TECHNISCHE UNIVERSITÄT DARMSTADT

Technische Universität Darmstadt | Hochschulstraße 10 | 64289 Darmstadt

Department of Computer Science and Engineering
University of West Bohemia
Univerzitni 8
306 14 Plzen
Czech Republic

Review of Jan Pašek's Master Thesis

Trustworthy Human Language Technologies

Dr. Ivan Habernal
Research Group Leader

Hochschulstraße 10
64289 Darmstadt

Phone: +49 6151 16 - 21677
ivan.habernal@tu-darmstadt.de
https://www.trusthlt.org/

Date
May 26, 2022

Dear members of the master's exam committee,

Below please find my review of the master thesis by Jan Pašek entitled "Source Code Generation from Descriptions in Natural Language".

The first chapter outlines the research questions of the thesis. The overarching goal is to explore deep neural models that take an input in English and automatically generate a Python code from it.

The second chapter introduces the essential background in language modeling. The description is sufficient to get a grasp about contemporary techniques in NLP and language generation. A minor detail is a potential misunderstanding of generative models. While Jan correctly clarifies the distinction between generative and discriminative machine learning models, most modern neural NLP models, including those for generating language, are in fact purely discriminative; they model the conditional probability of predicting the next token given the encoded input and the previously generated output. In the rest of the chapter, Jan explains the transformer architecture as a particular example of the encoder-decoder paradigm with implementations in GPT, BART, and T5. The theoretical part is finished by explaining sub-word unit tokenization, decoding strategies, and evaluation metrics. Overall, for a reader decently versed in deep learning but unfamiliar with contemporary NLP models, the chapter gives a sufficient introduction.

In chapter three, Jan first introduces various tasks related to NLP tools applied on the domain of source codes. Next, an overview of several existing datsets is presented accompanied by a list of transformer models applied on these data. This chapter concludes the survey part of the thesis.

Chapter four outlines the research questions, i.e., building models capable of generating source codes in Python from their description in English. Jan builds upon the BART architecture and first pre-trains the model in a self-supervised manner on source code denoising.

Collecting the pre-training dataset by scraping GitHub is presented in detail in Chapter five. Jan built a large dataset by utilizing metadata from each repository. Each example, from the machine learning perspective, consists of a single Python function which are extracted by using the built-in abstract syntax tree parser of Python.

Chapter six describes the encoder part of the transformer which were adapted from previously pre-trained encoders CodeBERT and MQDD. For training these models, Jan scraped data from the Stack Overflow platform. The overall results for generating source code from description also shows a newly introduced metric PV that formally checks whether the output code has a valid Python syntax. The results are solid and surpass the state of the art in some setups. One open question that could be addressed in future work is to examine the outputs and errors in a bigger detail than currently discussed at the end of Chapter six.

In summary, Jan addressed all the thesis goals in a thorough manner, compiled new large-scale datasets from scratch, pre-trained and fine-tuned several state-of-the-art model architectures, and evaluated the results. My final grade of his thesis is 'excellent' (*výborně*).

Sincerely,

Dr. Ivan Habernal