Replication of Neural Translation Models For Code Generation And Summarization, And Their Evaluation on the CoNaLa Challenge Dataset

Nicolas Chausseau, *Concordia University* nicolas.chausseau@gmail.com, Peter Rigby, *Concordia University* peter.rigby@concordia.ca

WHAT: **Two seq2seq translation models [LeClair 2019, Yin 2018] are replicated for the English-to-code task and the code-to-English task**. We compare the models' BLEU scores (1) on their original study datasets as well as (2) on the CoNaLa challenge dataset [Yin 2018], each time for translation and back-translation (i.e. English-to-code and code-to-English). The complete list of scores is available in the appendix (at the link under *Results*). We show that the Bi-LSTM with attention [Yin 2019] always outperforms the Transformer from LeClair, including on the relatively large (2M pairs) javadoc-based dataset from the original Transformer study [LeClair 2019]. **WHY**: With this work**,** we want to evaluate the state-of-the art on the English-to-code and code-to-English tasks and compare the performance of LSTM and Transformer architectures**. In addition, we provide a method to interpret the significance of a BLEU score result on a certain dataset and test-train split. Since some test-set questions are harder to answer due to a lack of relevant examples in the training set, we provide an adjusted BLEU score reflecting the difficulty of a given test-set question. We find that the "difficulty-adjusted" scores do not significantly affect the relative rankings of the models, but (i) help to shed light on the seq2seq models' capacity to generalize to unseen questions, (ii) helps to measure the degree of difficulty of a dataset and given test-train split and (iii) allows to identify whether some test-set questions are unanswerable**. **HOW**: The replications were done by re-training the models on the different datasets, using the publicly available code, and reporting the best epoch performance, without any change in hyperparameters or preprocessing from the original study. To better interpret results and to provide a reference point, we reported the scores from a third, semantic search model that ranks examples from the training corpus based on their semantic relevance to a given test-set question; this semantic code search model is reproduced from its published description [Sachdev 2018]. Secondly, we provided the theoretical ceiling score for the semantic code search model, by simply searching for the best-scoring train-set answer for each test-set question; we call this score the BLEU Optimal Score (BOS). The BOS is used as a second reference point for each test-set question and is used to provide the "difficulty adjusted score" (a BOS-adjusted score). (RQ1:) Can the seq2seq models surpass the first reference point, NCS, the semantic code search, retrieval-based approach? We find that this is frequently the case, especially on large, noisy datasets; we observe that semantic search tends to perform well on doctstring corpora, but, even on these corpora, is occasionally surpassed by the seq2seq models. (RQ2:) Secondly, can the seq2seq models surpass the second reference point, the BOS, for a given test-set question? We observe that this is rarely the case (< 0.5% of test-set questions), and manually examine results (see discussion below). **WHERE**: The two seq2seq models (Bi-LSTM and Transformer) are evaluated (1) on their original study dataset, to assert original results can be obtained, as well as (2) on the CoNaLa dataset, an English-code parallel corpus extracted from StackOverflow (https://conala-corpus.github.io/), for a fair comparison. For each dataset we first run the seq2seq models (a) on the original task from the replicated study (English-to-code for Neural Code Search, and Code-to-English for the Transformer model from LeClair) and (b) on the reverse task, (code-to-English for Neural Code Search, and English-to-code for the Transformer model from LeClair).

DISCUSSION: As we trained the published English-to-code and code-to-English models, we found that the BLEU scores varied widely from dataset to dataset and were difficult to interpret. Evaluating these models on the same CoNaLa challenge dataset [CoNaLa 2018] allowed us to compare their performance in a more objective way. **In addition, we wanted to provide a reference point for those scores, to help interpret the results. What constitutes a good result for a translation model, given a certain dataset and test-set split?** First, we started using the Neural Code Search (NCS) model (Sachdev 2018) as a first reference point for the scores; because it is a search model it can help to quickly inspect the most relevant training examples for a given test-set question, and inspect their BLEU scores. **As a complement to the Neural Code Search baseline score, we also provide the best-scoring code snippet from the training-set, along with its BLEU score, which we call the BLEU Optimal Score (BOS) for that test-set question**. This BOS score is the theoretical ceiling for the Neural Code Search model on a given dataset and is provided for each test-set question. Interestingly, we observe that this ceiling score for each CoNaLa test-set question is only rarely surpassed by the two other seq2seq neural models. We manually examined the very rare test-set questions on which it is surpassed, and observed that in almost all cases, seq2seq model answers that scored above BOS were not due to the seq2seq model generalizing based on training data, but rather explained by the presence of elements from the answer being already present in the question (that is, the model learned to directly return the question as an answer, instead of learning patterns from other similar answers in the training set). The remaining cases, we show, can be attributed to chance: they are extremely rare (< 0.5% of answers) and are, every time, only marginally higher than the BOS (< 5 percentage point increase). This finding has implications for the study of generalization in LSTM and transformer-based seq2seq based models used for translation. **Finally, we believe that this method can be adopted for any evaluation of seq2seq models, to (i) assess the difficulty of a specific question from the test-set, and (ii) discover what approximate ceiling score to expect from the seq2seq model on that test-set question**.