SMU
SINGAPORE MANAGEMENT UNIVERSITY
School of Computing and Information Systems

Compressing Pre-trained Models of Code into 3 MB

Jieke Shi, Zhou Yang, Bowen Xu, Hong Jin Kang, David Lo

Overview

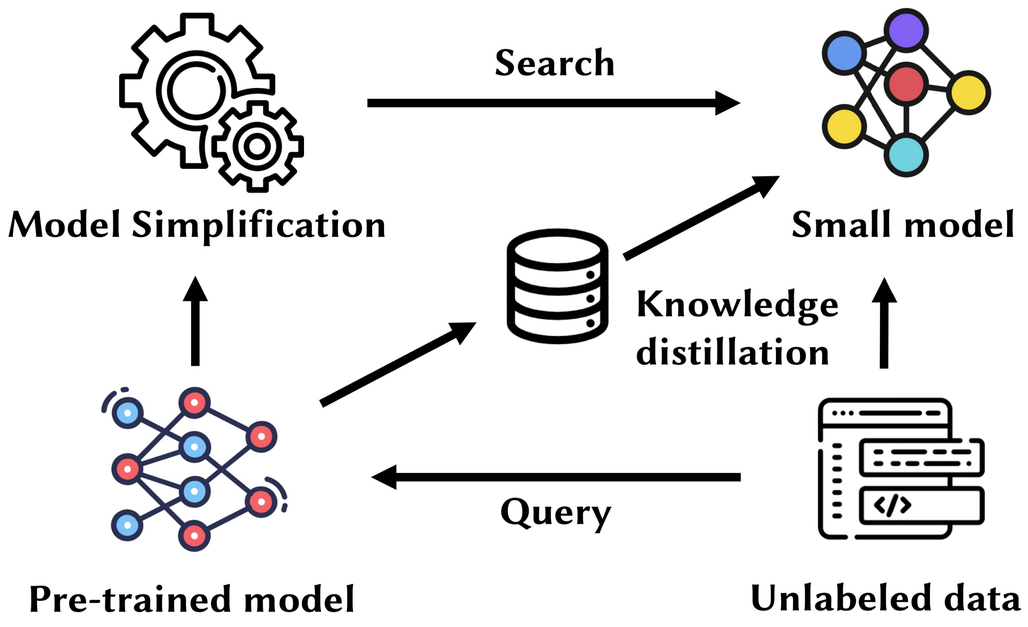
Motivation:

- Pre-trained models of code achieved great success but have huge model sizes and high response latency;
- For modern IDE or editor design, 3 MB model size and 0.1 second latency are preferred modern IDE or editor design [1, 2].

Contribution:

- *Compressor*, a novel compression method via genetic algorithm (GA)-guided model simplification and knowledge distillation.
- Evaluate *Compressor* with CodeBERT and GraphCodeBERT across two downstream tasks. The results validate the effectiveness.

Methodology



The overview of Compressor

Hyperparameter	Pre-trained Models	Search Space
number of network layers	12	[1, 12]
dimensionality of network layers	768	[16, 768]
number of attention heads	12	1, 2, 4, 8
dimensionality of feed-forward layers	3072	[32, 3072]
vocabulary size	50265	[1000, 50000]

Search space of GA-guided model simplification

$$Fitness(s) = GFLOP\,s - |t_s - T|$$

Fitness function of GA-guided model simplification

$$\mathcal{L} = -\frac{1}{n}\sum_{i}^{n} softmax(\frac{p_i}{T}) \log\left(softmax(\frac{q_i}{T})\right) T^2$$

Knowledge distillation loss function

Experiment Results

Model	Vulnerability Prediction		Clone Detection	
	Accuracy (%)	Drop (%)	Accuracy (%)	Drop (%)
CodeBERT (481 MB)	61.82	-	96.20	-
$BiLSTM_{soft}$ (7.5 MB)	57.86	3.96	83.93	12.27
Compressor (3 MB)	**59.44 (96.15%)**	**2.38 (-39.90%)**	**95.43 (99.20%)**	**0.77 (-93.72%)**
GraphCodeBERT (481 MB)	61.38	-	96.62	-
$BiLSTM_{soft}$ (7.5 MB)	58.02	3.36	84.08	12.54
Compressor (3 MB)	**59.99 (97.74%)**	**1.39 (-58.63%)**	**94.22 (97.52%)**	**2.4 (-80.86%)**
Average Maintained Accuracy/Improvements	**96.95%**	**-49.27%**	**98.36%**	**-87.29%**

RQ1: Can *Compressor* result in small accuracy loss when extremely compressing the pre-trained model?
Answer: Compressed models maintain 96.95% and 98.36% of the original performance on the two tasks.

Model	Vulnerability Prediction	Clone Detection
	Latency (ms)	
CodeBERT (481 MB)	1507	2675
Compressor (3 MB)	**347 (-76.97%)**	**625 (-76.64%)**
GraphCodeBERT (481 MB)	1209	1788
Compressor (3 MB)	**429 (-64.52%)**	**326 (-81.77%)**
Average Improvements	**-70.75%**	**-79.21%**

RQ3: How fast is *Compressor* in compressing pre-trained models?
Answer: *Compressor* only incurs 30.39% and 37.06% additional time to the fine-tuning time, which we believe to be reasonable.

Stage	Vulnerability Prediction	Clone Detection
	Time Cost (min)	
CodeBERT Fine-tuning	49	124
Compressor (3 MB)	13 (26.53%)	47 (37.90%)
GraphCodeBERT Fine-tuning	73	243
Compressor (3 MB)	25 (34.25%)	88 (36.21%)
Average Improvements	**30.39%**	**37.06%**

RQ2: How much efficiency improvement can the compressed models obtain?
Answer: Compressed CodeBERT and GraphCodeBERT are 4.31× and 4.15× faster than the original model at inference.

Reference

[1] Alexey Svyatkovskiy, Sebastian Lee, et al. "Fast and memory-efficient neural code completion." MSR 2021.
[2] Gareth Ari Aye, and Gail E. Kaiser. “Sequence model design for code completion in the modern IDE.” arXiv, 2020.

Artifacts:

Preprint:

Our Group:

SOAR
SOFTWARE ANALYTICS RESEARCH GROUP