FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Review report of a final thesis

Reviewer:	Pierre Donat-Bouillud, Ph.D.
Student:	Rudolf Raevskiy
Thesis title:	Machine Learning Techniques for Source Code Pattern Recognition
Branch / specialization:	Knowledge Engineering
Created on:	6 June 2022

Evaluation criteria

1. Fulfillment of the assignment

[1] assignment fulfilled
▸ **[2] assignment fulfilled with minor objections**
[3] assignment fulfilled with major objections
[4] assignment not fulfilled

This is a serious and extended work, which explores various ML models for source code classification, including graph neural networks and transformers (not Siamese neural networks though) and goes through many input preprocessing techniques, neural network types and architectures. They were implemented and tested. There is a description of the various representations of the source code and how they impact the ML model.
Obstacles, constraints, and recommendations are discussed, but the discussion should be expanded. The thesis aims at presenting which models work better for source code. In the conclusion itself, it is claimed that it was examined "why certain techniques are not suitable for source code or are not longer used". That was mainly undertaken by presenting the quantitative result of ML experiments on an R code dataset, but the reasons why some models perform less well than other ones are not that much explored.

Above all, the thesis should give a better intuition of the impact of the ML model on actual examples, especially in the clustering part. Indeed, the existing recommendations will be best understood by a ML specialist but less by someone not familiar with the field who would like to use the developed tools, such as a PL researcher or an R practitioner.

2. Main written part

80 / 100 (B)

The written part is long, detailed and accurate. It is a bachelor thesis and a first experience of writing a long text so inevitably, the writing could be improved.

The written part is dry and the text sometimes looks a bit like an enumeration of techniques. To improve the flow between paragraphs, between sections and so on, I would

recommend to add more transitions, to add small introductions in each sections and subsections. Using running examples is a great way of creating a better flow, and Section 4.9.4 "Crashtest" is really enjoyable to read, whereas the section about clustering is frustrating, as it does not give any insights about what could be in the clusters. Another example of that issue appears in the introduction, which stated that ML can be used to assess the quality of source code and could be used to detect suboptimal patterns in R. However, in practice, this motivation is not really followed through in the remaining of the written part.

The logical structure is fair but it seems some sections do not belong to their right parts. For instance, some related works are not located in part "Related Work" but in the Current Approach part (for instance, p. 27, Wordpiece and Unigram tokenization techniques). If creating a specific Related Work part, it is preferable to put all of the related works there and to refer back to them later if needed.

The "Theory" part could give a more thorough introduction to neural networks: the "Current Approach" part uses technical words to describe neural networks there, which a non-ML specialist would not understand. For instance, "probability of dropout, add a small warmup for the learning rate and add weight decay" p. 31.

I noticed a recurrent problem with figures: they are usually put there to illustrate but are often not explained. For example, Figure 3.1 in "Related Work" shows a RNN block just to illustrate RNNs but does not explain the elements in the figure. Some figures are actually not referred in the text (e.g. Figure 4.11 in "Current Approach"). There is a problem with Figure 2.5 in "Theory": it is the same as Figure 2.4 and does not show a PDG for sure!

The use of the English language is correct, even though it degrades a bit towards the end of the written part. I noticed few typos. There are also a few problems with the articles (the/a/no article).

Citations are complete and usually well used. There should just be more caution about where to place a citation for a whole enumeration: not after the first item, but at the end of the enumeration, such as in "Methods, such as word2vec [7], Autoencoders, and embedding layers in Dense Neural Networks" p. 9. In that case, [7] should be after Dense Neural Networks if it refers to the enumeration (if it refers just to word2vec, then there are two missing citations for the 2 last items of the enumeration)

3. Non-written part, attachments 90/100 (A)

The code quality is good, the technology used are suitable and adequate. Some documentation on how to use the code and execute it would be helpful and appreciated.

4. Evaluation of results, publication outputs and awards 95/100 (A)

The thesis brings a new dataset of R source codes and a ML model specialized for R, that could be integrated into an IDE for instance.
The results can probably be published soon in an applied ML conference.

The overall evaluation 88 /100 (B)

The technical, coding part - running the ML models and so on - was very well done. The written part should be improved to motivate the work better and make it flow more intuitively. I wish it gave more practical insights and recommendations about the R language, especially in the clustering part.

Questions for the defense

What would you do if you had to cluster R eval calls? Which of the presented algorithms would you use? Do you have an intuition of the clusters you would find, in terms of actual eval class in them?

Instructions

Fulfillment of the assignment

Assess whether the submitted FT defines the objectives sufficiently and in line with the assignment; whether the objectives are formulated correctly and fulfilled sufficiently. In the comment, specify the points of the assignment that have not been met, assess the severity, impact, and, if appropriate, also the cause of the deficiencies. If the assignment differs substantially from the standards for the FT or if the student has developed the FT beyond the assignment, describe the way it got reflected on the quality of the assignment's fulfilment and the way it affected your final evaluation.

Main written part

Evaluate whether the extent of the FT is adequate to its content and scope: are all the parts of the FT contentful and necessary? Next, consider whether the submitted FT is actually correct – are there factual errors or inaccuracies?

Evaluate the logical structure of the FT, the thematic flow between chapters and whether the text is comprehensible to the reader. Assess whether the formal notations in the FT are used correctly. Assess the typographic and language aspects of the FT, follow the Dean's Directive No. 52/2021, Art. 3.

Evaluate whether the relevant sources are properly used, quoted and cited. Verify that all quotes are properly distinguished from the results achieved in the FT, thus, that the citation ethics has not been violated and that the citations are complete and in accordance with citation practices and standards. Finally, evaluate whether the software and other copyrighted works have been used in accordance with their license terms.

Non-written part, attachments

Depending on the nature of the FT, comment on the non-written part of the thesis. For example: SW work – the overall quality of the program. Is the technology used (from the development to deployment) suitable and adequate? HW – functional sample. Evaluate the technology and tools used. Research and experimental work – repeatability of the experiment.

Evaluation of results, publication outputs and awards

Depending on the nature of the thesis, estimate whether the thesis results could be deployed in practice; alternatively, evaluate whether the results of the FT extend the already published/known results or whether they bring in completely new findings.

The overall evaluation

Summarize which of the aspects of the FT affected your grading process the most. The overall grade does not need to be an arithmetic mean (or other value) calculated from the evaluation in the previous criteria. Generally, a well-fulfilled assignment is assessed by grade A.