Translating Natural Language to Bash Commands using Deep Neural Networks

Daniel Jenson
djenson@stanford.edu

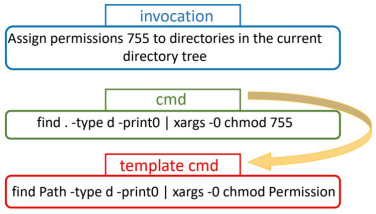
Yingxiao Liu
liuyx@stanford.edu

Introduction

The objective of this project is to generate bash commands from natural language using a deep neural network. Novitiates and even experienced engineers can often find the terminal interface perplexing and are quickly overwhelmed by the syntax of bash commands. This project aims to ease that burden on new and experienced users alike.

Dataset

The dataset we used is from the "The NLC2CMD Competition" consisting of 10,000 parallel translations of English (invocation) and bash (cmd). We further parsed the bash command into the corresponding template form for easier generalization during training.



Evaluation Metric

We used the cross-entropy loss to train the model, but to measure the model performance, we used the metric defined by the competition.

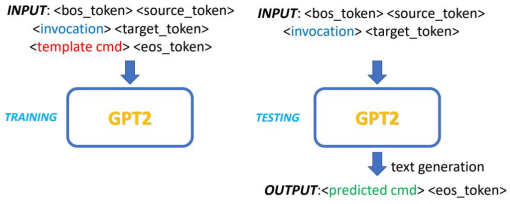
$$S(p) = \sum_{i\in[1,T]} \frac{1}{T} \times \left(\mathbb{I}[U(c)_i = U(C)i] \times \frac{1}{2}\left(1 + \frac{1}{N}(X)\right) - \mathbb{I}[U(c)_i \neq U(C)_i] \right)$$

$U(x)$: a sequence of bash utilities in a command x
c : predicted bash command; C: ground truth bash command
X $= 2 \times |F(U(c)_i) \cap F(U(C)_i)| - |F(U(c)_i) \cup F(U(C)_i)|$
$F(x)$: the set of bash flags in a command x
T : the maximum length between $U(c)$ and $U(C)$
N : the maximum size between $F(c)$ and $F(C)$

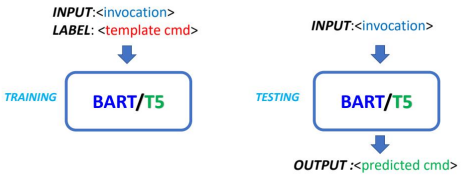
Methods

We experimented with several models, including GPT-2, BART, and T5, as well as different tokenization schemes to improve model performance on the NLC2CMD dataset.

- **Casual Language Modeling: GPT2**



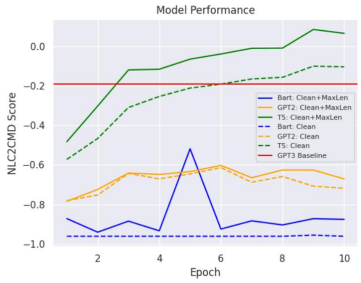
- **Seq2Seq Language Modeling: BART/T5**

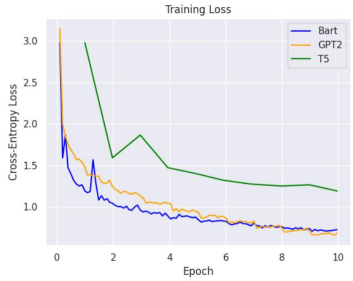


Tokenization mechanics play an enormous role in model performance, especially when the model you are training was originally trained for a different task, like GPT2.

Results & Analysis

We found that T5 performed best for this prediction task. It even outperformed the GPT3 baseline provided by the competition. It also continued to improve with training time. On the other hand, our GPT2 and BART models plateaued rather quickly, and never approached the GPT3 baseline using the NLC2CMD scoring metric.





While cross-entropy loss appears to be the worst for T5, T5 actually performed the best when scoring according to the competition metric. While cross-entropy loss is effective for training, it clearly doesn't parallel performance for our task.