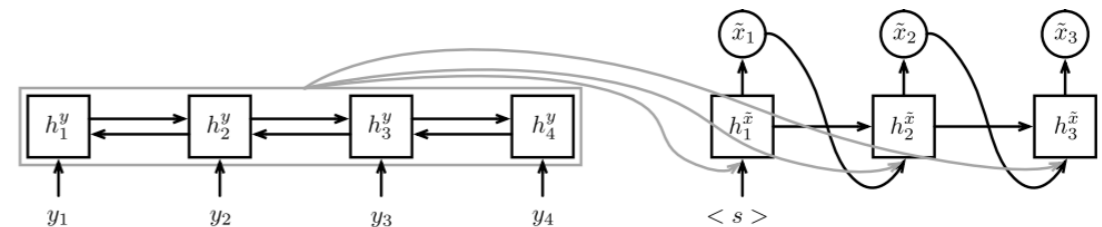
Corpus Construction



Domain

```
Victoria-MacBook-Pro-2:Projects xllin$ ls -l | display all files
total 0
drew-x-x 22 xllin staff 748 Jun 8 12:43 helper
drew-x-x 9 xllin staff 386 Apr 21 2016 opolye
drew-x-x 7 xllin staff 236 Jun 9 2016 opolye_kbp
drew-x-x 5 xllin staff 170 Dec 17 2015 pigwidgeon
drew-x-x 13 xllin staff 442 Mar 15 22:49 reflexnet
drew-x-x 12 xllin staff 486 Jun 6 14:14 resume
drew-x-x 26 xllin staff 884 Dec 18 2015 skim
drew-x-x 22 xllin staff 748 Jun 9 2016 toploc
drew-x-x 31 xllin staff 1894 Feb 21 14:13 task_platform
drew-x-x 16 xllin staff 544 Mar 29 18:18 tellino
drew-x-x 32 xllin staff 1768 May 10 20:44 tellino_fsm
drew-x-x 11 xllin staff 374 Mar 31 21:08 todopol3.github.io
drew-x-x@ 62 xllin staff 2186 Oct 9 2015 tutor
drew-x-x@ 16 xllin staff 544 Feb 12 12:13 vpsa-csg
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py" | find_all_files_named "beam_search.py"
./reflexnet/encoder_decoder/beam_search.py
./reflexnet/net/beam_search.py
./tellino/tellino_learning_module/encoder_decoder/beam_search.py
Victoria-MacBook-Pro-2:Projects xllin$ find -name "beam_search.py" -ctime -1 | find_all_files_named "beam_search.py" that was modified today
```

Data-Driven Approaches



Adaptions from state-of-the-art neural machine translation models

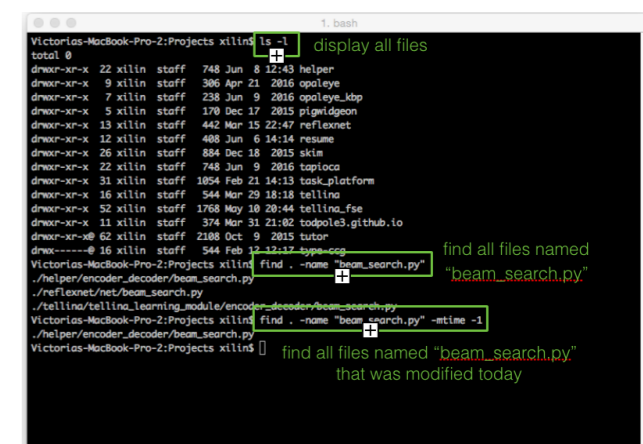
- System Performance
- Qualitative Analysis
- Live Demo

PROBLEM DEFINITION

- Natural Language → Command Translation
 - Generating **one-liners**
 - In most command languages complex semantics can be represented in short syntactic forms
 - Other work: code block generation (Polosukhin and Skidanov '18)
 - **Single-turn interaction** between the user & the system (building block for multi-turn system)
 - Other work: conversational natural language programming assistant (Pandita et. al. '18)
 - Semantic parsing can be a building block conversational programming assistant

DOMAIN - BASH

- Potentially Wide User Base
 - Most Linux users know bash, but not mastering it
- Command Interface Language
- Generalizable to other command languages



```
Victorias-MacBook-Pro-2:Projects xilinx$ ls -l
total 0
dwxr-xr-x  22 xilin  staff  748 Jun  8 12:43 helper
dwxr-xr-x   9 xilin  staff  306 Apr 21 2016 opalaye
dwxr-xr-x   7 xilin  staff  238 Jun  9 2016 opalaye_kbp
dwxr-xr-x   5 xilin  staff  170 Dec 17 2015 pigwidgeon
dwxr-xr-x  13 xilin  staff  442 Mar 15 22:47 reflexnet
dwxr-xr-x  12 xilin  staff  408 Jun  6 14:14 resume
dwxr-xr-x  26 xilin  staff  884 Dec 18 2015 skim
dwxr-xr-x  22 xilin  staff  748 Jun  9 2016 taploca
dwxr-xr-x  31 xilin  staff 1854 Feb 21 14:13 task_platform
dwxr-xr-x  16 xilin  staff  544 Mar 29 18:18 tellina
dwxr-xr-x  52 xilin  staff 1768 May 10 20:44 tellina_fsm
dwxr-xr-x  11 xilin  staff  374 Mar 31 21:02 todooe3.github.io
dwxr-xr-x@ 62 xilin  staff 2188 Oct  9 2015 tutor
dwxr-----@ 16 xilin  staff  544 Feb 12 12:17 type-csg
Victorias-MacBook-Pro-2:Projects xilinx$ find . -name "beam_search.py"
./reflexnet/net/beam_search.py
./tellina/tellina_learning_module/encoder_decoder/beam_search.py
Victorias-MacBook-Pro-2:Projects xilinx$ find . -name "beam_search.py" -mtime -1
./reflexnet/net/beam_search.py
./tellina/tellina_learning_module/encoder_decoder/beam_search.py
Victorias-MacBook-Pro-2:Projects xilinx$ find . -name "beam_search.py" -mtime -1
Victorias-MacBook-Pro-2:Projects xilinx$ find . -name "beam_search.py" -mtime -1
```

BASH EXAMPLE

- find all '*.c' files under \$HOME directory which contain the string "Salesforce"

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"Salesforce" {}
```

BASH EXAMPLE

- find all '*.c' files under \$HOME directory which contain the string "Salesforce"

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"Salesforce" }
```

Head command

BASH EXAMPLE

- find all '*.c' files under \$HOME directory which contain the string "Salesforce"

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"Salesforce" {}
```

Flag

BASH EXAMPLE

- find all '*.c' files under \$HOME directory which contain the string "Salesforce"

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"Salesforce" {}
```

Argument

BASH EXAMPLE

- find all '*.c' files under \$HOME directory which contain the string "Salesforce"

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"Salesforce" {}
```

Compound Commands

RELATED WORK

- Neural Networks: Natural Language → Formal Languages
 - ✓ NL → Syntactic parse trees (Vinyals et. al. '14)
 - ✓ NL → Regular expression (Locascio et. al. '16)
 - ✓ NL → Logical forms (Li & Lapata '16)
 - ✓ NL → Python (Wang et. al. '16)
 - ✓ NL → Python (Yin & Neubig '17, Rabinovich et. al. '17)

Rule-Based
Systems

Statistical Models over
Discrete Structures

RELATED WORK

- Neural Networks: Natural Language → Formal Languages
 - ✓ NL → Syntactic parse trees (Vinyals et. al. '14)
 - ✓ NL → Regular expression (Locascio et. al. '16)
 - ✓ NL → Logical forms (Li & Lapata '16)
 - ✓ NL → Python (Wang et. al. '16)
 - ✓ NL → Python (Yin & Neubig '17, Rabinovich et. al. '17)

Adapted from NMT methods for natural language translation

RELATED WORK

- Neural Networks: Natural Language → Formal Languages

✓ NL → Syntactic parse trees (Vinyals et. al. '14)

✓ NL → Regular expression (Locascio et. al. '16)

✓ NL → Logical forms (Li & Lapata '16)

✓ NL → Python (Wang et. al. '16)

✓ NL → Python (Yin & Neubig '17, Rabinovich et. al. '17)

Seq2Seq

Seq2Tree

Expressive → Simplest Data Representation

RELATED WORK

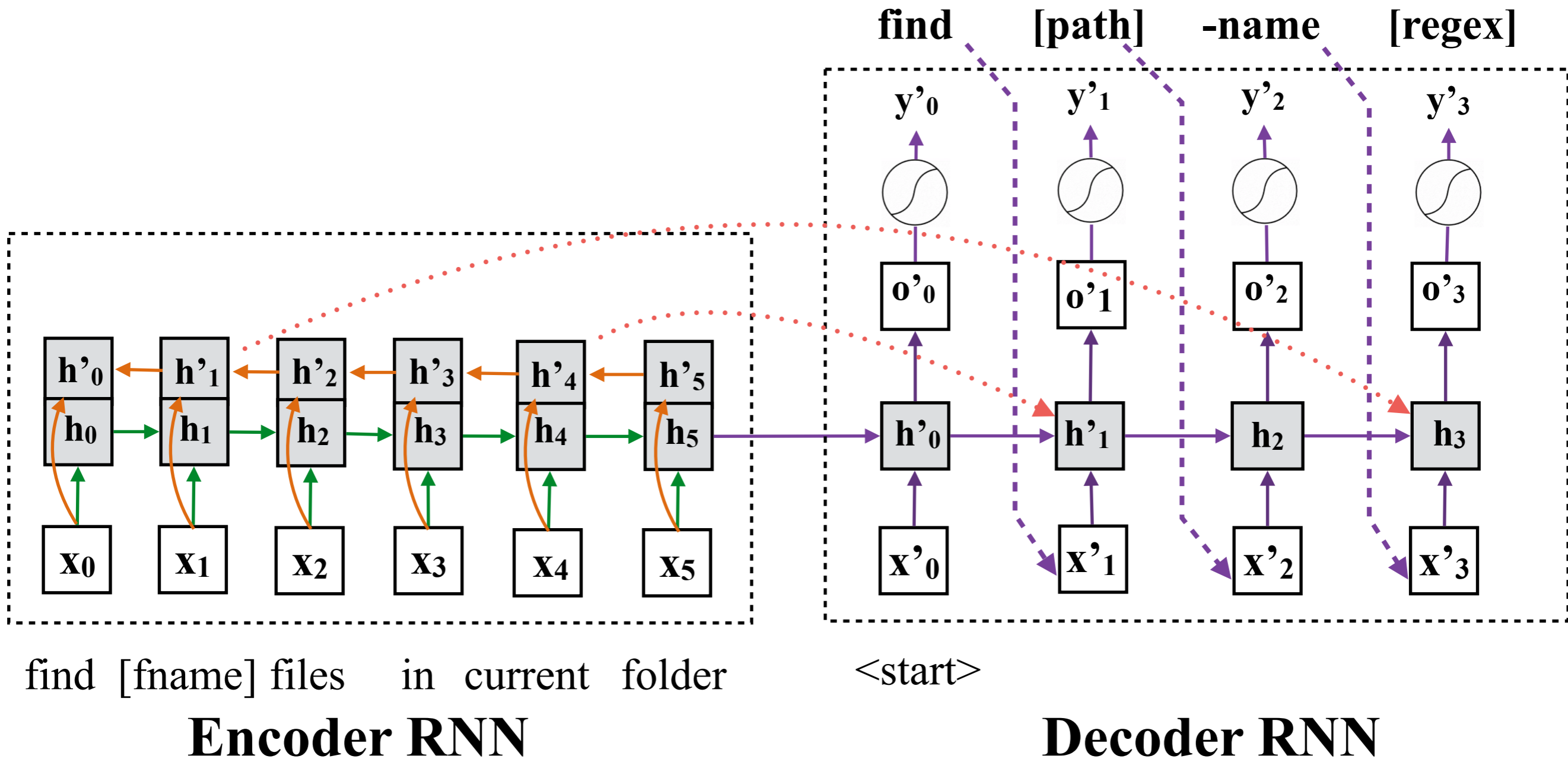
- Neural Networks: Natural Language → Formal Languages

- ✓ NL → Syntactic parse trees (Vinyals et. al. '14)
- ✓ NL → Regular expression (Locascio et. al. '16)
- ✓ NL → Logical forms (Li & Lapata '16)
- ✓ NL → Python (Wang et. al. '16)

Seq2Seq

Target Domain: Shallow Syntax Structure (No Formal Grammar)

SEQUENCE-TO-SEQUENCE NEURAL NETWORK



SEQ2SEQ + COPYING

- find all **'*.c'** files under **\$HOME** directory whose content has the string **"salesforce"**

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"salesforce" }
```

x Large number of out-of-vocabulary words (arguments)

SEQ2SEQ + COPYING

- find all **'*.c'** files under **\$HOME** directory whose content has the string **"salesforce"**

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep "salesforce" }
```

x Large number of out-of-vocabulary words

SEQ2SEQ + COPYING

- find all `'*.c'` files under `$HOME` directory whose content has the string `"salesforce"`

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"salesforce" }
```

- ✗ Many command arguments are source tokens transformed through atomic string edits

SEQ2SEQ + COPYING

- find all **'*.c'** files under **\$HOME** directory whose content has the string **"salesforce"**

```
find "$HOME" -name "*.c" -print0 | xargs -0 -I {} grep "salesforce" }
```

- ✗ Many command arguments are source tokens transformed through atomic string edits

Character models? Very long sequences...

SUB-TOKEN COPYING

- find all '***.c**' files under **\$ HOME** directory whose content has the string "**salesforce**"

```
find "$ HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"salesforce" }
```

Split the constant tokens in both the source and target sequences into a sequence of sub-tokens consists of the following:

1. Consecutive sub-sequences of alphabetical letters
2. Consecutive sub-sequences of digits
3. All other special tokens

Run CopyNet on the sub-tokens

SUB-TOKEN COPYING

- find all '***.c**' files under **\$ HOME** directory whose content has the string "**salesforce**"

```
find "$ HOME" -name "*.c" -print0 | xargs -0 -I {} grep  
"salesforce" }
```

Enables learning of

1. Substring addition
2. Substring deletion
3. Substring replacement
4. Semantics of the special characters such as "\$", quotation marks, "*", etc.

DATA COLLECTION

- Bash programmers hired **upwork**TM
- Collect bash commands and their natural language descriptions from the web



✓ web interface to control the collection process

BASH COMMAND FILTERING

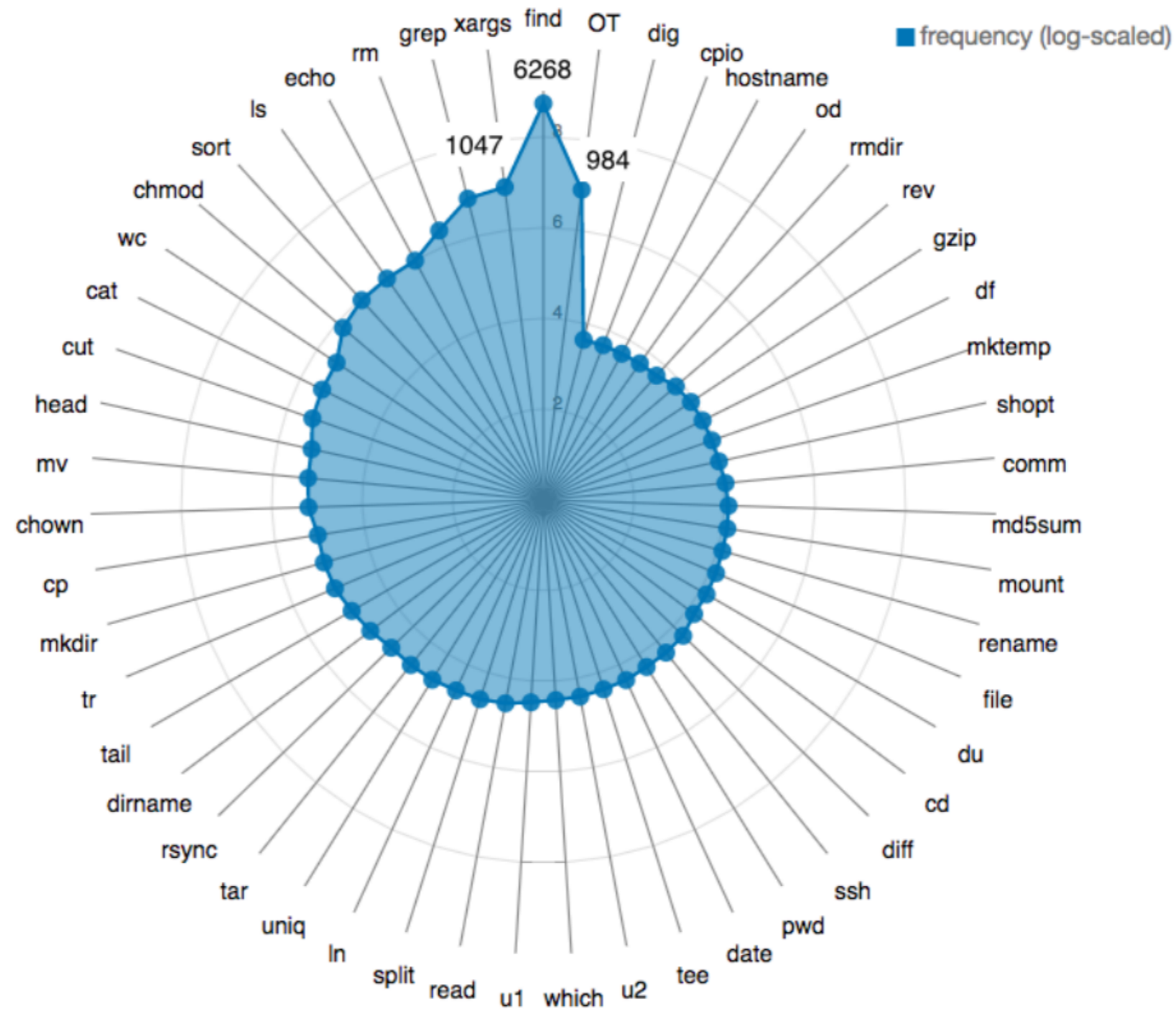
- Bash Command

In-scope	Single command	
	Logical connectives	&&, , ()
		pipeline
	Nested command	command substitution \$() process substitution <()
Out-of-scope	I/O redirection	<, <<
	Variable assignment	=
	Parameters	e.g. \$1, \$HOME
	Multi-statement	if, for, while, until, etc.
	Regex structure	e.g. x*y*
	Non-bash programs	triggered by awk, java, etc.

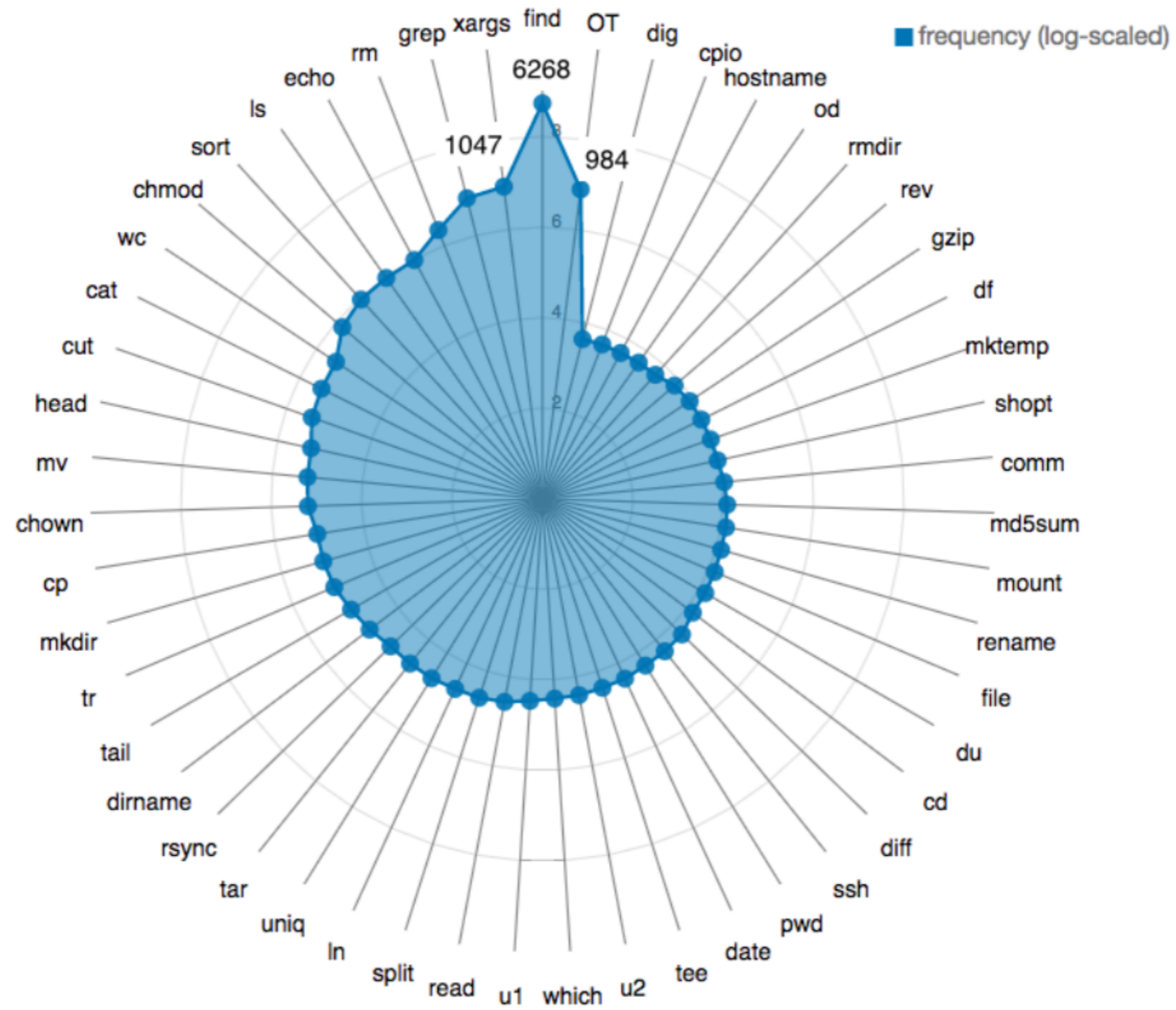
DATA STATISTICS

- 12,609 pairs \rightarrow 9,301 pairs after filtering
- 8,090 train, 609 dev, 606 test
- 100+ unique bash commands, 537 unique flags

TOP-50 COMMAND HISTOGRAM



TOP-50 COMMAND HISTOGRAM



The rest combined: 984

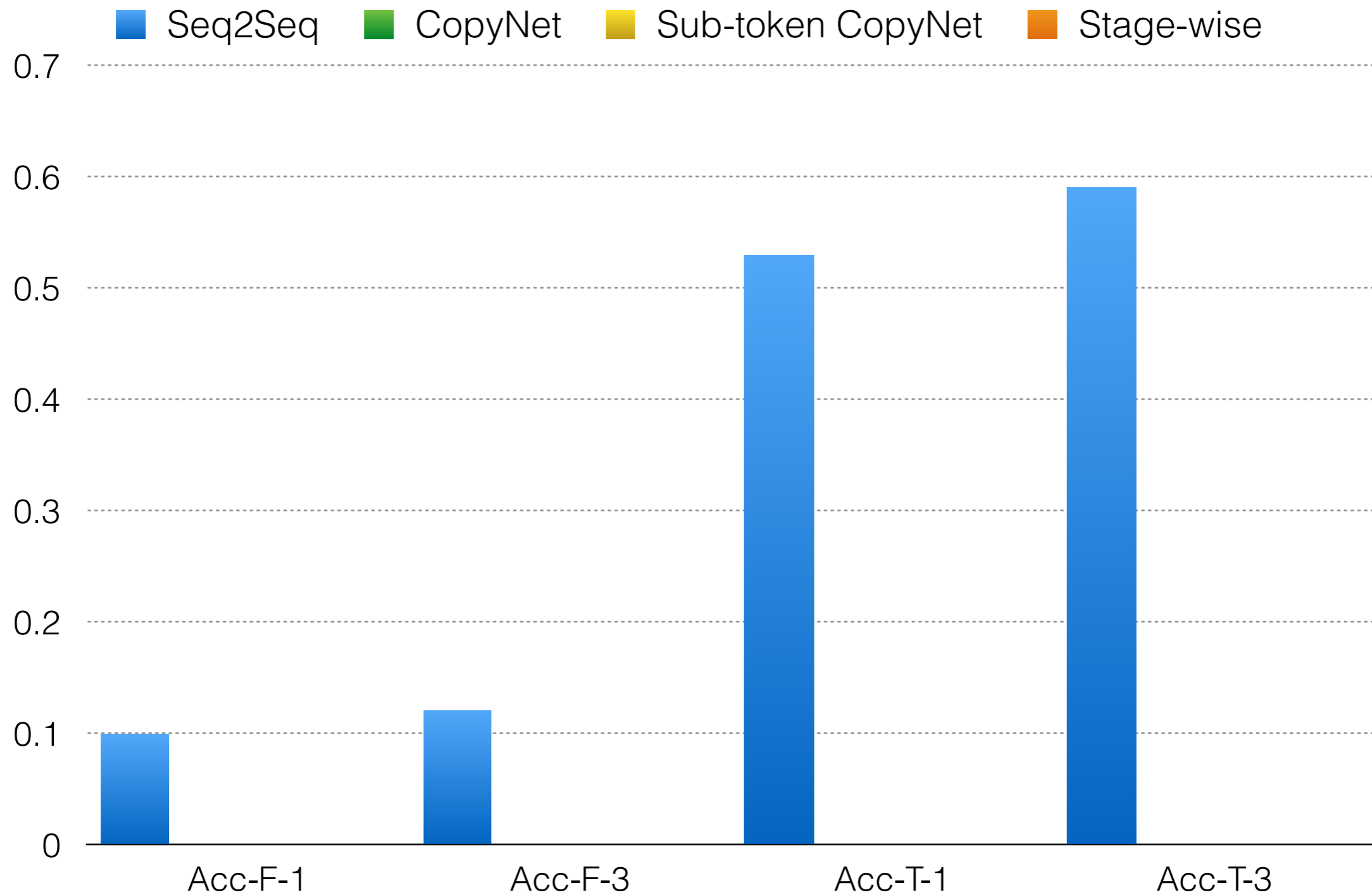
EVALUATION METHODOLOGY

- Manual Evaluation (Multiple Correct Solutions)
 - 3 bash programmers (hired via **upwork**[™]) judged the top-3 suggestions of each test example
 - Full command correctness
 - Command template correctness
- find [path] -name [regex] -print0 | xargs -0 -I {} grep [regex] {}**
- Final judgement: majority vote
 - Inter-annotator agreement: 0.89, 0.83, 0.80

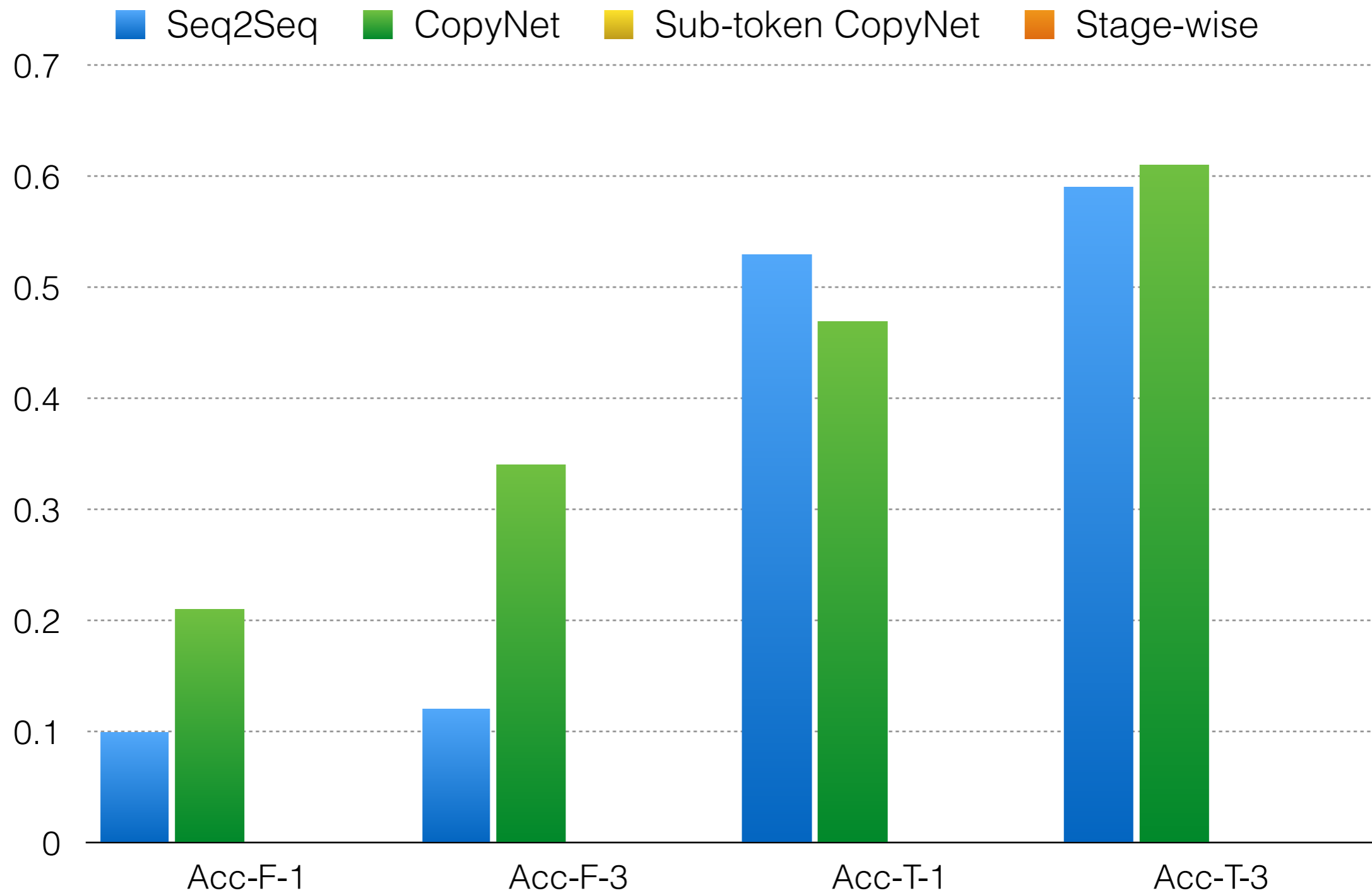
BASELINES

- Vanilla Seq2Seq (Sutskever et. al. '14)
- CopyNet (Gu et. al. '17)
- Three-stage translation model (Lin et. al. '17)
 1. Convert both NL and bash command to templates
 2. Apply Seq2Seq translation on the templates
 3. Fill arguments using heuristics

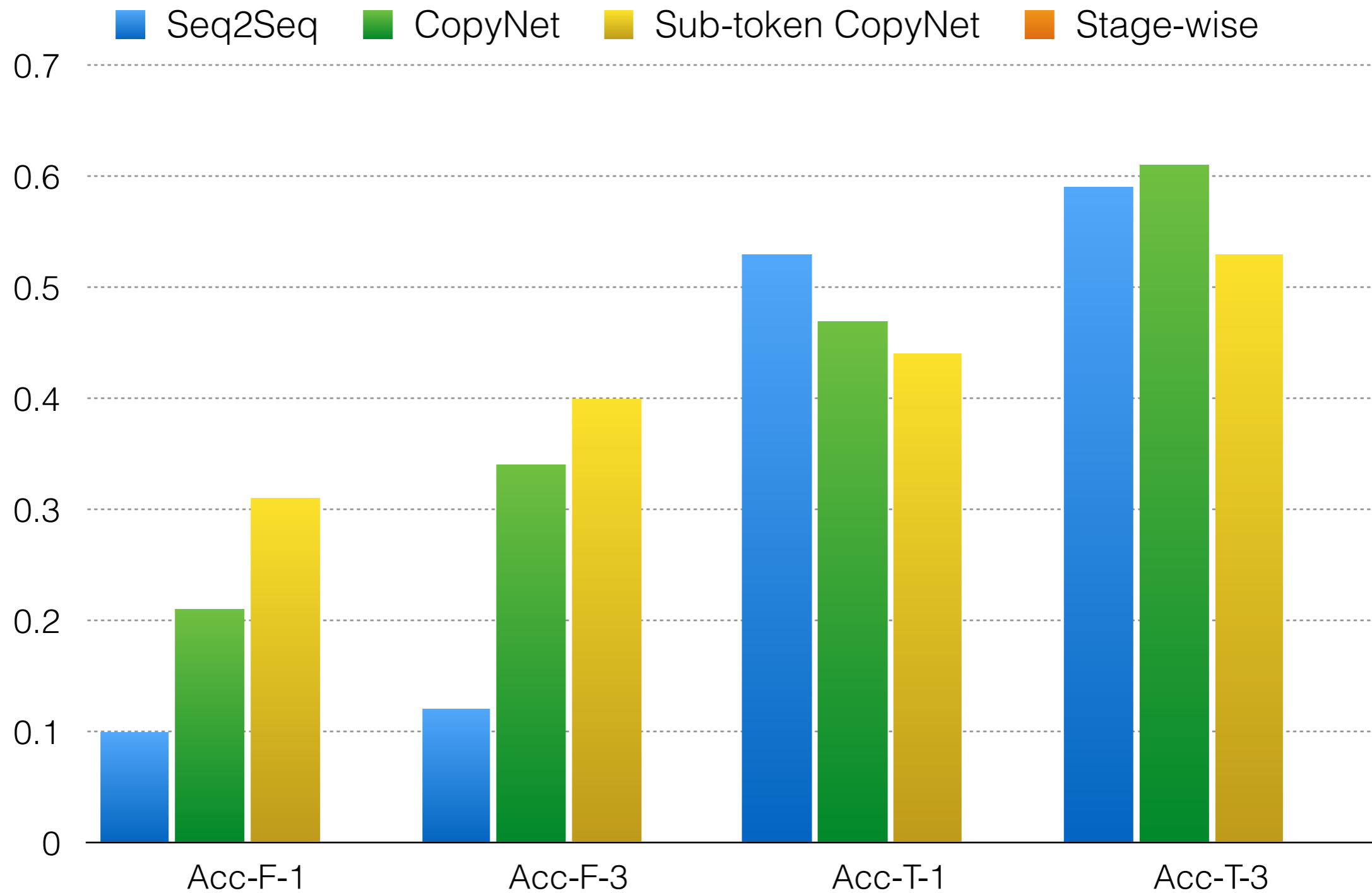
SYSTEM PERFORMANCE (Dev Set)



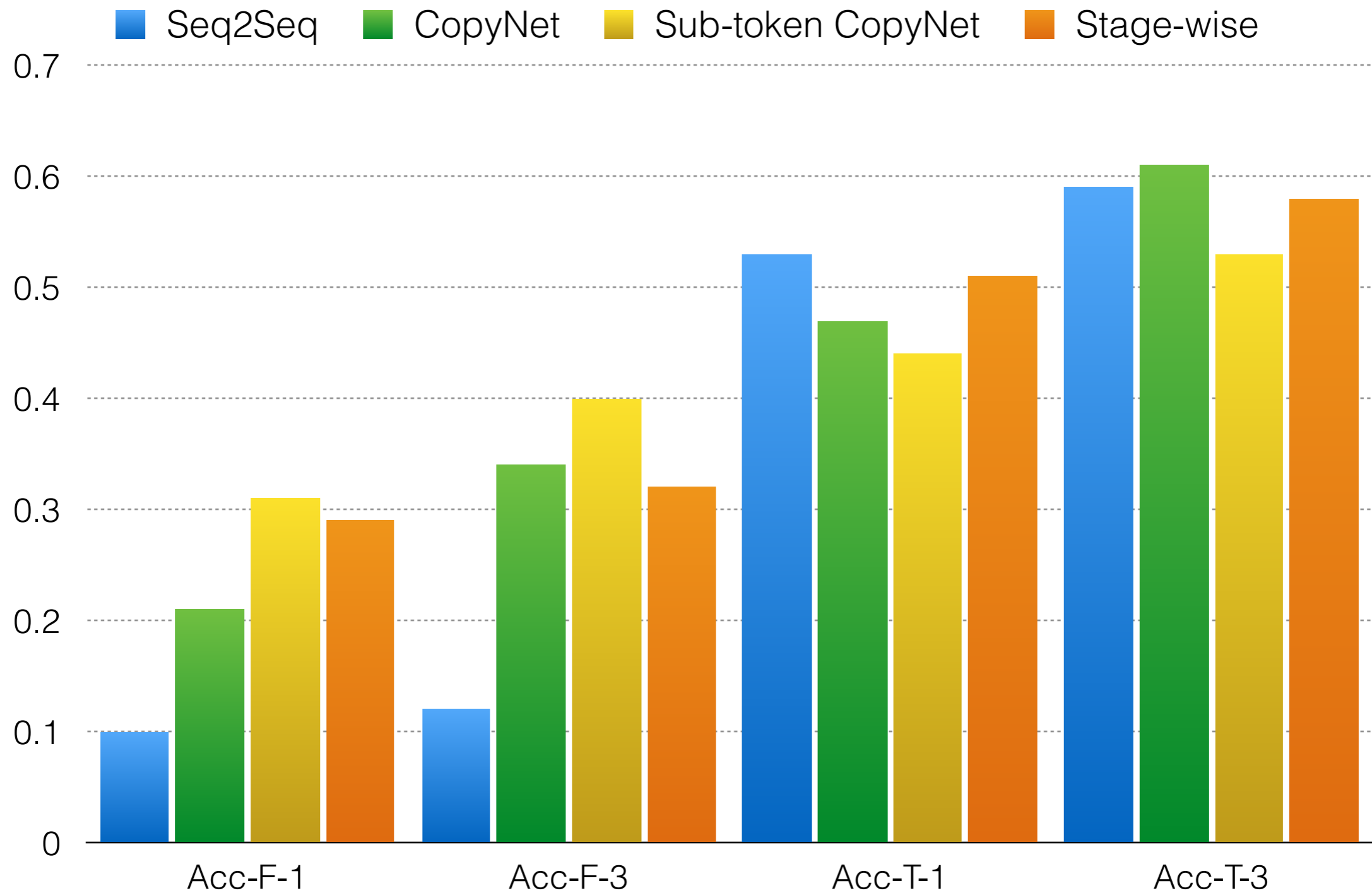
SYSTEM PERFORMANCE (Dev Set)



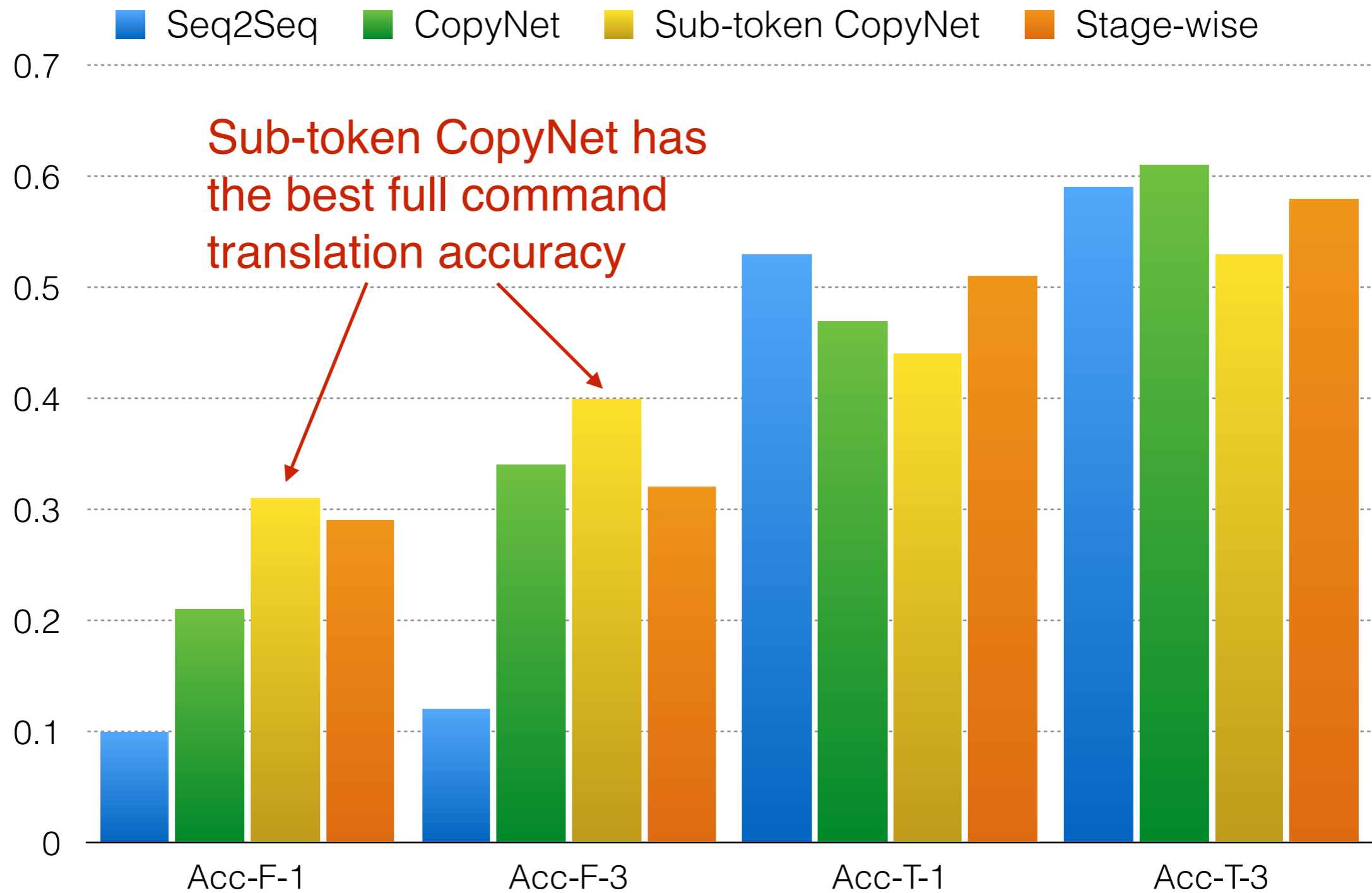
SYSTEM PERFORMANCE (Dev Set)



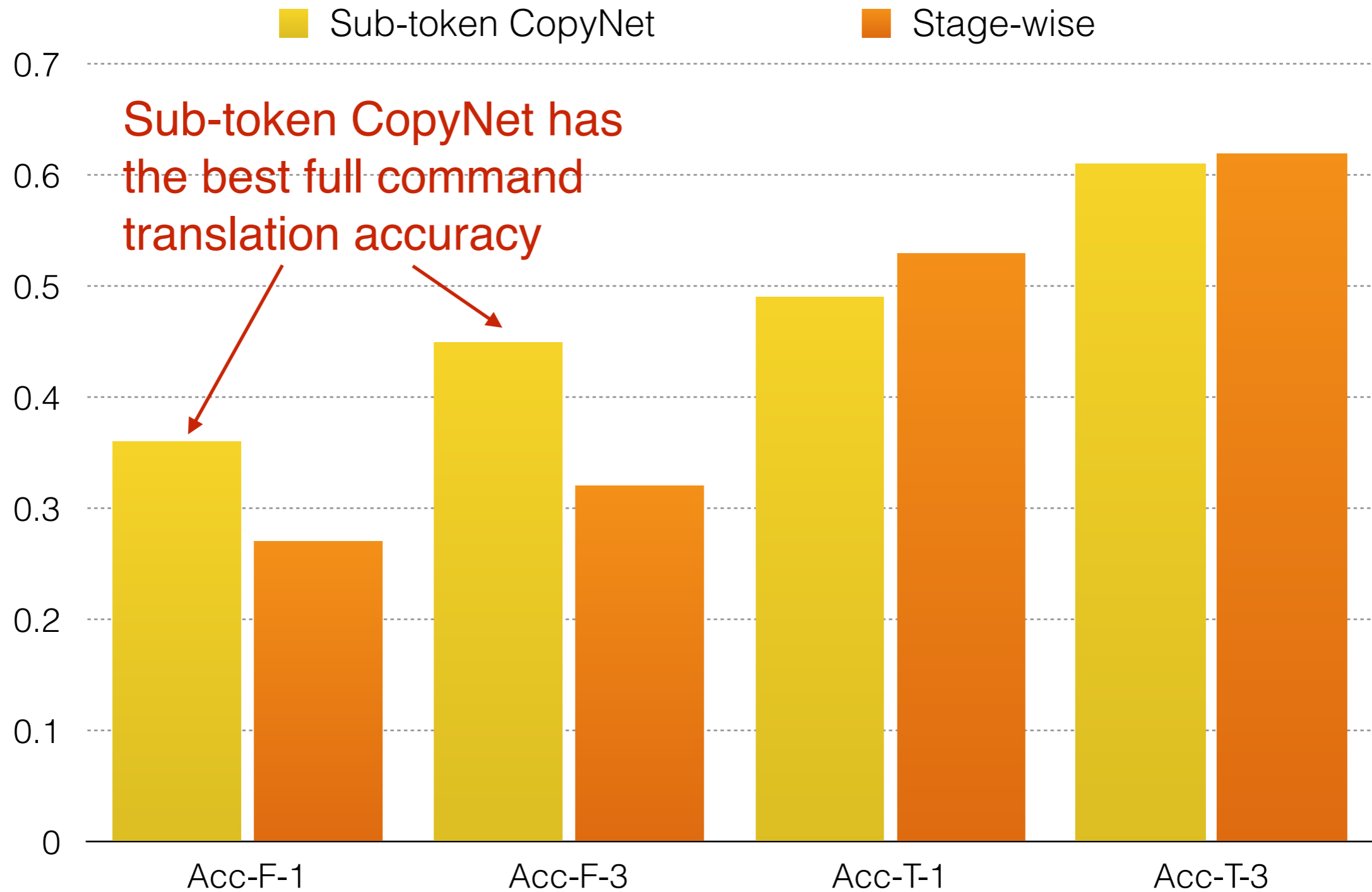
SYSTEM PERFORMANCE (Dev Set)



SYSTEM PERFORMANCE (Dev Set)



SYSTEM PERFORMANCE (Test Set)



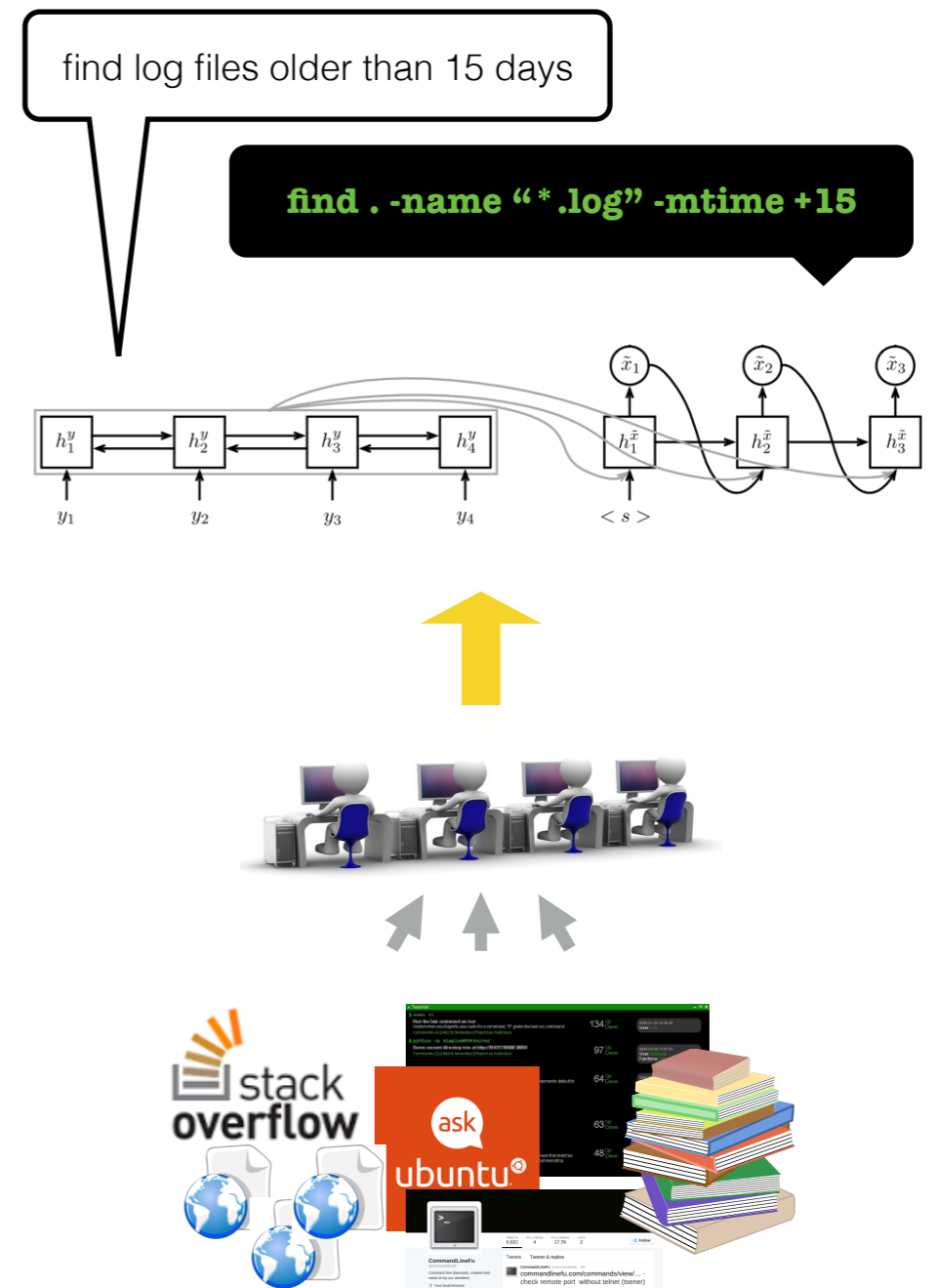
QUALITATIVE ANALYSIS

- Live Demo: <http://tellina.rocks>
- Split '/usr/bin/gcc' into 10 files of about equal size
- Which files in the computer were modified more than 30 days ago and larger than 500M
- Find all *company* (case-insensitive) files/directories under /basedir with null character as the delimiter

Github: <https://github.com/TellinaTool>

Demo: <http://tellina.rocks>

- **Corpus:** 10k real-world bash commands, paired with human-written English descriptions
- **Data-driven baselines:** motivated by SOTA neural machine translation approaches *copying*, *sub-token modeling*
- **Huge space for improvements**
- To appear in LREC 2018 conference proceedings
- Contact: xilin@salesforce.com

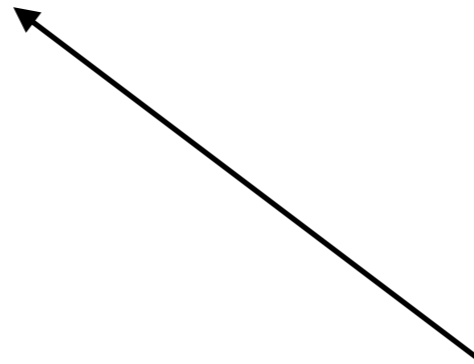


BKI - SEQ2SEQ OUTPUT PROBABILITY

Generation Probability



|target vocabulary|



BKII - COPYNET OUTPUT PROBABILITY

Generation Probability

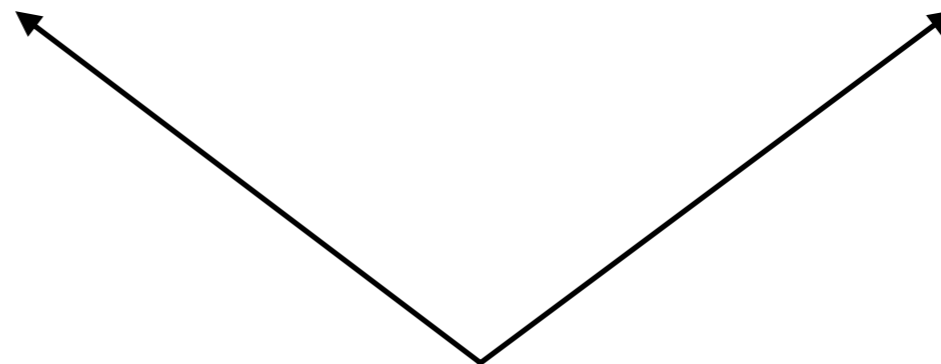


|target vocabulary|

Copy Probability



|source sequence|



BKIII - COPYNET (Gu et. al. 2016)

Generation Probability



|target vocabulary|

Copy Probability



|source sequence|

$$p(y_t | \mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M}) = p(y_t, \mathbf{g} | \mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M})$$

$$+ p(y_t, \mathbf{c} | \mathbf{s}_t, y_{t-1}, \mathbf{c}_t, \mathbf{M})$$

“hidden state”

“copying context”

BKIV- COPYNET (Gu et. al. 2016)

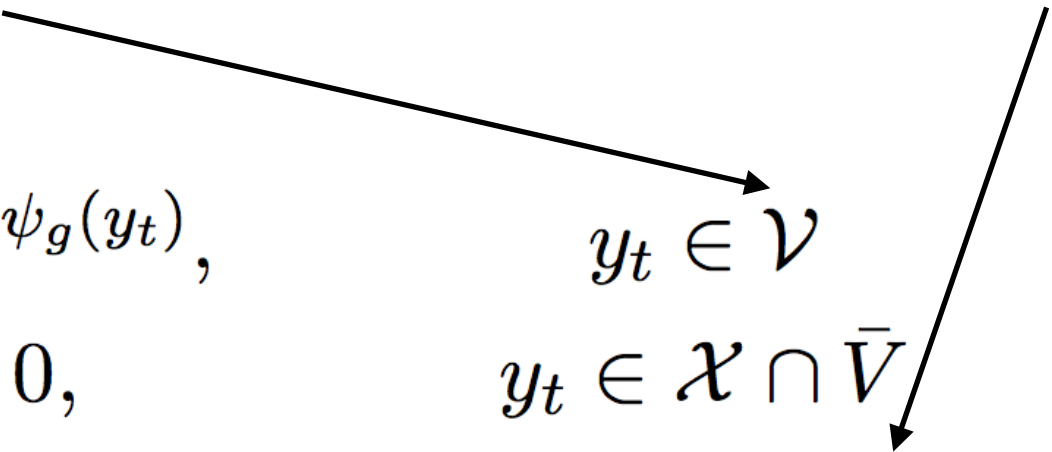
Generation Logit

Copy Logit

softmax ( )

|target vocabulary|

|source sequence|

$$p(y_t, \mathbf{g}|\cdot) = \begin{cases} \frac{1}{Z} e^{\psi_g(y_t)}, & y_t \in \mathcal{V} \\ 0, & y_t \in \mathcal{X} \cap \bar{\mathcal{V}} \\ \frac{1}{Z} e^{\psi_g(\text{UNK})} & y_t \notin \mathcal{V} \cup \mathcal{X} \end{cases}$$
$$p(y_t, \mathbf{c}|\cdot) = \begin{cases} \frac{1}{Z} \sum_{j:x_j=y_t} e^{\psi_c(x_j)}, & y_t \in \mathcal{X} \\ 0 & \text{otherwise} \end{cases}$$


BKV - SPEED-UP EXPERT SOURCING

Command2NL Logout (victoria-lin)

mkdir join set source touch env ln uname which cd awk

chown mount tee more hostname split mktemp od column file

read ssh yes basename less nl rsync zcat rev readlink

shopt paste who fold gzip seq tr whoami comm scp

su tree mv tac jobs pwd ssh-keygen gunzip alias ⁵²head

cat cpio date dig expert chmod dirname history kill ping

sleep top crontab md5sum rmdir awk cut tail cal rename

df diff rm watch ls md5 uniq curl screen ps

chgrp pstree cp nohup sort w bind tar wget apt-get

urls annotated: 21
pairs annotated: 356

Figure 2. Data Collection Interface Screenshot

BKVI - THREE-STAGE TRANSLATION APPROACH

natural language input:

find all log files older than 15 days



**Stage 1: rule-based open-vocabulary
entity recognition**

entity mentions: {filename: "log",
timespan: "15 days"}

natural language template:

find all [filename] files older than [timespan]

**Stage 3: Argument filling and
post-processing**

synthesized program templates:

find . -name "*.log" -mtime
+15d

find . -type f -name "*.log" -mtime
+15d

...

**Stage 2: NL template to program
template translation**