Third Milestone Report: Non-parametric Language Models for Natural Language to Code Generation

Alex Xie
https://axie66.github.io/07400-project/

Mentored by Vincent Hellendoorn
Institute for Software Research

February 16, 2022

1 Major Changes

The only major change since the last milestone is that I plan on moving the kNN experiments to the larger CodeSearchNet dataset [3]. The datasets currently being used, CoNaLa and Concode, are relatively small in size (relative to other datasets used with kNN models), each consisting of about 100k examples and yielding datastores consisting of 2-3 million entries. In comparison, WMT '19, a machine translation dataset used by kNN-MT [2], consists of over 20 million examples and generates a datastore with billions of entries. Hence, we hypothesize that the advantages of kNN retrieval should be more pronounced on CodeSearchNet, which is over twice the size of our current datasets.

2 Progress Report

2.1 Accomplishments

	w/o kNN	w/ kNN	Δ
CoNaLa	36.39	36.74	+0.35
Concode	39.60	39.61	+0.01

Table 1: BLEU scores on Concode and CoNaLa test sets.

As mentioned in my previous milestone report, I have obtained results for the Java Concode dataset, shown in Table 1. Somewhat surprisingly, kNN yields virtually no improvements for Concode (and much less than it did on CoNaLa). This may be due to the increased difficulty of Concode; while the objective in CoNaLa is to generate one-line code snippets, the objective in Concode is to generate full functions in programmatic context. As such, the Concode datastore may not be representative of all the patterns present at test time.

Under the advice of Professor Hellendoorn, I then conducted an analysis of the suitability of the CoNaLa and Concode datasets for kNN retrieval. For CoNaLa in particular, I found that over a third of test examples had few syntactically similar examples (by BLEU score) in the datastore; further, I observed qualitatively that a large number of the neighbors retrieved by kNN were irrelevant to the current context, or only similar in some spurious way. Altogether, these results indicate that CoNaLa and Concode may not be particularly suited for kNN retrieval, necessitating the switch to CodeSearchNet.

2.2 Previous Milestone Goals

While I have obtained Concode results as planned, due to the time spent on analysis of the datasets, I have not yet had the chance to try the more sophisticated kNN approaches detailed in my last milestone; I plan to revisit them in future milestones.

2.3 Surprises

As described earlier, the main surprise is my findings regarding the inadequacy of the CoNaLa and Concode datasets for kNN retrieval.

3 Next Steps

3.1 Looking Ahead

By the next milestone, I plan to have results for kNN code generation on the CodeSearchNet dataset. The base parametric model is currently being trained, though this is taking a while since the dataset is quite large. Depending on the CodeSearchNet results, I may also use CoDesc [1], a very large parallel dataset consisting of over 4.2 million Java functions with documentation.

Further, in parallel with the experiments on CodeSearchNet, I plan on experimenting with the modified kNN formulations described in the last milestone. In particular, an adaptive retrieval mechanism that selectively performs retrieval given the context appears to be promising.

3.2 Revisions to Future Milestones

While the dataset being used has been changed, the overall goals of the project remain the largely same, and no future milestones need to be revised at this time.

References

[1] Masum Hasan, Tanveer Muttaqueen, Abdullah Al Ishtiaq, Kazi Sajeed Mehrab, Md. Mahim Anjum Haque, Tahmid Hasan, Wasi Ahmad, Anindya Iqbal, and Rifat Shahriyar. CoDesc: A large code–description parallel dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 210–218, Online, August 2021. Association for Computational Linguistics.

[2] Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. In *International Conference on Learning Representations (ICLR)*, 2021.

[3] Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, Ge Li, Lidong Zhou, Linjun Shou, Long Zhou, Michele Tufano, Ming Gong, Ming Zhou, Nan Duan, Neel Sundaresan, Shao Kun Deng, Shengyu Fu, and Shujie LIU. CodeXGLUE: A machine learning benchmark dataset for code understanding and generation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.