

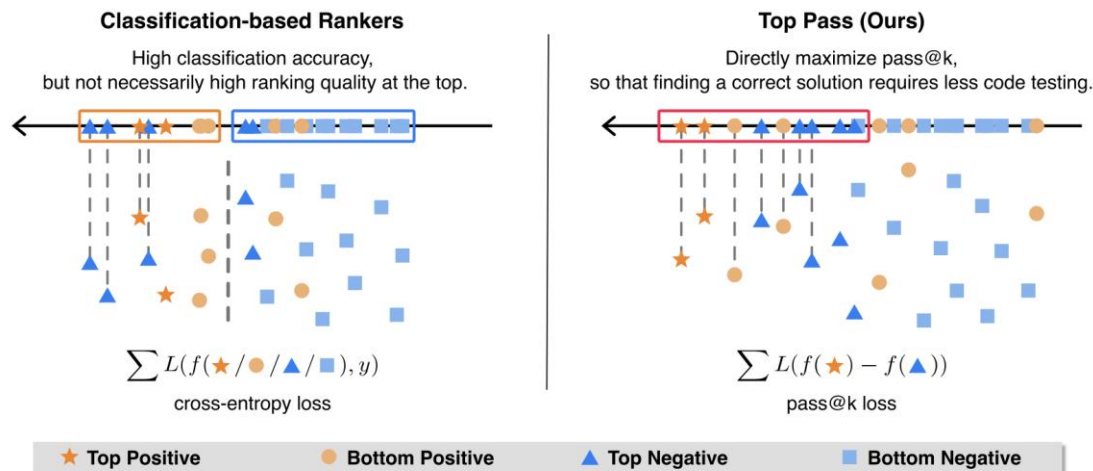
Top Pass: Improve Code Generation by Pass@k-Maximized Code Ranking

Zhi-Cun LYU, Xin-Ye LI, Zheng XIE, Ming LI

Frontiers of Computer Science, DOI: [10.1007/s11704-024-40415-9](https://doi.org/10.1007/s11704-024-40415-9)

Problems & Ideas

- Problems of code generation:
 - LLM-based code generation approaches struggle to generate error-free code in a few tries. Ensuring the correct solution within a few candidates remains challenging.
 - Current rankers model the code ranking as binary classification and ignores the goal is ranking any positive sample to the top.
- Ideas: We propose to optimize the metric ‘pass@k’ directly, which emphasizes that ranker model should rank the first positive program before the k-th negative program.



Top Pass minimizes a novel pass@k loss function that enhances the ranking quality at the top of the code candidate list, so that the user can solve the programming task with fewer attempts.

Main Contributions

- Contributions:
 - We claim that optimizing pass@k is beneficial for code ranking in generation systems. Unlike previous studies which focus on leveraging additional information for classification, we emphasize the ranking nature of the problem;
 - We propose Top Pass, a novel approach for ranking generated code candidates based on their probabilities of correctness;
 - Experimental results demonstrate that our approach, Top Pass, substantially enhances the utility of current code generation models. by improving the pass@k performance.

Category	Method	Pass@1	Pass@2	Pass@3	Pass@5	Pass@10
Standalone LLM	PG-TD	0.7	1.1	–	–	2.5
	Codex	0.7	1.2	–	–	3.0
	WizardCoder	2.0	–	–	3.3	–
	WizardCoder + CodeChain	2.5	–	–	3.3	–
	ChatGPT	2.9	4.8	6.3	8.6	12.1
	DeepSeek-Coder	5.2	7.9	9.8	12.5	16.4
LLM with testcases	CodeT	2.1	2.3	–	–	5.3
	ALGO	5.6	5.6	–	–	7.7
LLM with ranker	CodeRanker w/ ChatGPT	6.1	9.1	9.7	10.3	12.7
	Top Pass w/ ChatGPT	7.3 ^{+19.7%}	10.3 ^{+13.2%}	12.7 ^{+30.9%}	13.3 ^{+29.1%}	15.2 ^{+19.7%}
	CodeRanker w/ DeepSeek-Coder	7.3	9.1	10.3	13.3	17.0
	Top Pass w/ DeepSeek-Coder	9.7 ^{+32.9%}	12.7 ^{+39.6%}	13.3 ^{+29.1%}	14.5 ^{+9.0%}	18.2 ^{+7.1%}

