CAPACITY AND REDUNDANCY TRADE-OFFS IN MULTI-TASK LEARNING

Asif Khan
Harvard Medical School, USA
asif.khan@hms.harvard.edu

ABSTRACT

In multi-task learning (MTL) negative transfer is often considered as an optimization artifact. In this paper, we frame it as a consequence of limited shared capacity and weak task redundancy. Using Capacity–Redundancy identity we decompose the sum of per-task predictive informations into joint predictive information that includes label redundancy defined via total correlation, and a residual coupling term that quantifies interference left unresolved by the shared representation. Additionally, we show two key results: (i) a clustering-gap decomposition that gives a necessary and sufficient condition for clustered sharing to outperform global sharing, and (ii) a gradient–TC bridge in a Gaussian multi-task model that formally justifies gradient cosine similarity as a proxy for redundancy ordering. Empirically, we estimate Δ from validation residual correlations, showing that clustered LoRA substantially reduces $\widehat{\Delta}$, outperforms size-matched random partitions, and results in statistically significant gains with multi-seed confidence intervals.

1 INTRODUCTION

Classical multi task learning (MTL) includes a shared encoder with task-specific output heads, which allows information to be transferred across tasks while retaining task specialization (Caruana, 1997; Ruder, 2017). Such a training approach has been successfully applied in several areas including computer vision (CV) (Zhang et al., 2014; Misra et al., 2016), natural language processing (NLP) (Collobert and Weston, 2008), and speech recognition (Deng et al., 2013). When multiple tasks share a single encoder, they compete for representational capacity which is a main cause of negative transfer (when training on additional tasks degrades rather than improving performance) (Wang et al., 2019).

Recent developments in large pre-trained language models (LLMs) have renewed interest in MTL in the context of parameter-efficient fine-tuning (PEFT). Methods such as adapters (Houlsby et al., 2019), prefix-tuning (Li and Liang, 2021; Ding et al., 2023), and low-rank adaptation (LoRA) (Hu et al., 2022) make it possible to adapt massive models to new tasks with only a small number of additional parameters. A design choice here is to decide when multiple tasks should share the same lightweight module encouraging cross-task transfer, or should each task be assigned its own module to avoid interference? While previous work has explored both ends, as well as a range of hybrid sharing approaches (He et al., 2021; Karimi Mahabadi et al., 2021; Mahabadi et al., 2021), our focus is on a theoretical framework that can be used to decide when to share and when to specialize model parameters.

When tasks are heterogeneous they compete for a restricted shared subspace as a result gradient updates for one task can degrade performance across another (Yu et al., 2020; Chen et al., 2018). By giving each task its own private features one can avoid this interference but can reduce efficiency. Hybrid approaches such as task clustering, mixture-of-adapters, or routing using task embeddings (Gururangan et al., 2021; Mudrakarta et al., 2018), attempt to balance these trade-offs but lack a unifying theoretical justification.

In this paper, we treat the shared encoder in MTL as a finite-capacity channel. Under the given budget $I(X; Z_s) \leq C_s$ on the shared latent Z_s, we prove a CR inequality that shows the total predictive information a single shared latent can provide across tasks is bounded by its capacity plus the label

redundancy. Thus, correlations let the same bits be reused across tasks, while weakly related tasks force competition for capacity and make negative transfer unavoidable. We extend the bound to shared–private representations with per-task budgets and derive conditional variants together with a Bayes-error lower bound that formalize when performance trade-offs cannot be avoided. Finally, we model LoRA as a capacity-constrained channel and show how the CR perspective explains how similarity-based sharing helps when task redundancy is high, whereas low-redundancy task sets require growing effective adapter capacity (rank) or private routes to prevent interference.

2 Related Work

Classical multi-task learning. Classical MTL use shared-backbone, task-specific heads, structured regularizers, and cross style mechanisms to exploit relatedness while preserving specialization (Caruana, 1997; Collobert and Weston, 2008; Zhang et al., 2014; Misra et al., 2016). Beyond deep nets, a large body of convex/regularization approaches formalizes parameter sharing via matrix norms or task relationships (Evgeniou and Pontil, 2004; Ando et al., 2005; Argyriou et al., 2008). We refer readers to Ruder (2017) for a survey of MTL.

Parameter-efficient fine-tuning and adapters. Adapters, prefix or prompt tuning, and LoRA are used to adapt large frozen backbones using a small added budget(Houlsby et al., 2019; Li and Liang, 2021; Ding et al., 2023; He et al., 2021). Multi-task PEFT variants share adapters globally, allocate them per task or combine both through routing, hypernetworks, or mixtures (Mahabadi et al., 2021; Karimi Mahabadi et al., 2021; Mudrakarta et al., 2018; Gururangan et al., 2021). However, in practice these choices remain largely heuristic.

Empirical analyses of task relatedness and negative transfer. Standley et al. (2020) investigate which tasks benefit from being trained together and which cause conflicts, while Zamir et al. (2018) chart the transferability between tasks through a large-scale task graph. Gradient-based analyses of task conflict and mitigation are widely used in practice (Chen et al., 2018; Kendall et al., 2018; Wang et al., 2019; Yu et al., 2020; Chai et al., 2023). These methods typically operate at the level of optimization dynamics rather than at the level of distributional limits.

Theoretical generalization and representation sharing. Baxter (2000) formalized task families and bias learning; Evgeniou and Pontil (2004) analyzed kernelized regularization for MTL; and Maurer et al. (2016) show the benefits of learning a shared representation with task-averaged Rademacher complexity bounds. Several works explicitly partition representations or allocate small task-specific heads to mitigate conflicts (Newell et al., 2020; Wang et al., 2020; Lin et al., 2020; Momma et al., 2022). These approaches provide generalization guarantees as functions of task similarity and hypothesis class capacity. In contrast, our focus is on a distribution-level converse on the achievable predictive information through a shared latent, independent of any estimator.

Multi-task information bottleneck. Multi-task variational information bottleneck (IB) $I(Z;Y) - \beta I(Z;X)$ formulations optimize multi-task IB objectives ($\sum_t I(Z;Y^{(t)})$) directly (Qian et al., 2020), offering algorithmic objectives but not converses (Tishby et al., 2000; Alemi et al., 2016). Classical results used in our analysis include the independence bound on entropy (Cover, 1999) and its conditional form and the data processing inequality (Cover, 1999), total correlation (Watanabe, 1960) as a redundancy measure, and subset entropy inequalities (Sun, 1975; Madiman and Tetali, 2010). Our results characterize when sharing is fundamentally limited and when structured sharing (clustering/private capacity) is provably beneficial, independent of a particular training algorithm.

3 Capacity–Redundancy Inequality

Let T be the number of tasks with labels $Y^{(1)}, \dots, Y^{(T)}$ and input X. A shared representation $Z_s = f(X)$ is used by all tasks. Throughout this section we assume the standard representation setting in which $Y^{(1:T)} - X - Z_s$ forms a Markov chain (equivalently, $I(Z_s; Y^{(1:T)} \mid X) = 0$), and we impose the shared capacity budget $I(Z_s; X) \le C_s$. We use total correlation (TC) as a measure of label redundancy, $TC(Y^{(1:T)}) = \sum_{t=1}^{T} H\big(Y^{(t)}\big) - H\big(Y^{(1:T)}\big) \ge 0$. $TC(Y^{(1:T)}) = 0$ iff the labels are mutually independent. The sum of per-task information carried by a shared bottleneck is always bracketed by the joint information and the joint information plus redundancy. Here, we present the key results and refer readers to the Appendix for detailed proofs and preliminaries.

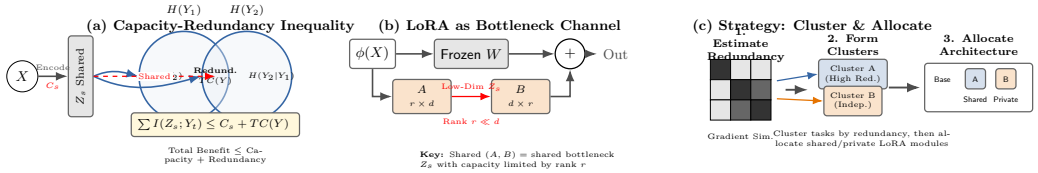


Figure 1: **CR Tradeoff in Multi-Task Low-Rank Adaptation.** **(a)** A two task example of the CR intuition. The overlap between Y_1 and Y_2 given by $TC(Y_1, Y_2) = I(Y_1, Y_2)$ represents redundancy that allows shared bits to be reused across tasks. The shared bottleneck Z_s has capacity C_s, and the blue arrows denote task-specific predictive information $I(Z_s; Y_t)$. **(b)** LpRA as a capacity constrained channel, rank-r matrices (A, B) add to frozen weights W, and the rank $r \ll d$ restricts the effective shared capacity C_s. Sharing (A, B) across tasks creates the shared bottleneck. **(c)** CR guided allocation, estimate redundancy from gradient similarity, cluster related tasks by redundancy, and allocate shared LoRA modules within clusters and private modules across weakly related tasks

Lemma 3.1 (Chain-rule sandwich (Watanabe, 1960; Madiman and Tetali, 2010))**.** *For random variables* $(Z, Y^{(1)}, \ldots, Y^{(T)})$,

$$I\Big(Z; Y^{(1:T)}\Big) \;\le\; \sum_{t=1}^{T} I\Big(Z; Y^{(t)}\Big) \;\le\; I\Big(Z; Y^{(1:T)}\Big) + TC\Big(Y^{(1:T)}\Big). \tag{1}$$

Next, we combine the sandwich with the Markov chain and the capacity budget to define CR inequality.

Theorem 3.2 (Capacity–Redundancy (CR) inequality)**.** *Under the Markov chain and capacity budget following bound holds,*

$$\sum_{t=1}^{T} I\Big(Z_s; Y^{(t)}\Big) \;\le\; C_s + TC\Big(Y^{(1:T)}\Big). \tag{2}$$

and the canonical sandwich of Lemma 3.1 holds with $Z = Z_s$,

$$I\Big(Z_s; Y^{(1:T)}\Big) \;\le\; \sum_{t=1}^{T} I\Big(Z_s; Y^{(t)}\Big) \;\le\; I\Big(Z_s; Y^{(1:T)}\Big) + TC\Big(Y^{(1:T)}\Big). \tag{3}$$

The upper bound Equation (2) is induced by an exact identity,

$$\sum_{t=1}^{T} I\Big(Z_s; Y^{(t)}\Big) = I\Big(Z_s; Y^{(1:T)}\Big) + \underbrace{\Big(TC(Y^{(1:T)}) - TC(Y^{(1:T)} \mid Z_s)\Big)}_{\text{dependence explained by } Z_s}.$$

which is equivalent to $\sum_{t=1}^{T} I\big(Z_s; Y^{(t)}\big) = I\big(Z_s; Y^{(1:T)}\big) + TC(Y^{(1:T)}) - \Delta$, where $\Delta = TC(Y^{(1:T)} \mid Z_s) \ge 0$.

Δ is the residual multi-task coupling that remains after observing Z_s, $\Delta = 0$ iff the labels become conditionally independent given Z_s. When Δ is large, the right inequality in Equation (3) can be loose, signaling that the shared representation has not factorized the task joint distribution, and that additional shared or task-specific capacity (Section 5) is required. Moreover, when $(TC(Y^{(1:T)}))$ is large and $(TC(Y^{(1:T)} \mid Z_s))$ is small, the shared representation captures cross-task dependence, so the same shared bits can be reused efficiently across tasks rather than competed over.

The tightness of Equation (2) depends on conditional independence of tasks given Z_s $\Delta = TC(Y^{(1:T)} \mid Z_s) \approx 0$, and capacity saturation $I(Z_s; Y^{(1:T)}) \approx I(Z_s; X) \approx C_s$. In a linear–Gaussian setting dependence is explained by Z_s spanning the label-relevant subspace so that cross-task coupling disappears after conditioning, while saturation corresponds to operating at the channel capacity of the bottleneck.

4 Consequences and Impossibility Results

We use the CR inequality to show that if labels aren't redundant, per-task performance must degrade as the number of tasks grows unless capacity scales. CR inequality implies that the total per-task information extractable from a shared bottleneck cannot exceed shared capacity plus label redundancy $\sum_{t=1}^{T} I(Z_s; Y^{(t)}) \leq C_s + TC(Y^{(1:T)})$.

Corollary 4.1 (Average per-task information)**.** *Under the assumptions of Theorem 3.2,*

$$\frac{1}{T}\sum_{t=1}^{T} I(Z_s; Y^{(t)}) \leq \frac{C_s}{T} + \frac{TC(Y^{(1:T)})}{T}. \tag{4}$$

In particular, there exists at least one task $t^\star$ such that $I(Z_s; Y^{(t^\star)}) \leq \frac{C_s + TC(Y^{(1:T)})}{T}$.

When tasks are weakly dependent, $TC(Y^{(1:T)})$ does not grow with T, and Equation (4) forces the average per-task information to decay like $O(1/T)$ for fixed C_s.

Corollary 4.2 (Linear capacity is necessary for maintaining per-task signal)**.** *Let $TC(Y^{(1:T)}) \leq \tau$ for all T, now for a uniform per-task information level $I(Z_s; Y^{(t)}) \geq b$ for all $t \in [T]$, then necessarily $C_s \geq Tb - \tau$. In particular, for independent tasks ($TC = 0$), maintaining $I(Z_s; Y^{(t)}) \geq b$ for all t forces $C_s \geq Tb$.*

Corollary 4.2 tells us when negative transfer is unavoidable. If new tasks are added without increasing shared capacity, then some tasks must lose predictive signal through the shared bottleneck unless the new tasks are sufficiently redundant with the existing ones. Equivalently, under weak redundancy, maintaining a fixed per-task information level requires shared capacity to grow approximately linearly with the number of tasks. We next connect this phenomenon to standard prediction losses.

Log-loss lower bound. For each task t, let $q_t(\cdot \mid Z_s)$ denote any predictor and consider the log-loss $\ell_t = -\log q_t(Y^{(t)} \mid Z_s)$. Then by the Gibbs inequality,

$$\mathbb{E}[\ell_t] \geq H(Y^{(t)} \mid Z_s) = H(Y^{(t)}) - I(Z_s; Y^{(t)}). \tag{5}$$

Theorem 4.3 (Total log-loss lower bound with residual coupling)**.** *Under the assumptions of Theorem 3.2, for any collection of per-task predictors $\{q_t\}_{t=1}^{T}$,*

$$\sum_{t=1}^{T} \mathbb{E}[-\log q_t(Y^{(t)} \mid Z_s)] \geq \sum_{t=1}^{T} H(Y^{(t)} \mid Z_s) = H(Y^{(1:T)} \mid Z_s) + \Delta, \tag{6}$$

where $\Delta = TC(Y^{(1:T)} \mid Z_s) \geq 0$. Moreover, using $I(Z_s; Y^{(1:T)}) \leq I(Z_s; X) \leq C_s$,

$$\sum_{t=1}^{T} \mathbb{E}[-\log q_t(Y^{(t)} \mid Z_s)] \geq H(Y^{(1:T)}) - C_s + \Delta. \tag{7}$$

The lower bound in Equation (6) implies that $\Delta > 0$ forces extra aggregate log-loss for any set of decoupled per-task predictors $\{q_t(\cdot \mid Z_s)\}$, unless the shared features Z_s makes tasks conditionally independent. This is why private features are useful as they can reduce Δ by explaining away residual coupling that a shared representation leaves behind.

For completeness, in appendix S1.1, we include a Fano-style per-task error lower bound as a consequence of the CR inequality.

When should tasks share? If tasks are redundant (large $TC(Y^{(1:T)})$ or large explained dependence $TC(Y^{(1:T)}) - TC(Y^{(1:T)} \mid Z_s)$), then sharing features is favorable. If tasks are heterogeneous (small redundancy) and T grows Theorem 4.2 implies that any globally shared bottleneck must either increase capacity roughly linearly in T or accept degraded per-task signal. Using this observation, we suggest clustered sharing and private capacity allocation as an approach to task heterogeneity.

5 SHARED–PRIVATE EXTENSION

In practice, multi-task models often augment shared features with task-private features that can absorb heterogeneity and mitigate interference. Here, we formalize and derive a corresponding shared–private CR bound. Let $Z_s = f_s(X)$ denote shared features and, for each task $t \in [T]$, let $Z_t = f_t(X)$ denote task-private features. The task-specific representation is $Z^{(t)} := (Z_s, Z_t)$. We assume the (test-time) Markov property $Y^{(1:T)} - X - (Z_s, Z_1, \ldots, Z_T)$, and impose separated information budgets

$$I(Z_s; X) \le C_s, \qquad I(Z_t; X \mid Z_s) \le C_t \quad \forall t \in [T]. \tag{8}$$

where $I(Z_t; X \mid Z_s)$ is a conditional budget that measures incremental private capacity beyond what the shared bottleneck already covers. This results in an additive upper bound on the total information extracted from X by the collection $(Z_s, Z_1, \ldots, Z_T)$, even when the private feature maps are statistically dependent through X.

Theorem 5.1 (Shared–private capacity–redundancy bound)**.** *Under Markov chain and Equation* (8)*,*

$$\sum_{t=1}^{T} I\Big(Z^{(t)}; Y^{(t)}\Big) \;\le\; C_s + \sum_{t=1}^{T} C_t \;+\; TC\Big(Y^{(1:T)}\Big). \tag{9}$$

Theorem 5.1 mirrors the shared-only CR inequality, with the shared capacity C_s replaced by a total budget $C_s + \sum_t C_t$. Thus, private features can reduce the residual task coupling left unexplained by a shared bottleneck. Shared-only slack Δ_s measures how task labels remain coupled after conditioning on the shared representation. When Δ_s is large, the shared representation has not factorized the joint task structure. In such a setting, global sharing is prone to interference unless C_s grows.

Next we use shared–private features to measure the post-augmentation residual coupling $\Delta_{sp} = TC(Y^{(1:T)} \mid Z_{\text{all}})$, where $Z_{\text{all}} = (Z_s, Z_1, \ldots, Z_T)$. Since conditioning cannot increase entropy, $TC(\cdot \mid \cdot)$ is monotone in the conditioning set, implying $\Delta_{sp} \le \Delta_s$. Thus private features can only decrease residual coupling, and doing so is what makes per-task heads easier to fit without forcing the shared bottleneck to encode heterogeneous, task-specific details.

The bound in Equation (9) becomes tight when the total representation saturates the information budget $I(Z_{\text{all}}; X) \approx C_s + \sum_t C_t$, and the residual dependence is largely removed, $\Delta_{sp} \approx 0$ (labels become approximately conditionally independent given Z_{all}). In contrast, if Δ_s remains large at fixed C_s, then increasing private budgets $\{C_t\}$ is the only way within this framework to reduce Δ_{sp} and avoid the shared-only obstruction identified in Section 4.

The shared–private inequality Equation (9) suggests to share capacity within groups of redundant tasks, and use private capacity to address residual coupling. Later in experiments we validate this by estimating task similarity and comparing global sharing, clustered sharing, and private adapters under matched total rank budgets.

6 SUBSETS AND SIDE INFORMATION

CR inequality constrains the aggregate information across all tasks T using a single redundancy term $TC(Y^{(1:T)})$ and a shared capacity C_s. However, in many settings, tasks are not uniformly related instead they organize into natural clusters, for instance when tasks share a common data modality or exhibit a similar label structure. In such cases, aggregate CR bounds over all tasks is too coarse to capture finer-grained task relationships. To address this, we consider subset-wise version of the inequality to capture redundancy within groups of related tasks.

Subset bounds. Let $S \subseteq [T]$ be any subset of tasks and denote $Y^S = \{Y^{(t)} : t \in S\}$. Then the subset total correlation is defined as $TC(Y^S) = \sum_{t \in S} H(Y^{(t)}) - H(Y^S)$. Applying Theorem 3.1 and Theorem 3.2 to the subcollection results in an immediate tightening.

Corollary 6.1 (Subset CR inequality)**.** *Under the assumptions of Theorem 3.2, for any $S \subseteq [T]$,*

$$\sum_{t \in S} I(Z_s; Y^{(t)}) \;\le\; C_s + TC(Y^S). \tag{10}$$

Moreover, the canonical sandwich holds on S,

$$I(Z_s; Y^S) \;\le\; \sum_{t\in S} I(Z_s; Y^{(t)}) \;\le\; I(Z_s; Y^S) + TC(Y^S). \tag{11}$$

The subset bound is often tighter than Equation (2) because redundancy is not uniform, tasks often form clusters with high intra-cluster dependence but low inter-cluster dependence. This refinement suggests that shared capacity should be allocated relative to $TC(Y^S)$ within each cluster, with respect to the global quantity $TC(Y^{(1:T)})$.

Side information and conditional CR. Let W denote side information available when constructing or analyzing the representation. The conditional total correlation is defined as $TC(Y^{(1:T)} \mid W) = \sum_{t=1}^{T} H(Y^{(t)} \mid W) - H(Y^{(1:T)} \mid W) \ge 0$.

Theorem 6.2 (Conditional CR inequality)**.** *Under a conditional Markov property $Y^{(1:T)} - X - Z_s$ given W, and a shared capacity budget $I(Z_s; X \mid W) \le C_s(W)$,*

$$\sum_{t=1}^{T} I(Z_s; Y^{(t)} \mid W) \;\le\; C_s(W) + TC(Y^{(1:T)} \mid W). \tag{12}$$

More generally, for any subset $S \subseteq [T]$,

$$\sum_{t\in S} I(Z_s; Y^{(t)} \mid W) \;\le\; C_s(W) + TC(Y^S \mid W). \tag{13}$$

Residual coupling with side information. The exact decomposition also holds conditionally $\sum_{t=1}^{T} I(Z_s; Y^{(t)} \mid W) = I(Z_s; Y^{(1:T)} \mid W) + \Big(TC(Y^{(1:T)} \mid W) - TC(Y^{(1:T)} \mid Z_s, W)\Big)$ and the conditional slack $\Delta(W) = TC(Y^{(1:T)} \mid Z_s, W) \ge 0$ again measures residual task coupling after observing Z_s, now at a fixed value of side information W. During training, the representation parameters depend on labels, so the unconditional Markov chain $Y^{(1:T)} - X - Z_s$ need not hold. However, at evaluation time weights are fixed which makes it possible to use conditional Markov property, DPI and CR arguments at test time. This is crucial as it separates learning dynamics from the representational limit that are the main focus of this work.

If W includes task identity or context, then $TC(Y^{(1:T)} \mid W)$ can be much smaller than $TC(Y^{(1:T)})$, since conditioning can decouple labels across tasks. In such a setting, using conditional adapters or MoE routing increase effective performance by reducing conditional redundancy terms or allowing $C_s(W)$ to vary with W. The conditional CR bound Equation (12) serves as a unified lens for these designs.

Implications for clustered sharing. Subset and conditional refinements together suggest to first estimate task similarity (as a proxy for dependence), cluster tasks into subsets $\{S_k\}$, and allocate shared capacity per cluster. Formally, applying Equation (10) within each cluster gives $\sum_{t\in S_k} I(Z_s^{(k)}; Y^{(t)}) \le C_s^{(k)} + TC(Y^{S_k})$ so at fixed total budget $\sum_k C_s^{(k)}$, clustered sharing targets the local redundancy structure rather than paying for global heterogeneity. We refer readers to Section S2 for a Gaussian model where both shared capacity and redundancy can be computed in closed form.

7 Application to LoRA Fine-tuning

Next, we introduce a capacity proxy to link adapter rank and scale to an effective shared budget C_s, and results in a testable predictions about when globally shared low-rank updates saturate and when clustered/private updates are necessary.

LoRA as a low-rank information channel. Consider a frozen backbone with a hidden representation $\phi(X) \in \mathbb{R}^d$. A LoRA update replaces a linear map $W \in \mathbb{R}^{o\times d}$ by

$$W \;\mapsto\; W + \Delta W, \quad \Delta W = BA, \quad A \in \mathbb{R}^{r\times d},\; B \in \mathbb{R}^{o\times r}, \tag{14}$$

where $r \ll \min\{o, d\}$ is the LoRA rank. We view LoRA branch as a feature channel that compresses the input into a low-dimensional signal $Z_s = A\,\phi(X) \in \mathbb{R}^r$, which is then mixed into the output

via B.[1] In a multi task setting, this implies a structural constraint that tasks sharing the same LoRA factors share the same bottleneck Z_s. Thus, the shared representation capacity is controlled by the rank r and the scale of A relative to the feature distribution $\phi(X)$. This is consistent with the intrinsic-dimensionality perspective (Aghajanyan et al., 2021).

CR predictions for LoRA design. Let us consider T tasks fine-tuned by sharing the same LoRA factors (A, B) (and thus the same bottleneck Z_s). Applying Theorem 3.2 with shared budget $C_s \approx I(Z_s; \phi(X))$ the capacity can be upper bound via Equation (S3.32). If tasks are heterogeneous so that $TC(Y^{(1:T)})$ is small, then CR prediction implies that the sum of task-relevant information is controlled primarily by $I(Z_s; \phi(X))$, which grows slowly with r. Consequently, performance under a single globally shared LoRA should saturate quickly as rank increases, and adding tasks without increasing rank induces interference (Section Section 4).

To obtain a per-task information level $I(Z_s; Y^{(t)}) \gtrsim b$ across many weakly dependent tasks, Corollary 4.2 tells us that the total shared information $I(Z_s; \phi(X))$ must scale approximately like Tb (modulo redundancy). Under our rank-controlled bound, LoRA rank (or effective rank) must grow roughly linearly in the number of distinct task directions or clusters. If tasks form clusters $\{S_k\}$ with high within-cluster dependence but low across-cluster dependence, then we can use the subset CR bound, sharing LoRA within each cluster and allocating rank per cluster,

	r=4	r=8	r=16
Shared LoRA	0.763	0.773	0.755
Private LoRA	0.796	0.795	0.797
Clustered LoRA	0.812	0.817	0.806

Table 1: **Mean validation accuracy** on SST2/MRPC/RTE for budget-valid total ranks. Clustered sharing consistently outperforms global sharing under matched rank budgets.

$$\sum_{t \in S_k} I(Z_s^{(k)}; Y^{(t)}) \;\le\; C_s^{(k)} + TC(Y^{S_k}), \quad \sum_k C_s^{(k)} \le C_{\text{tot}}.$$

Estimating residual coupling Δ from validation residuals. We use validation data $\{(x_i, y_i^{(1:T)})\}_{i=1}^n$ to compute per-task residuals $e_t^{(i)} = y_t^{(i)} - \hat{y}_t^{(i)}$. Let $\widehat{R}_e$ be the sample correlation matrix of $(e_1, \ldots, e_T)$. We approximate $\widehat{\Delta} = -\frac{1}{2} \log\det(\widehat{R}_e + \epsilon I) \approx \mathrm{TC}\big(Y^{(1:T)} \mid Z_s\big)$, and report it across different settings.

8 Empirical Validation

8.1 Validation on synthetic data.

We first verify the CR inequality in a controlled linear–Gaussian setting where all terms are known in closed form. Let $X \sim \mathcal{N}(0, I_d)$ with $d = 20$ and task labels be $Y^{(t)} = a_t^\top X + \varepsilon_t$, with $\varepsilon_t \sim \mathcal{N}(0, 0.1^2)$. By varying the alignment of $\{a_t\}$, we sweep label redundancy, parallel vectors results in large $TC(Y^{(1:T)})$, while near-orthogonal vectors result in $TC(Y^{(1:T)}) \approx 0$. We use a rank-r linear encoder $Z_s = W^\top X + \eta$ with $\eta \sim \mathcal{N}(0, 0.5^2 I_r)$ to get the representations. If $\mathrm{row}(A)$ contains $\mathrm{span}\{a_t\}_{t=1}^T$, then $\mathrm{TC}(Y^{(1:T)} \mid Z_s) = 0$ and the CR identity achieves equality. When $r < \mathrm{rank}(\mathrm{span}\{a_t\})$, the CR slack equals the residual coupling $\big(C_s + \mathrm{TC}(Y^{(1:T)})\big) - \sum_{t=1}^T I(Z_s; Y^{(t)}) = \mathrm{TC}(Y^{(1:T)} \mid Z_s)$ which decreases monotonically as r increases). We compute $I(Z_s; Y^{(t)})$ via Gaussian conditional-variance formulas and estimate $TC(Y^{(1:T)})$ from the covariance of Y.

The CR bound holds across all encoders and becomes tight when encoder row-space contains the task subspace (Figure 2). Once the encoder spans the task subspace, residual coupling vanishes and additional rank results in diminishing returns.

[1] In transformers LoRA is typically applied to W_q, W_v (and sometimes W_o, W_k). Our analysis applies to each adapted linear map and one can interpret $\phi(X)$ as a relevant input to that map at any given layer.

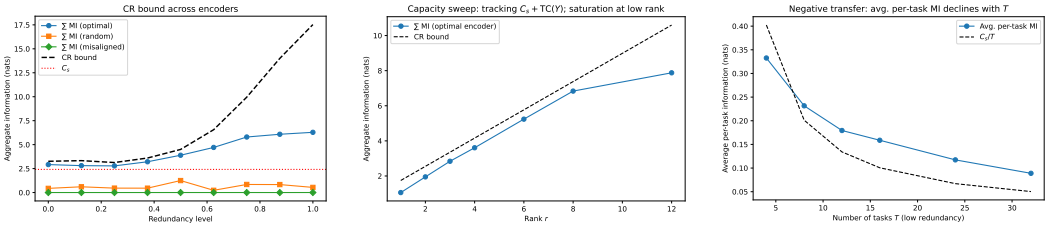


Figure 2: Left: CR holds across encoder types, with tightness for the optimal encoder and slack for random or misaligned ones. Middle: Capacity sweep shows aggregate information tracks $C_s + \mathrm{TC}(Y)$, saturating at low rank. Right: Negative transfer demonstrated, as average per-task information declines when the number of tasks increases under fixed capacity and low redundancy.

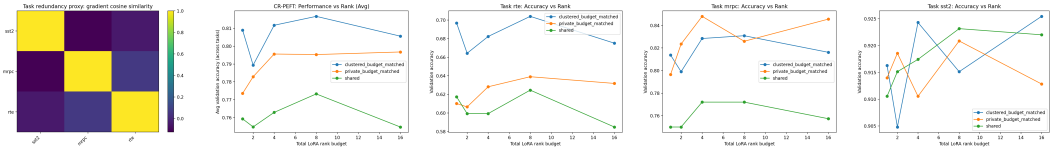
	SST-2			**MRPC**			**RTE**		
	r=4	r=8	r=16	r=4	r=8	r=16	r=4	r=8	r=16
Shared	.917	.923	.922	.772	.772	.757	.599	.625	.585
Private	.911	.921	.913	**.848**	.826	**.846**	.628	.639	.632
Clustered	.924	.915	**.926**	.828	**.831**	.816	**.682**	**.704**	**.675**

Table 2: Per-task validation accuracy on SST-2 / MRPC / RTE. Clustering shows improvement on the harder, low-resource tasks (RTE: +8 MRPC: +6 over shared), consistent with the CR prediction that low-redundancy tasks benefit most from dedicated capacity.

8.2 CR-ALIGNED PEFT ON GLUE

Next we evaluate whether CR is applicable in a practical setting by multi-task PEFT on a GLUE subset using a frozen `bert-base-uncased` encoder and LoRA adapters on query and value projections in attention layers. We train jointly on five tasks (SST-2, MRPC, RTE, QNLI, QQP) with separate heads and a shared or structured adapter budget, full details are in Appendix S4. We first compare three adapter-sharing variants under matched rank budgets (Table 1). Figure 3 shows clustered sharing dominates at every budget with the largest margin on RTE.

A single shared LoRA adapter of total rank $r_{\text{tot}} = 8$ results in a mean validation accuracy of 0.811 across the five tasks (Table S4.3) with $\mathrm{F1} = 0.849$) for MRPC. SST-2 is largely invariant across settings (≈0.92), confirming it is capacity-saturated at low rank. The improvement from clustering is significant on RTE and MRPC, both of which have low label redundancy with SST-2 (negative gradient cosine similarity Table S4.4) and therefore suffer most from forced parameter sharing consistent with the CR capacity constraint prediction. Additional plots and tables are provided in Appendix S4.2.



(a) Task Similarity (b) Mean GLUE (c) RTE Accuracy (d) MRPC Accuracy (e) SST-2 Accuracy

Figure 3: **GLUE Performance and Task Redundancy.** (a) Gradient similarity shows MRPC–RTE alignment, while SST2 is weakly antagonistic. (b–e) Accuracy vs. rank budget across tasks: Clustered sharing outperforms global sharing when tasks are heterogeneous, consistent with CR limits and task redundancy patterns.

9 CONCLUSION

We introduced the CR inequality, a distribution-level converse that upper-bounds the total predictive information extractable from a shared latent by its information capacity plus the redundancy among tasks. This result explains negative transfer as an unavoidable consequence of low redundancy and finite capacity, and provides a principled approach for the design of shared–private architectures, including LoRA-based fine-tuning schemes. Because TC measures redundancy, not synergy, CR does

not capture complementary information across tasks. Future extensions can use partial information decomposition or multivariate interaction measures (e.g., O-information (Rosas et al., 2019)) to go beyond redundancy. Moreover, extending the analysis to dynamic training settings may explain how optimization implicitly allocates capacity across tasks.

REFERENCES

Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision*, pages 94–108. Springer, 2014.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3994–4003, 2016.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167, 2008.

Li Deng, Geoffrey Hinton, and Brian Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. *ICASSP*, pages 8599–8603, 2013.

Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11293–11302, 2019.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bryan Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799, 2019.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353/.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature machine intelligence*, 5(3):220–235, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.

Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in neural information processing systems*, 34:1022–1035, 2021.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. *arXiv preprint arXiv:2106.04489*, 2021.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018.

Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A Smith, and Luke Zettlemoyer. Demix layers: Disentangling domains for modular language modeling. *arXiv preprint arXiv:2108.05036*, 2021.

Pramod Kaushik Mudrakarta, Mark Sandler, Andrey Zhmoginov, and Andrew Howard. K for the price of 1: Parameter-efficient multi-task and transfer learning. *arXiv preprint arXiv:1810.10703*, 2018.

Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117, 2004.

Rie Kubota Ando, Tong Zhang, and Peter Bartlett. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of machine learning research*, 6(11), 2005.

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine learning*, 73(3):243–272, 2008.

Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *International conference on machine learning*, pages 9120–9132. PMLR, 2020.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

Heyan Chai, Jinhao Cui, Ye Wang, Min Zhang, Binxing Fang, and Qing Liao. Improving gradient trade-offs between tasks in multi-task text classification. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2565–2579, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.144. URL https://aclanthology.org/2023.acl-long.144/.

Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12: 149–198, 2000.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

Alejandro Newell, Lu Jiang, Chong Wang, Li-Jia Li, and Jia Deng. Feature partitioning for efficient multi-task architectures, 2020. URL https://openreview.net/forum?id=B1eoyAVFwH.

Yuyan Wang, Zhe Zhao, Bo Dai, Christopher Fifty, Dong Lin, Lichan Hong, and Ed H Chi. Small towers make big differences. *arXiv preprint arXiv:2008.05808*, 2020.

Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhu, and James Kwok. Controllable pareto multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 20337–20348, 2020. URL https://proceedings.neurips.cc.

Michinari Momma, Chaosheng Dong, and Jia Liu. A multi-objective / multi-task learning framework induced by pareto stationarity. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15895–15907. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/momma22a.html.

Weizhu Qian, Bowei Chen, Yichao Zhang, Guanghui Wen, and Franck Gechter. Multi-task variational information bottleneck. *arXiv preprint arXiv:2007.00339*, 2020.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.

TH Sun. Linear dependence structure of the entropy space. *Inf. Control*, 29(4):337–368, 1975.

Mokshay Madiman and Prasad Tetali. Information inequalities for joint distributions, with interpretations and applications. *IEEE Transactions on Information Theory*, 56(6):2699–2713, 2010.

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7319–7328, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.568. URL https://aclanthology.org/2021.acl-long.568/.

Fernando E Rosas, Pedro AM Mediano, Michael Gastpar, and Henrik J Jensen. Quantifying high-order interdependencies via multivariate extensions of the mutual information. *Physical Review E*, 100(3):032305, 2019.

SUPPLEMENTARY MATERIAL

S1 NOTATION AND PRELIMINARIES

Symbol	Meaning
X	input variable
$Y_{1:T} = (Y_1, \dots, Y_T)$	task labels
Z	learned representation (shared or composite)
Z_s	shared representation
Z_t	private representation for task t
W	parameterized weight matrix
$H(\cdot)$	Shannon entropy
$I(A; B \mid C)$	conditional mutual information
$\mathrm{TC}(Y_{1:T} \mid U)$	conditional total correlation
C_s, C_t	capacity budgets (information or proxy budget)

Total correlation. For label sequence $Y_{1:T}$, the TC and conditional TC terms are defined as,

$$\mathrm{TC}(Y_{1:T}) = \sum_{t=1}^{T} H(Y_t) - H(Y_{1:T}), \qquad \mathrm{TC}(Y_{1:T} \mid U) = \sum_{t=1}^{T} H(Y_t \mid U) - H(Y_{1:T} \mid U). \tag{S1.15}$$

where $\mathrm{TC}(Y_{1:T} \mid U) \geq 0$ with equality iff $Y_1, \dots, Y_T$ are conditionally independent given U.

Identity underlying CR. For any representation Z and side information W,

$$\sum_{t=1}^{T} I(Z; Y_t \mid W) = I(Z; Y_{1:T} \mid W) + \mathrm{TC}(Y_{1:T} \mid W) - \mathrm{TC}(Y_{1:T} \mid Z, W). \tag{S1.16}$$

Corollary S1.1 (Per-task error lower bound via Fano)**.** *Assume each $Y^{(t)}$ takes values in a finite alphabet of size $|\mathcal{Y}_t| \geq 2$ and define the Bayes error given Z_s, $P^{\star}_{e,t} = \inf_{\hat{y}_t(Z_s)} \mathbb{P}[\hat{y}_t(Z_s) \neq Y^{(t)}]$. Then*

$$P^{\star}_{e,t} \;\geq\; \frac{H(Y^{(t)} \mid Z_s) - 1}{\log |\mathcal{Y}_t|} \;=\; \frac{H(Y^{(t)}) - I(Z_s; Y^{(t)}) - 1}{\log |\mathcal{Y}_t|}. \tag{S1.17}$$

There exists a task $t^{\star}$ with

$$P^{\star}_{e,t^{\star}} \;\gtrsim\; \frac{H(Y^{(t^{\star})}) - \frac{C_s + TC(Y^{(1:T)})}{T}}{\log |\mathcal{Y}_{t^{\star}}|}. \tag{S1.18}$$

S2 GAUSSIAN SPECIALIZATION AND TIGHTNESS

A linear–Gaussian model offers an interpretable setting where we can evaluate the CR bounds exactly. Additionally, it shows exact conditions such as span alignment, orthogonality, and residual coupling under which these bounds become tight.

Linear–Gaussian setup Let $X \sim \mathcal{N}(0, \Sigma_X) \in \mathbb{R}^d$, we assume tasks are generated as noisy linear measurements,

$$Y^{(t)} \;=\; u_t^{\top} X + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma_t^2), \tag{S2.19}$$

where $\varepsilon_{1:T}$ independent of X, we write $Y^{(1:T)} = U^{\top} X + \varepsilon$ where $U = [u_1, \dots, u_T] \in \mathbb{R}^{d \times T}$ and $\varepsilon \sim \mathcal{N}(0, \Sigma_{\varepsilon})$ with $\Sigma_{\varepsilon} = \mathrm{diag}(\sigma_1^2, \dots, \sigma_T^2)$. A shared representation is produced by a linear map with an additive Gaussian noise:

$$Z_s \;=\; AX + \eta, \qquad \eta \sim \mathcal{N}(0, \sigma_z^2 I_m), \qquad \eta \perp (X, \varepsilon), \tag{S2.20}$$

where $A \in \mathbb{R}^{m \times d}$ is a design matrix, and σ_z^2 controls the bottleneck noise.

S2.1 CLOSED-FORM CAPACITY AND REDUNDANCY

For the Gaussian channel equation S2.20,

$$I(Z_s; X) = \frac{1}{2} \log \det \left(I_m + \frac{1}{\sigma_z^2} A \Sigma_X A^\top \right). \tag{S2.21}$$

capacity here is dependent on the spectrum of $A\Sigma_X^{1/2}$ explicit and at fixed output dimension m, it is governed by the nonzero singular values and saturates when $A\Sigma_X A^\top$ becomes low-rank or small-norm.

Since $Y^{(1:T)}$ is jointly Gaussian with covariance $\Sigma_Y = U^\top \Sigma_X U + \Sigma_\varepsilon$, the total correlation reduces to a log-determinant ratio,

$$TC(Y^{(1:T)}) = \frac{1}{2} \log \frac{\prod_{t=1}^T \Sigma_{Y,tt}}{\det(\Sigma_Y)}. \tag{S2.22}$$

$TC(Y^{(1:T)})$ is large when task outputs are strongly correlated (e.g., when $\{u_t\}$ have large overlap in the Σ_X-geometry), and vanishes when Σ_Y is diagonal (independent tasks).

CR bound is tight in a Gaussian specialization under capacity saturation $I(Z_s; Y^{(1:T)}) \approx I(Z_s; X)$, and explained dependence $\Delta = TC(Y^{(1:T)} \mid Z_s) \approx 0$ (labels become nearly conditionally independent given Z_s).

When sharing is optimal. In the Gaussian setting $I(Z_s; Y^{(1:T)})$ depends on how well the row space of A captures the subspace spanned by $\{u_t\}$. Let the task subspace be $\mathcal{U} := \mathrm{span}\{u_1, \ldots, u_T\} \subseteq \mathbb{R}^d$ in the Σ_X-geometry. If the row space of A contains $\mathcal{U}$ (equivalently, AX preserves all task-relevant directions), then Z_s is a sufficient statistic for $U^\top X$ up to noise, and the joint information $I(Z_s; Y^{(1:T)})$ approaches $I(Z_s; X)$ as $\sigma_z^2 \to 0$ under a fixed power constraint on A. Thus, conditioning on Z_s removes most of the cross-task dependence induced by the shared latent X, so $\Delta \approx 0$ when the representation is informative enough about $U^\top X$. This results in a near-tightness of CR,

$$\sum_{t=1}^T I(Z_s; Y^{(t)}) \approx I(Z_s; X) + TC(Y^{(1:T)}). \tag{S2.23}$$

When sharing must fail at fixed rank. Suppose task directions are nearly orthogonal in the Σ_X inner product, so that $U^\top \Sigma_X U$ is close to diagonal. Then $TC(Y^{(1:T)})$ is small and there is little redundancy to reuse across tasks. Additionally, the representation rank is limited then Z_s cannot preserve all task directions simultaneously. In this scenario, $I(Z_s; Y^{(t)})$ must be small for many tasks and the sum information is bounded by capacity alone,

$$\sum_{t=1}^T I(Z_s; Y^{(t)}) \lesssim I(Z_s; X) \leq C_s, \tag{S2.24}$$

when $TC(Y^{(1:T)}) \approx 0$. This shows negative transfer under global sharing adding heterogeneous tasks forces interference unless the shared capacity scales with the number of distinct task directions.

Proposition S2.1 (Near-tightness under span capture)**.** *Assume $\Sigma_X = I_d$ and $\sigma_t^2 = \sigma^2$ for all t. Let $\mathcal{U} = \mathrm{span}\{u_1, \ldots, u_T\}$ have dimension k and suppose A has rank at least k and row space containing $\mathcal{U}$. Under a fixed power constraint $\mathrm{tr}(AA^\top) \leq P$, there exists a choice of A and σ_z^2 such that*

$$\sum_{t=1}^T I(Z_s; Y^{(t)}) \geq I(Z_s; X) + TC(Y^{(1:T)}) - \epsilon. \tag{S2.25}$$

S3 FULL PROOFS

S3.1 CR INEQUALITY AND SUBSET REGION

Theorem S3.1 (CR bound)**.** *Assume the Markov chain $Y_{1:T} - X - Z_s$ holds given W and the fixed capacity constraint $I(X; Z_s \mid W) \leq C_s$. Then*

$$\sum_{t=1}^T I(Z_s; Y_t \mid W) \leq C_s + \mathrm{TC}(Y_{1:T} \mid W). \tag{S3.26}$$

For any subset $S \subseteq [T]$ we can write,

$$\sum_{t \in S} I(Z_s; Y_t \mid W) \le C_s + \mathrm{TC}(Y_S \mid W). \tag{S3.27}$$

Proof. Using CR identity with $Z = Z_s$ and non-negativity of TC,

$$\sum_{t=1}^{T} I(Z_s; Y_t \mid W) \le I(Z_s; Y_{1:T} \mid W) + \mathrm{TC}(Y_{1:T} \mid W).$$

By data-processing under $Y_{1:T} - X - Z_s$ given W, $I(Z_s; Y_{1:T} \mid W) \le I(Z_s; X \mid W) \le C_s$. The subset statement repeats the same argument for Y_S. □

S3.2 Shared–private extension

Theorem S3.2 (Shared–private CR bound)**.** *Assume $Y_{1:T} - X - (Z_s, Z_{1:T})$ is a Markov chain given W and $I(X; Z_s \mid W) \le C_s$, $I(X; Z_t \mid W) \le C_t$ for each t. Then,*

$$\sum_{t=1}^{T} I((Z_s, Z_t); Y_t \mid W) \le C_s + \sum_{t=1}^{T} C_t + \mathrm{TC}(Y_{1:T} \mid W). \tag{S3.28}$$

Proof. For each task t, $I((Z_s, Z_t); Y_t \mid W) \le I((Z_s, Z_{1:T}); Y_t \mid W)$. Summing over t and applying CR identity with $Z = (Z_s, Z_{1:T})$ gives,

$$\sum_{t=1}^{T} I((Z_s, Z_{1:T}); Y_t \mid W) \le I((Z_s, Z_{1:T}); Y_{1:T} \mid W) + \mathrm{TC}(Y_{1:T} \mid W).$$

Using data-processing, $I((Z_s, Z_{1:T}); Y_{1:T} \mid W) \le I((Z_s, Z_{1:T}); X \mid W)$ with chain rule and non-negativity,

$$I((Z_s, Z_{1:T}); X \mid W) \le I(Z_s; X \mid W) + \sum_{t=1}^{T} I(Z_t; X \mid W) \le C_s + \sum_{t=1}^{T} C_t.$$

□

S3.3 LoRA capacity bound (Gaussian-noise channel)

Theorem S3.3 (Linear-Gaussian channel upper bound)**.** *Let U be any random vector with finite covariance $\Sigma_U = \mathrm{Cov}(U \mid W)$. Let $Z = AU + \xi$ with $\xi \sim \mathcal{N}(0, \sigma^2 I)$ independent of U given W. Then*

$$I(U; Z \mid W) \le \frac{1}{2} \log \det\big(I + \sigma^{-2} A \Sigma_U A^\top\big). \tag{S3.29}$$

If additionally $\mathrm{rank}(A) \le r$ and $\mathrm{tr}(A \Sigma_U A^\top) \le \kappa$, then

$$I(U; Z \mid W) \le \frac{r}{2} \log\Big(1 + \frac{\kappa}{r\sigma^2}\Big). \tag{S3.30}$$

Proof. Conditioning on W, we can write $\Sigma_Z = A\Sigma_U A^\top + \sigma^2 I$. Since $Z|U$ is a Gaussian noise, $h(Z \mid U) = h(\xi)$. By maximum-entropy $h(Z) \le \frac{1}{2} \log((2\pi e)^m \det(\Sigma_Z))$. Thus $I(U; Z) = h(Z) - h(Z \mid U) \le \frac{1}{2} \log \frac{\det(\Sigma_Z)}{\det(\sigma^2 I)}$. For the rank/trace bound, apply AM–GM on nonzero eigenvalues of $M = \sigma^{-2} A \Sigma_U A^\top \succeq 0$: $\log\det(I + M) \le r \log(1 + \mathrm{tr}(M)/r)$. □

A Gaussian surrogate for LoRA capacity. Consider a Gaussian/noisy-channel surrogate at inference time,

$$Z_s \;=\; A\,\phi(X) + \eta, \qquad \eta \sim \mathcal{N}(0, \sigma^2 I_r), \tag{S3.31}$$

where η is a noise used to make mutual information finite for deterministic encoders. Let $\Sigma_\phi := \mathrm{Cov}(\phi(X))$. Then, by the Gaussian channel formula,

$$I(Z_s;\phi(X)) = \frac{1}{2}\log\det\left(I_r + \frac{1}{\sigma^2} A\,\Sigma_\phi\,A^\top\right). \tag{S3.32}$$

where $\phi(X)$ is a deterministic function of X given frozen weights. Now using DPI $I(Z_s;X) \geq I(Z_s;\phi(X))$ and $I(Z_s;Y^{(1:T)}) \leq I(Z_s;X)$,, which implies equation S3.32 is a conservative, architecture-linked proxy for the shared budget.

Rank- and power-controlled upper bound. Assume a power constraint $\mathrm{tr}(AA^\top) \leq \kappa$ and denote by $\lambda_1(\Sigma_\phi) \geq \cdots \geq \lambda_d(\Sigma_\phi)$ the eigenvalues of Σ_ϕ. By eigenvalue majorization and log-det bounds, $I(Z_s;\phi(X)) \leq \frac{r}{2}\log\left(1 + \frac{\kappa}{r\,\sigma^2}\,\lambda_1(\Sigma_\phi)\right) \leq \frac{1}{2}\sum_{i=1}^{r}\log\left(1+\frac{\kappa}{\sigma^2}\,\lambda_i(\Sigma_\phi)\right)$.

This shows that effective shared capacity grows at most linearly with rank r, and it saturates if Σ_ϕ is low-rank or if κ is small.

S4 Experimental details

Here we provide full details for the PEFT experiments.

Tasks and data. We use a subset of GLUE tasks {SST-2, MRPC, RTE, QNLI, QQP} . Each task $t \in \mathcal{T}$ is a supervised problem with input x and label y_t. The evaluation for classification tasks use accuracy and for STS-B (regression) Pearson correlation.

Encoder. We use `bert-base-uncased` as frozen backbone except for LoRA parameters. It has $L = 12$ transformer layers, hidden size $d = 768$, and 12 attention heads.

Representation. We use the pooled output using the [CLS] token embedding as the feature vector $h(x) \in \mathbb{R}^{768}$.

S4.1 LoRA parameterization and budget

We use LoRA for the attention projection modules (query and value) in each layer. For a weight matrix $W \in \mathbb{R}^{d\times d}$, LoRA uses $\Delta W = (\alpha/r)\,BA$ where $A \in \mathbb{R}^{r\times d}$ and $B \in \mathbb{R}^{d\times r}$. Each LoRA module contributes $r(d+d) = 2rd$ parameters for $W \in \mathbb{R}^{d\times d}$. With two targets (query/value) per layer across L layers, a single adapter of rank r adds $2\times L\times 2\times(2rd) = 8Lrd$ trainable parameters up to a constant. Thus, fixing $\sum_i r_i$ across adapters approximately matches trainable parameter budgets.

When using K adapters (private $K = T$, clustered $K = \#$clusters), strict budget-matching requires $r_i \geq 1$ and $\sum_{i=1}^{K} r_i = R$. If $R < K$, the allocation is infeasible and the point is filtered. Each task has a separate linear head, we minimize mean task loss under multi-task sampling using cross-entropy for classification and MSE for STS-B.

We train LoRA parameters and task heads using AdamW with $\eta = 3\times 10^{-4}$, $\beta = (0.9, 0.999)$, weight decay $= 0.01$. We use a linear learning-rate schedule with warmup ratio 0.06. We use batch size 16, max sequence length is 128, and training is done for 3 epochs using mixed precision (fp16).

Let total rank budget be R, consider one adapter (rank R) shared by all tasks and one adapter per task (T adapters), then rank is allocated $\{r_t\}$ such that $\sum_t r_t = R$ and $r_t \geq 1$.

Clustered (similarity-aware, budget-matched). We first partition tasks into K clusters and use one adapter per cluster with ranks $\{r_k\}$ such that $\sum_k r_k = R$ and $r_k \geq 1$.

Task similarity estimation We estimate task similarity without training full adapters by attaching a probe LoRA adapter of rank 1, denoted a_{probe}, and computing per-task gradient vectors,

$$g_t := \frac{1}{B}\sum_{b=1}^{B}\nabla_{\theta_{\mathrm{probe}}}\ell_t(\theta_{\mathrm{probe}};\text{batch } b), \qquad \tilde{g}_t := \frac{g_t}{\|g_t\|_2}.$$

where for similarity we use cosine $s_{ij} = \langle \tilde{g}_i, \tilde{g}_j \rangle$. We use agglomerative clustering on the distance matrix $D_{ij} = 1 - s_{ij}$ with average linkage and fixed K.

To test whether improvements come from using similarity rather than using clusters, we generate M random partitions that preserve the similarity-based cluster sizes and train the clustered regime. We report mean and standard deviation over random partitions and compute a per-rank z-score,

$$z(R) = \frac{\mathrm{Score}_{\mathrm{clustered}}(R) - \mathbb{E}[\mathrm{Score}_{\mathrm{random}}(R)]}{\mathrm{Std}[\mathrm{Score}_{\mathrm{random}}(R)]}.$$

For each task pair (i, j) we estimate the benefit of sharing by comparing random partitions in which i, j co-cluster versus those in which they are separated,

$$\Delta_{ij}(R) := \mathbb{E}[\mathrm{Score}(R) \mid i, j \text{ co-cluster}] - \mathbb{E}[\mathrm{Score}(R) \mid i, j \text{ separated}].$$

We then correlate $\Delta_{ij}(R)$ with s_{ij} to link redundancy (similarity) and the value of sharing.

S4.2 Additional results

Table S4.3 reports the validation scores for the shared-adapter run at $r_{\mathrm{tot}} = 8$. Table S4.4 reports the estimated gradient-similarity matrix, and Table S4.5 gives the induced clustering.

Table S4.3: Shared LoRA (rank r=8) validation performance on GLUE-5.

Task	**Accuracy**	**F1**	**# Val**
SST-2	0.908	—	872
MRPC	0.777	0.849	408
RTE	0.599	—	277
QNLI	0.893	—	5,463
QQP	0.880	—	40,430
Mean	0.811		

S4.3 Multi-seed evaluation and random clustering baseline

We run S seeds per configuration. We report mean±std and compute paired differences where applicable. For each total rank budget, we sample $M = 50$ random partitions preserving the cluster sizes of the CR-guided partition. We report the random baseline mean±std and compute z-scores $z = \frac{\mathrm{acc}_{\mathrm{CR}} - \mu_{\mathrm{rand}}}{\sigma_{\mathrm{rand}}}$.

Table S4.4: Task–task gradient cosine similarity $s(t, t')$ under a rank-1 probe adapter (rounded to 3 decimals).

	SST-2	MRPC	RTE	QNLI	QQP
SST-2	1.000	-0.082	-0.147	-0.020	-0.168
MRPC	-0.082	1.000	0.021	-0.013	0.179
RTE	-0.147	0.021	1.000	0.016	-0.016
QNLI	-0.020	-0.013	0.016	1.000	0.175
QQP	-0.168	0.179	-0.016	0.175	1.000

Table S4.5: Clusters obtained by agglomerative clustering on distance $1 - s(t, t')$.

Cluster ID	Tasks
0	{MRPC, QQP}
1	{RTE}
2	{SST-2}
3	{QNLI}