Could we spend a week optimising?

155 trillion

tokens served / cluster / day

DeepSeek V4.1-Flash · 20 Oct 2026

A hypothetical deployment

10,000 GPUs · 90 days

One week of tuning	0.1% speedup
8 GPUs × 7 days	1.001× faster
56 GPU-days	**≈900 GPU-days**

TENSAT's main configuration

Time (s)	Exploration	Extraction
NasRNN	0.10	0.37
BERT	0.83	0.60
ResNeXt-50	0.68	0.05
NasNet-A	8.81	1.79
SqueezeNet	0.16	0.12
VGG-19	0.42	0.02
Inception-v3	4.38	0.75

Table 3. Optimization time breakdown for TENSAT.

More search: what changes?

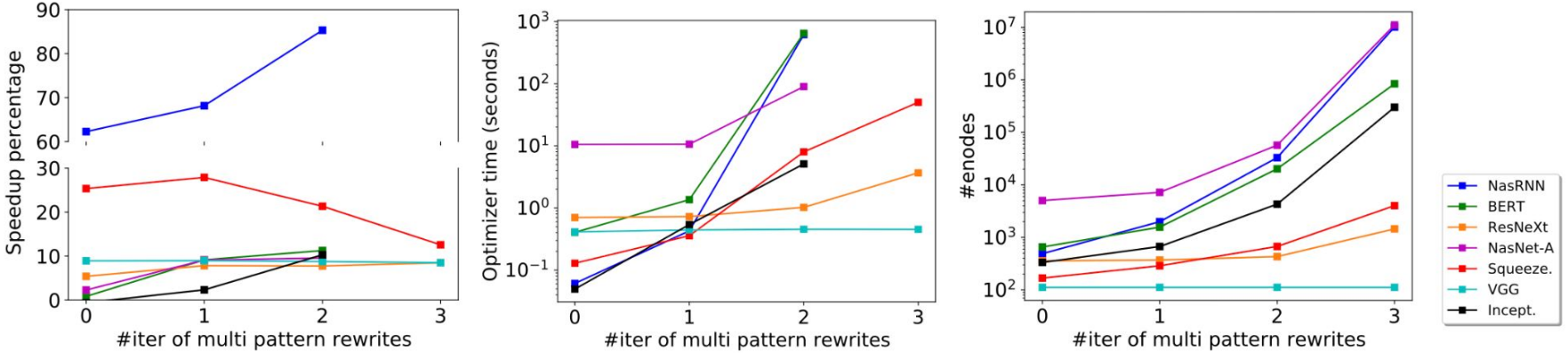


Figure 7. Effect of varying the number of iterations of multi-pattern rewrites k_{multi}. For BERT, NasNet-A, NasRNN, Inception-v3, the ILP solver times out at one hour for $k_{\text{multi}} = 3$. Left: speedup of the optimized graphs (the y-axis is split for clarity). Middle: time taken by TENSAT. Right: final e-graph size (number of e-nodes). The middle and right figures are in log scale.

A hypothetical choice

Graph	Predicted	Measured
A	8 ms	8 ms
B	7 ms	9 ms

Optimal — according to what?

Can extraction stay cheap?

Extraction time (s)	k_{multi}	**With cycle** real	**With cycle** int	**Without cycle**
BERT	1	0.96	0.98	**0.16**
	2	>3600	>3600	**510.3**
NasRNN	1	1116	1137	**0.32**
	2	>3600	>3600	**356.7**
NasNet-A	1	424	438	**1.81**
	2	>3600	>3600	**75.1**

Table 5. Effect of whether or not to include cycle constraints in ILP on extraction time (in seconds), on BERT, NasRNN, and NasNet-A. For the cycle constraints, we compare both using real variables and using integer variables for the topological order variables t_m.

Where would you spend your time next when you have to further optimize TENSAT?

Explore

Extract

Measure

Where would you spend your time next when you have to further optimize TENSAT?

Explore

Extract

Measure

When should we optimise again?