GraSS: Scalable Data Attribution with Gradient Sparsification and Sparse Projection

Pingbang Hu[1] Joseph Melkonian[2] Weijing Tang[3] Han Zhao[1] Jiaqi W. Ma[1]
[1]University of Illinois Urbana-Champaign [2]Womp Labs [3]Carnegie Mellon University

Background: What is Data Attribution?

Given a dataset $D = \{z_i\}_{i=1}^n$ parametrized by a weight $w \in \mathbb{R}^n$, the corresponding model is trained via ERM $\mathcal{A}$ as:

$$\hat{\theta}_w = \mathcal{A}(w) := \arg\min_{\theta \in \mathbb{R}^p} \sum_{i=1}^n w_i \ell_i, \quad \ell_i := \ell(z_i; \theta).$$

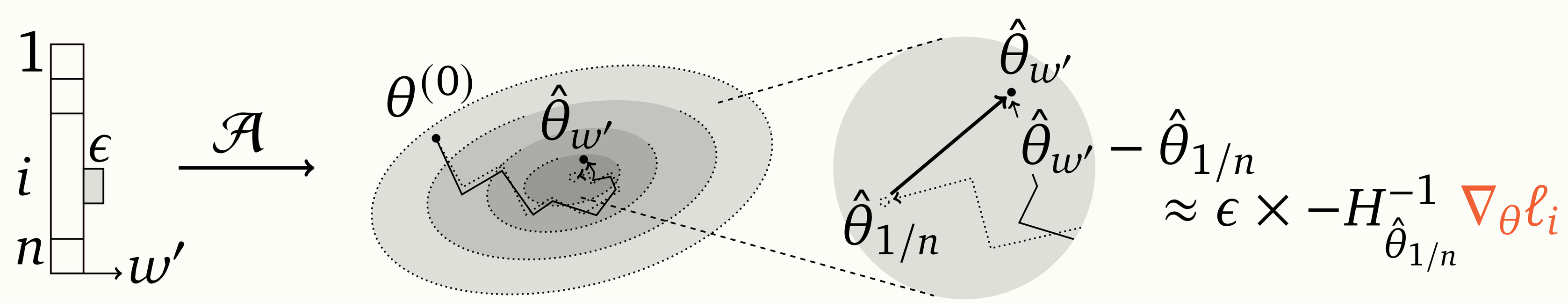
Default weight is $w = 1/n \in \mathbb{R}^p$, and we will first train $\hat{\theta}_{1/n}$.

Data attribution quantifies the **counterfactual effect** for *dataset perturbation* when w becomes w'. The key is to estimate $\hat{\theta}_{w'} - \hat{\theta}_w$.

Motivation: Gradient-Based Data Attribution

Most popular data attribution methods are gradient-based:

Intuition. Taylor-expand $\hat{\theta}_w$ around the default weight $1/n$ [5]:



Problem: Computing $H_{\hat{\theta}}^{-1} \nabla_\theta \ell_i$ is expensive, due to the size...

1. **Compress** $g_i := \nabla_\theta \ell_i$ from $\mathbb{R}^p$ to $\hat{g}_i \in \mathbb{R}^k$ with $k \ll p$!
2. **Replace** $H_{\hat{\theta}}$ with *Fisher Information Matrix* $\frac{1}{n}\sum_{i=1}^n \hat{g}_i \hat{g}_i^\top \in \mathbb{R}^{k \times k}$.

These two tricks, although effective, comes with costs.

Existing Approaches: Compression incurs a large overhead!

- Gaussian/Rademacher: $P g_i = \hat{g}_i$, $O(pk)$ per projection.
- SOTA (FJLT): $\widetilde{O}(p)$ per projection.
- SOTA (LoGra): $O(\sqrt{pk})$ per projection for **linear layers**.

Contributions

We design two *sub-linear* gradient compression algorithms:

1. GraSS: $O(k')$ per projection with $k < k' \ll p$.
2. FactGraSS: $O(k')$ but *without materializing* g_i for **linear layers**!

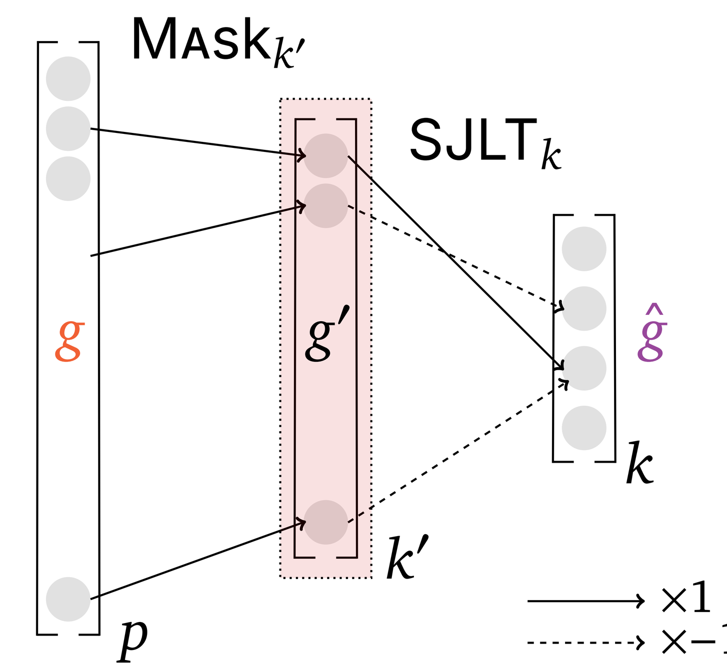
GraSS: Gradient Sparsification and Sparse Projection

GraSS compresses $g \in \mathbb{R}^p$ to $\hat{g} \in \mathbb{R}^k$ in $O(k')$ where $k < k' \ll p$:

Mask$_{k'}$. Sparsification:
- Select few parameters from g
- $\Rightarrow$ *Sub-linear* complexity!

\+

SJLT$_k$. Sparse projection:
- Sparsify projection matrix P
- $\Rightarrow$ *Linear* complexity!

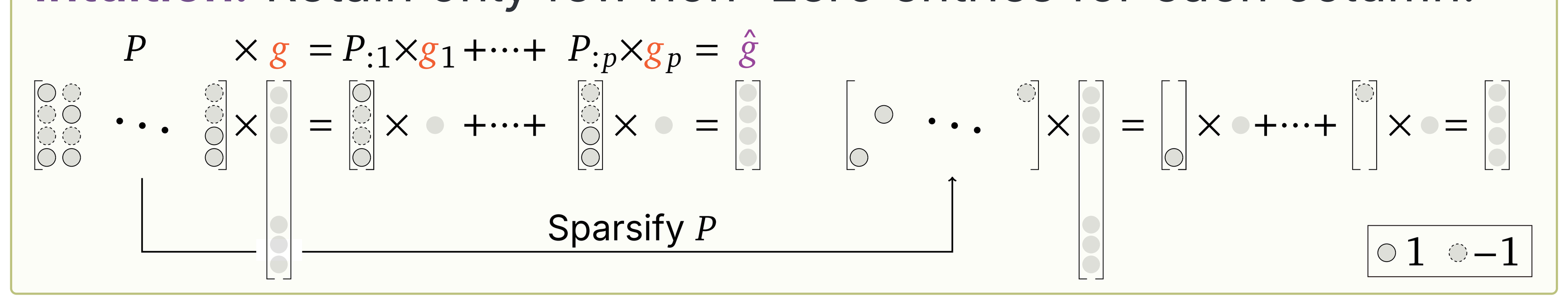


Mask is well-explored in the literature:

Example. Lottery Ticket Hypothesis [3], Localize [4], etc.

Sparse Johnson-Lindenstrauss transform [2] is also famous:

Intuition. Retain only few non-zero entries for each column!



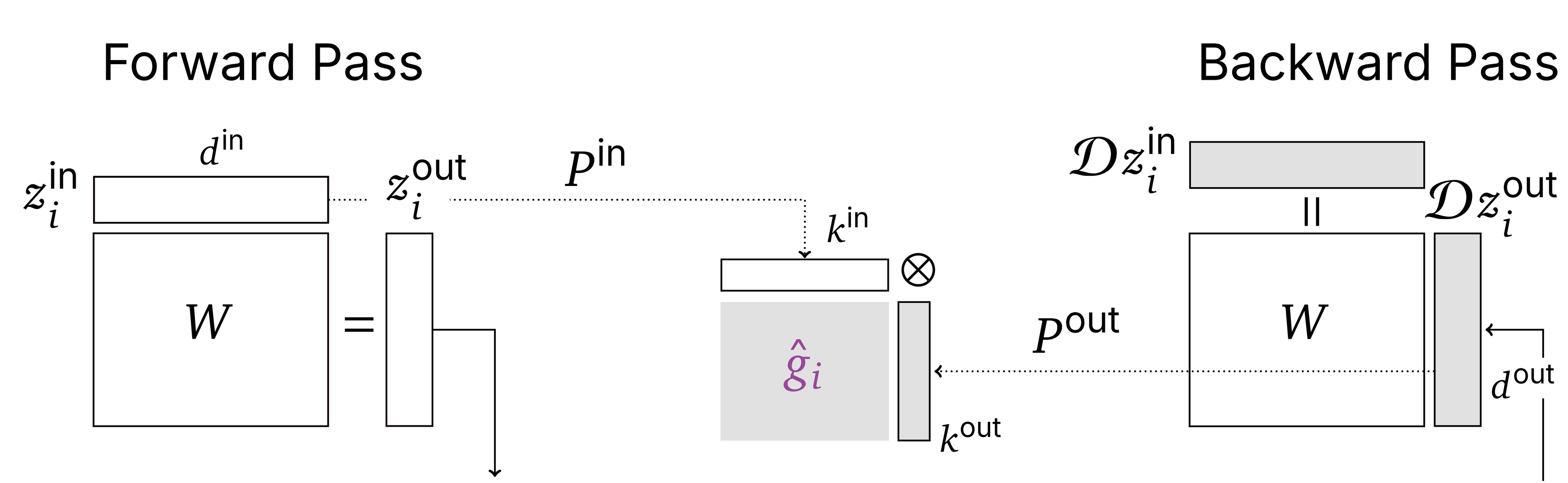
Problem of GraSS: Gradient Materialization

GraSS is already fast. But it requires materializing g.

Q: Is this even a concern? **A:** Sadly, yes... Consider linear layers:

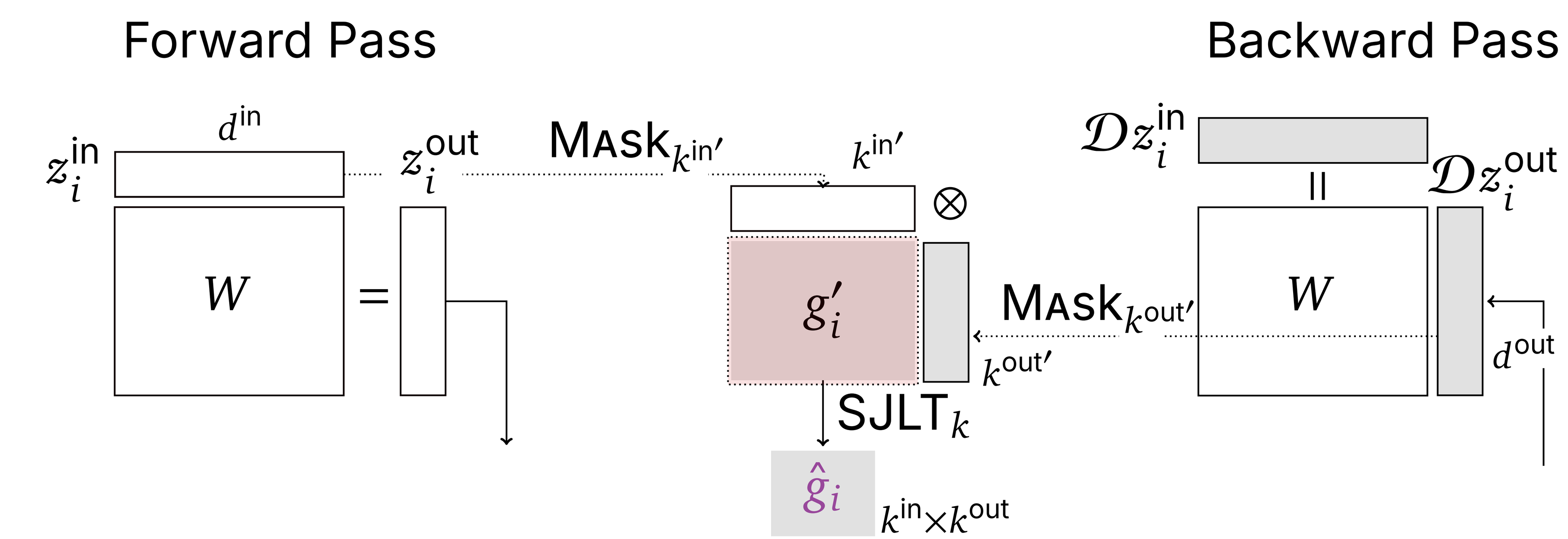
$$g_i = \frac{\partial \ell_i}{\partial W} = \frac{\partial \ell_i}{\partial z_i^{\text{out}}} \frac{\partial z_i^{\text{out}}}{\partial W} = z_i^{\text{in}} \otimes \frac{\partial \ell_i}{\partial z_i^{\text{out}}}$$

Previous SOTA gradient compression, LoGra [1], exploits this:



FactGraSS: Factorized GraSS—New SOTA

GraSS can also exploit this structure cleverly!

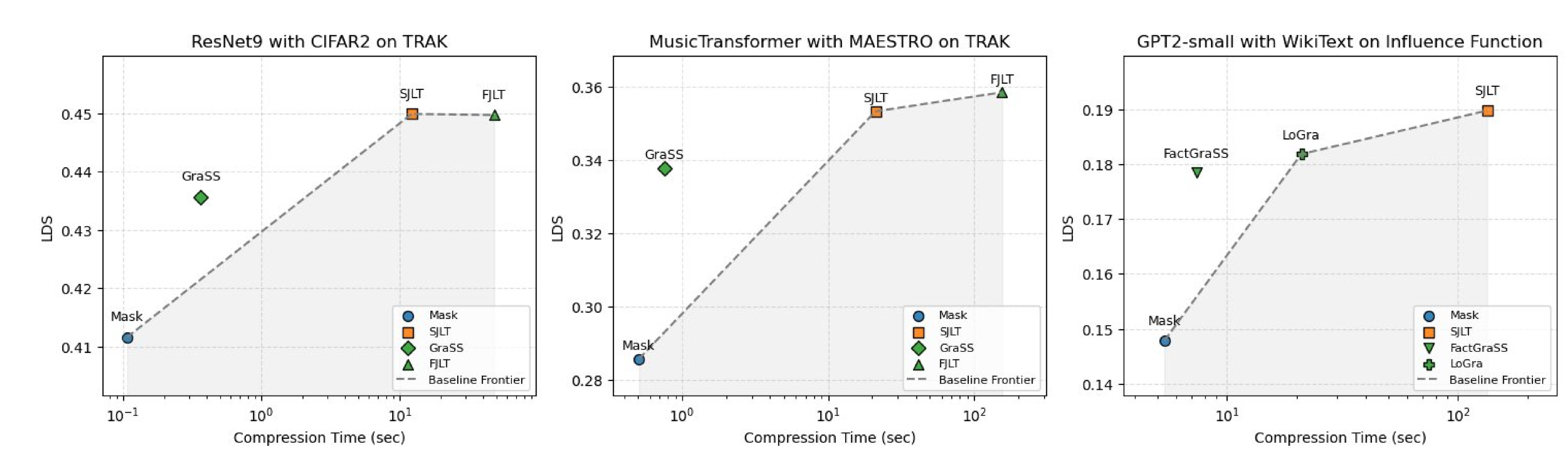


- **Intuition**: (1) **Factorized** Mask $\Rightarrow$ (2) Reconstruct $\Rightarrow$ (3) SJLT!
- **Bottlenecks**: SJLT's input size, $k' := k^{\text{in}'} \times k^{\text{out}'}$

Theorem. There is a *sub-linear* compression algorithm with complexity $O(k')$ where $k < k' \ll P$. Moreover, this extends to **linear layers**, where full gradients are **never** materialized.

Experimental Results

GraSS establishes new SOTA, pushing the Pareto frontier!



Billion Scale. FactGraSS achieves 160% speedup (72684 v.s. 27255 tokens/sec) on Llama-**8B**-Instruct.

[1] Choe et al. What Is Your Data Worth to GPT? LLM-Scale Data Valuation with Influence Functions. May 22, 2024. arXiv: 2405.13954 [cs].
[2] Dasgupta, Kumar, and Sarlós. A sparse johnson: Lindenstrauss transform. 2010.
[3] Frankle and Carbin. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks.
[4] He et al. Localize-and-Stitch: Efficient Model Merging via Sparse Task Arithmetic. *Transactions on Machine Learning Research* (2025). ISSN: 2835-8856. URL: https://openreview.net/forum?id=9CWU8Oi86d.
[5] Koh and Liang. Understanding black-box predictions via influence functions. PMLR. 2017.