BELIEFMERGE: BAYESIAN GAME THEORETIC MODEL MERGING WITH JOINT BELIEFS

Anonymous authors
Paper under double-blind review

ABSTRACT

Model merging combines multiple task-specific models fine-tuned from a shared base model into a single model with consolidated capabilities. Yet, model merging is inherently uncertain because each task-specific model cannot directly observe the private characteristics of others, making contribution coordination difficult. Existing methods typically rely on public model information or locally available task information without explicitly accounting for such uncertainty. To address this issue, we introduce `BeliefMerge`, a Bayesian game-theoretic framework for uncertainty-aware model merging. `BeliefMerge` constructs correlated joint beliefs over unobserved model characteristics and coordinates interdependent contributions via a Bayesian Nash Equilibrium. Theoretically, sufficient conditions are established for equilibrium uniqueness and convergence of exact best-response updates. Experiments across CLIP and GLUE show that `BeliefMerge` achieves the best performance on all nine CLIP configurations and five of eight GLUE tasks, demonstrating its empirical superiority.

1 INTRODUCTION

Fine-tuning the same base model on different tasks produces task-specific models with distinct capabilities. Model merging combines these models into a single model, enabling capability reuse across tasks (Wortsman et al., 2022; Ilharco et al., 2023). Existing merging methods can be broadly categorized according to the information accessible during merging: methods using only **public information**, including model parameters and parameter-derived structures, such as singular components and subspaces (Ilharco et al., 2023; Gargiulo et al., 2025); and methods additionally using **private information**, including task-local auxiliary data and data-dependent statistics that are associated with individual task-specific models and are not observable to other models (Matena & Raffel, 2022; Jin et al., 2023; Yang et al., 2024b).

Among *public-information* approaches, early methods operate directly on task-specific parameter changes, while more recent methods exploit structured representations derived from model parameters. Task Arithmetic, an early and widely adopted approach, defines a task vector as the parameter difference between a task-specific model and the base model, and linearly combines these vectors (Ilharco et al., 2023). Later, to mitigate interference across tasks, TIES-Merging removes low-magnitude parameter updates and resolves sign conflicts before aggregation (Yadav et al., 2023). TSV-M further applies singular value decomposition (SVD) to layer-wise task matrices and decorrelates task singular vectors across tasks to reduce interference (Gargiulo et al., 2025). Building on this decomposition-based view, Iso-CTS further identifies common and task-specific subspaces from the singular components of task matrices and applies isotropic scaling to preserve task-specific directions that may otherwise be underrepresented during merging (Marczak et al., 2025).

Among *private-information* approaches, merging additionally relies on task-specific information that remains local to the corresponding model or task and cannot be recovered from public model parameters alone. Such information is mainly exploited from three perspectives: **task sensitivity**, **representation relevance**, and **performance benefit**. Task sensitivity characterizes how parameter changes affect model performance on local task data. Fisher-weighted merging uses Fisher information to quantify parameter importance for each task (Matena & Raffel, 2022), while BE-CAME uses Fisher-based posterior precision for adaptive merging (Li et al., 2025). Representation relevance characterizes how task inputs induce changes in hidden representations or layer outputs.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

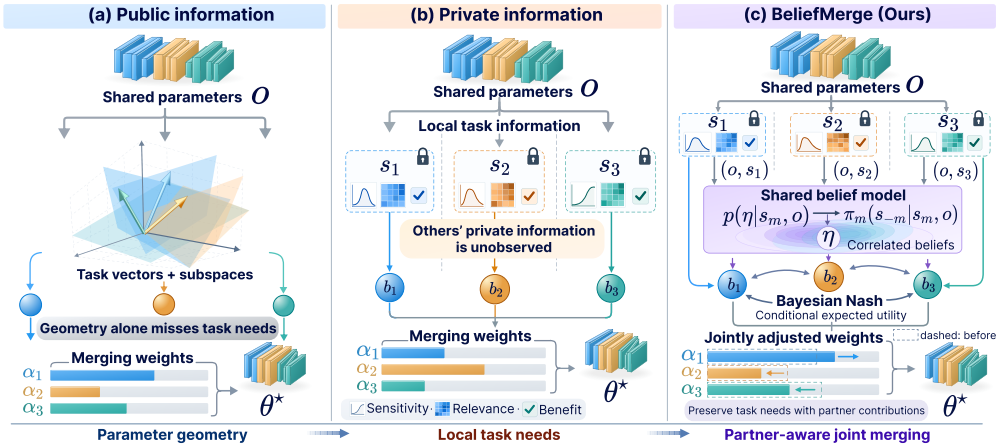


Figure 1: **Landscape of model merging under different sources.** Existing approaches rely on either (a) public model parameters only or (b) both public parameters and private task-local data. `BeliefMerge` (c) instead forms correlated joint beliefs and balances model contributions via a Bayesian Nash Equilibrium.

RegMean uses representation statistics to minimize discrepancies between merged and task-specific layer outputs (Jin et al., 2023), while BMM incorporates such statistics with an anchor prior into Bayesian regression to estimate merged parameters (Li et al., 2026a). MASS and Twin-Merging further use input representations to dynamically decide task-specific parameter components during inference (Crisostomi et al., 2026; Lu et al., 2024). Finally, performance benefit directly evaluates, or approximates, how a candidate merge performs on local task data. Typical examples, including AdaMerging (Yang et al., 2024b), aTLAS (Zhang et al., 2024), and MOBO-Merge (Agarwal et al., 2026), use prediction entropy, task-specific losses to optimize merging coefficients, respectively. Fig. 1(a)–(b) illustrate the landscape of existing model merging approaches under different sources of merging evidence.

Despite promising results, the balance among models remains underexplored especially with uncertainty of how well each individual model performs on their own task. We translate this into the following game scenario: each task-specific model, as a player in this game, has limited knowledge of the private characteristics of other models. That is, task-specific properties, such as parameter sensitivity or task performance on local data, are not directly observable across models. Arguably, this uncertainty complicates contribution coordination because all task models jointly determine the merged parameters, and adjusting one model's contribution can affect performance on other tasks. Each model can therefore benefit from estimating such unobserved private characteristics of others and decide its own model contribution. This naturally leads to the following question:

> ***How to dynamically balance the contributions of task-specific models while incorporating the uncertainty of performance of all models so that the merged model reaches optimum for most of the tasks without a full blown fine tuning?***

Coming from this angle, this paper introduces `BeliefMerge`, a Bayesian game-theoretic framework for uncertainty-aware model merging. Specifically, a shared base model is introduced to define a common prior linking private characteristics across models. Each model conditions this prior on public information and its own private information to form correlated joint beliefs over the unobserved private characteristics of other models, thereby capturing uncertainty in their contributions during merging. This uncertainty-aware merging is resolved through a Bayesian Nash Equilibrium, where interdependent contribution strategies are optimized via conditional expected utilities under these beliefs and materialized as merging weights. Our contributions are summarized as follows[1]:

- We are the *first*, to the best of our knowledge, to frame model merging as a *Bayesian game*, where a Bayesian Nash Equilibrium jointly addresses merging uncertainty and interdependent contributions.

[1]Source codes are available on https://anonymous.4open.science/r/Beliefmerge.

- We propose `BeliefMerge`, a unified framework spanning public/private information, prior and posterior belief inference, contribution strategies, and expected utility, providing *a general blueprint* for uncertainty-aware model merging.
- We provide *cyclic optimization scheme* for contribution strategies under the inferred joint beliefs and establish *sufficient conditions* for a unique pure-strategy Bayesian Nash Equilibrium and convergence of the optimization.
- We evaluate `BeliefMerge` on a variety of CLIP and GLUE settings, achieving the *best performance* on all 9 CLIP configurations and 5 out of 8 GLUE tasks, empirically confirming its superiority.

2 PRELIMINARIES

Problem formulation. Consider a base model parameterized by $\boldsymbol{\theta}_0$, from which M task-specific models are independently fine-tuned. Model m ($\in [1, M]$) is parameterized by $\boldsymbol{\theta}_m$ and retains a local dataset $\mathcal{D}_m = \{(\boldsymbol{x}_{m,n}, y_{m,n})\}_{n=1}^{N_m}$. The corresponding *task vector* is defined as $\Delta\boldsymbol{\theta}_m = \boldsymbol{\theta}_m - \boldsymbol{\theta}_0$. Furthermore, let $\boldsymbol{o}$ denote the *public information* observable to all models, such as the base parameters $\boldsymbol{\theta}_0$, task-specific parameters $\boldsymbol{\theta}_m$, and parameter-derived structures such as singular components of task vectors. In contrast, each model m possesses *private information* ($\boldsymbol{s}_m$), derived from its local dataset $\mathcal{D}_m$ and unobservable to other models. Such private information mainly characterizes *task sensitivity*, *representation relevance*, and *task-local performance*[2]. Let $\boldsymbol{s} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_M)$ collect the private information of all models, and let $\boldsymbol{s}_{-m}$ denote the private information of all models except m. A broad class of contribution-based merging methods (Matena & Raffel, 2022; Yang et al., 2024b; Crisostomi et al., 2026; Nguyen et al., 2026b) can be formulated as

$$\boldsymbol{\theta}^\star = \boldsymbol{\theta}_0 + \sum_{m=1}^{M} \alpha_m(\mathcal{I}_m)\Delta\boldsymbol{\theta}_m, \tag{1}$$

where $\alpha_m(\mathcal{I}_m)$ denotes the contribution weight from model m, determined by its available information $\mathcal{I}_m$. Public-information methods therefore rely solely on $\boldsymbol{o}$, yielding $\alpha_m(\mathcal{I}_m) = \alpha_m(\boldsymbol{o})$, whereas private-information methods additionally exploit $\boldsymbol{s}_m$, yielding $\alpha_m(\mathcal{I}_m) = \alpha_m(\boldsymbol{o}, \boldsymbol{s}_m)$.

Contribution assessment. Considering model merging with an 8-task CLIP setting using a ViT-B-32 encoder (Radford et al., 2021). Four task-specific models (*i.e.*, Cars, DTD, EuroSAT, and MNIST) are fixed after 64 fine-tuning steps, where they achieve approximately 80% average accuracy (Horoi et al., 2026), while the remaining four models continue training. Fig. 2 shows that the resulting merged performance varies substantially as the evolving models change, despite the four fixed models remaining unchanged. Notably, at step 2048, the merged accuracy falls below that at step 64, even though the corresponding task-specific models achieve higher standalone performance. This observation shows that *stronger individual models do not necessarily yield better merging outcomes*. Instead, model contributions are inherently interdependent and governed by a dynamic process. Furthermore, since task-specific characteristics have inborn randomness from stochastic optimization (as shades in Fig. 2 along the curves), this brings inevitable uncertainty. We formulate this as incomplete information scenario in a Bayesian Nash Game, *i.e.* a model (player) does not observe private information from others. Each player has to form its beliefs to some extent over unobserved characteristics and selects its action while accounting for others. Equilibrium of the dynamics is reached when no player benefits from unilateral deviation. More detailed background is provided in Appendix B.

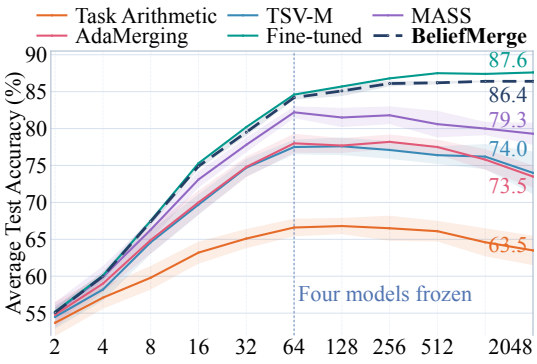


Figure 2: Merging performance with evolving task-specific models.

[2]Table 4 in Appendix C summarizes private-information categories and representative methods.

Yet, directly applying Bayesian Nash game theory to model merging is non-trivial. It requires defining meaningful private characteristics from task-specific models, constructing joint beliefs that preserve cross-model dependencies, and designing utilities that connect strategic decisions to high-dimensional parameter merging. Accordingly, `BeliefMerge` takes a first step toward Bayesian game-theoretic framework specifically tailored to model merging under incomplete information, while its empirical advantage over existing methods can be observed in Fig. 2.

3 METHOD

This section introduces `BeliefMerge`, a model merging method formulated from a Bayesian game-theoretic perspective, as illustrated in Fig. 3. At its core, `BeliefMerge` handles uncertainty over unobserved task-specific characteristics with correlated joint beliefs and coordinates model contributions through a Bayesian Nash Equilibrium.

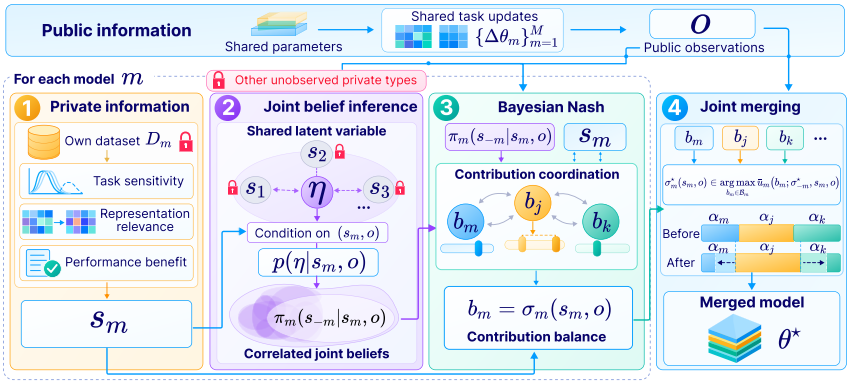


Figure 3: **Overview of `BeliefMerge`.** Each model combines public information with its own private information to infer correlated joint beliefs over unobserved characteristics of other models, and then coordinates interdependent contributions through Bayesian Nash Equilibrium.

3.1 PUBLIC AND PRIVATE INFORMATION

We first specify the public and private information available during model merging as follows.

Public information characterizes the task-vector geometry shared across models. Specifically, for each model m, the magnitude and normalized direction of its task vector $\Delta\boldsymbol{\theta}_m \in \mathbb{R}^d$ are defined as $g_m = \frac{|\Delta\boldsymbol{\theta}_m|_2}{|\boldsymbol{\theta}_0|_2+\epsilon}$ and $\widehat{\Delta\boldsymbol{\theta}}_m = \frac{\Delta\boldsymbol{\theta}_m}{|\Delta\boldsymbol{\theta}_m|_2+\epsilon}$, respectively, where $\epsilon > 0$ stabilizes normalization. The normalized task vectors are stacked as $\boldsymbol{G} = [\widehat{\Delta\boldsymbol{\theta}}_1, \ldots, \widehat{\Delta\boldsymbol{\theta}}_M] \in \mathbb{R}^{d\times M}$. Accordingly, let the shared singular basis $\boldsymbol{U}$ be defined as

$$\boldsymbol{G} = \widetilde{\boldsymbol{U}}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \qquad \boldsymbol{U} = \widetilde{\boldsymbol{U}}_{[:,1:r]} \in \mathbb{R}^{d\times r}, \qquad \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_r, \tag{2}$$

where $r \le \mathrm{rank}(\boldsymbol{G})$. This basis $\boldsymbol{U}$ provides a common coordinate system for comparing task-vector directions (Gargiulo et al., 2025; Crisostomi et al., 2026). The public observation of model m is then defined as

$$\boldsymbol{o}_m = [g_m; \boldsymbol{z}_m] \in \mathbb{R}^{r+1}, \tag{3}$$

where $\boldsymbol{z}_m = \boldsymbol{U}^\top\widehat{\Delta\boldsymbol{\theta}}_m$ represents the task-vector direction in the shared coordinate system, while g_m preserves its relative magnitude. Finally, $\boldsymbol{o} = (\boldsymbol{o}_1, \ldots, \boldsymbol{o}_M)$ denotes the public observations across all models.

Private information captures task-dependent characteristics derived from each model (with $\boldsymbol{\theta}_m$) and its local data ($\mathcal{D}_m$), and remains unavailable to other models. Specifically, these characteristics are measured layer-wise, as different layers can exhibit substantially different task dependencies that would be obscured by model-level aggregation. Three complementary aspects are considered: *task **sens**itivity*, ***repr**esentation relevance*, and ***perf**ormance benefit*. Each produces an L-dimensional score vector, where L is the total number of layers, forming the private information of model m as

$$\boldsymbol{s}_m = [\boldsymbol{s}_{m,\mathrm{sens}}^{(1,\cdots,\ell,\cdots,L)}; \boldsymbol{s}_{m,\mathrm{repr}}^{(1,\cdots,\ell,\cdots,L)}; \boldsymbol{s}_{m,\mathrm{perf}}^{(1,\cdots,\ell,\cdots,L)}]. \tag{4}$$

❶ **Task sensitivity.** To quantify how strongly the task relies on each layer-wise update, the sensitivity score is defined, with a small stabilizing constant $\epsilon > 0$, as

$$s^{(\ell)}_{m,\text{sens}} = \frac{\sum_{j=1}^{d_\ell} F^{(\ell)}_{m,j}([\Delta\boldsymbol{\theta}_m]_{\ell,j})^2}{\sum_{j=1}^{d_\ell} F^{(\ell)}_{m,j}([\Delta\boldsymbol{\theta}_m]_{\ell,j}) + \epsilon}, \tag{5}$$

where $[\Delta\boldsymbol{\theta}_m]_{\ell,j}$ is the update of its j-th parameter at layer ℓ with total d_ℓ parameters and $F^{(\ell)}_{m,j}$ is the corresponding empirical Fisher diagonal value computed on $\mathcal{D}_m$ (Matena & Raffel, 2022).

❷ **Representation relevance.** To assess how strongly a layer-wise task update affects the hidden representations $\boldsymbol{H}^{(\ell)}_m$, which is the layer-$\ell$ input representation matrix computed by θ_m on $\mathcal{D}_m$, following (Jin et al., 2023; Yao et al., 2025), the relevance score is defined as

$$s^{(\ell)}_{m,\text{repr}} = \frac{\|\boldsymbol{H}^{(\ell)}_m[\Delta\boldsymbol{\theta}_m]_\ell\|^2_F}{\|\boldsymbol{H}^{(\ell)}_m\|^2_F\|[\Delta\boldsymbol{\theta}_m]_\ell\|^2_F + \epsilon}, \tag{6}$$

where $[\Delta\boldsymbol{\theta}_m]_\ell$ denotes the parameter update of layer ℓ.

❸ **Performance benefit.** Let $\mathcal{E}_m(\cdot)$ denote the empirical mean loss on $\mathcal{D}_m$ (Chen et al., 2025), and $\boldsymbol{\theta}_{m,-\ell}$ be the parameters of model m with layer ℓ replaced by the same layer from $\boldsymbol{\theta}_0$. To quantify the performance benefit of the corresponding layer update, the score is defined as

$$s^{(\ell)}_{m,\text{perf}} = \text{sigmoid}\left(\frac{\mathcal{E}_m(\boldsymbol{\theta}_{m,-\ell}) - \mathcal{E}_m(\boldsymbol{\theta}_m)}{|\mathcal{E}_m(\boldsymbol{\theta}_m)| + \epsilon}\right). \tag{7}$$

Notably, the above construction is modular rather than restrictive: additional task-dependent statistics can be incorporated as private information, while other globally accessible quantities can be treated as public information. An ablation study is provided to examine how alternative information compositions affect merging performance.

3.2 Joint Belief Inference

Since task-specific models are fine-tuned using the same base model, their private information may exhibit statistical dependencies. To capture such dependencies, a shared latent variable $\boldsymbol{\eta}$ is introduced across all models, with prior $p(\boldsymbol{\eta}) = \mathcal{N}(\boldsymbol{\eta} \mid \boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta)$. Furthermore, let $f : \mathbb{R}^{r+1} \to \mathbb{R}^{3L}$ map $\boldsymbol{o}_m$ to $\boldsymbol{s}_m$. The transformed private information can be modeled as

$$\boldsymbol{s}_m = f(\boldsymbol{o}_m) + \boldsymbol{\eta} + \boldsymbol{\varepsilon}_m, \qquad \boldsymbol{\varepsilon}_m \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\varepsilon), \tag{8}$$

where $\boldsymbol{\varepsilon}_m$ captures model-specific uncertainty and is mutually independent across models and independent of $\boldsymbol{\eta}$. Accordingly, we have $p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m) = \mathcal{N}(\boldsymbol{s}_m \mid f(\boldsymbol{o}_m) + \boldsymbol{\eta}, \boldsymbol{\Sigma}_\varepsilon)$. Furthermore, conditioned on $\boldsymbol{\eta}$, the private information across models is independent to have $p(\boldsymbol{s} \mid \boldsymbol{\eta}, \boldsymbol{o}) = \prod_{m=1}^{M} p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m)$. Marginalizing $\boldsymbol{\eta}$ then induces the common prior over all private information:

$$\pi(\boldsymbol{s} \mid \boldsymbol{o}) = \int p(\boldsymbol{\eta}) \prod_{m=1}^{M} p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m) d\boldsymbol{\eta}. \tag{9}$$

Although the models are conditionally independent given $\boldsymbol{\eta}$, marginalizing $\boldsymbol{\eta}$ induces cross-model dependencies, allowing correlated private characteristics to be represented jointly.

Posterior belief over $\boldsymbol{\eta}$. Bayes' rule gives $\pi_m(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o}) \propto p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m)p(\boldsymbol{\eta})$. By Gaussian conjugacy, the posterior remains Gaussian $\pi_m(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o}) = \mathcal{N}\left(\boldsymbol{\eta} \mid \boldsymbol{\mu}_{\eta|m}, \boldsymbol{\Sigma}_{\eta|m}\right)$, where $\boldsymbol{\Sigma}_{\eta|m} = \left(\boldsymbol{\Sigma}_\eta^{-1} + \boldsymbol{\Sigma}_\varepsilon^{-1}\right)^{-1}$, and $\boldsymbol{\mu}_{\eta|m} = \boldsymbol{\Sigma}_{\eta|m}\left[\boldsymbol{\Sigma}_\eta^{-1}\boldsymbol{\mu}_\eta + \boldsymbol{\Sigma}_\varepsilon^{-1}(\boldsymbol{s}_m - f(\boldsymbol{o}_m))\right]$.

Belief over other models. For any $j \neq m$, model m cannot observe $\boldsymbol{s}_j$ but observes its public information $\boldsymbol{o}_j$. Its predictive belief over $\boldsymbol{s}_j$ is obtained by marginalizing the shared latent variable: $\pi_m(\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o}) = \int p(\boldsymbol{s}_j \mid \boldsymbol{\eta}, \boldsymbol{o}_j)\pi_m(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o})d\boldsymbol{\eta}$. Since both distributions are Gaussian, the integral admits the closed-form solution $\pi_m(\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o}) = \mathcal{N}\big(\boldsymbol{s}_j \mid f(\boldsymbol{o}_j) + \boldsymbol{\mu}_{\eta|m}, \boldsymbol{\Sigma}_\varepsilon + \boldsymbol{\Sigma}_{\eta|m}\big)$. Hence, $\mathbb{E}_{\pi_m}[\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o}] = f(\boldsymbol{o}_j) + \boldsymbol{\mu}_{\eta|m}$, where $f(\boldsymbol{o}_j)$ provides the public-information-based estimate and $\boldsymbol{\mu}_{\eta|m}$ incorporates the belief update induced by model m's private information.

Correlated joint belief. The individual predictive beliefs above do not capture the dependencies among other models induced by $\boldsymbol{\eta}$. Let $\boldsymbol{s}_{-m}$ denote the transformed private information of all models except m. Model m therefore forms the joint belief

$$\pi_m(\boldsymbol{s}_{-m} \mid \boldsymbol{s}_m, \boldsymbol{o}) = \int \left[\prod_{j \neq m} p(\boldsymbol{s}_j \mid \boldsymbol{\eta}, \boldsymbol{o}_j) \right] \pi_m(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o}) d\boldsymbol{\eta}, \tag{10}$$

which preserves the dependencies among unobserved private information through the shared latent variable $\boldsymbol{\eta}$. Detailed derivation is in Appendix C.2.

3.3 Bayesian Nash Equilibrium-based Merging

The contribution of each model is controlled by a learnable strategy conditioned on $\boldsymbol{o}$ and $\boldsymbol{s}_m$. Specifically, the strategy produces a model-specific action that determines its relative contribution to the merged model. Let σ_m, implemented as an MLP, denote the strategy function of model m, with its action defined as

$$b_m = \sigma_m(\boldsymbol{s}_m, \boldsymbol{o}), \tag{11}$$

where a larger b_m indicates a stronger preference for a higher merging contribution. Let $\boldsymbol{b} = (b_1, \ldots, b_M)$ denote the joint action profile for all models. The contribution weight of model m is obtained through

$$\alpha_m(\boldsymbol{b}) = \frac{\exp(b_m)}{\sum_{j=1}^{M} \exp(b_j)}, \tag{12}$$

such that $\alpha_m(\boldsymbol{b}) > 0$ and $\sum_{m=1}^{M} \alpha_m(\boldsymbol{b}) = 1$. Accordingly, the resulting merged task vector is

$$\Delta\boldsymbol{\theta}(\boldsymbol{b}) = \sum_{m=1}^{M} \alpha_m(\boldsymbol{b}) \Delta\boldsymbol{\theta}_m, \tag{13}$$

and the merged model is parameterized by $\boldsymbol{\theta}(\boldsymbol{b}) = \boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}(\boldsymbol{b})$.

The following defines the utility function and formulates model merging as a Bayesian game, in which the strategy functions and belief-model parameters are optimized toward a Bayesian Nash Equilibrium. Specifically, for model m, the utility is defined as

$$u_m(\boldsymbol{b}; \boldsymbol{s}_m, \boldsymbol{o}) = -\lambda_t \frac{\|\Delta\boldsymbol{\theta}(\boldsymbol{b}) - \Delta\boldsymbol{\theta}_m\|_2^2}{\|\Delta\boldsymbol{\theta}_m\|_2^2 + \epsilon} + \lambda_o \frac{\|\boldsymbol{U}^\top \Delta\boldsymbol{\theta}(\boldsymbol{b})\|_2^2}{\|\Delta\boldsymbol{\theta}(\boldsymbol{b})\|_2^2 + \epsilon} - \lambda_b b_m^2, \tag{14}$$

where $\epsilon > 0$ is a stabilizing constant, and λ_t, λ_o, and λ_b are the hyperparameters ($\lambda_t + \lambda_o + \lambda_b = 1$), respectively. The first term encourages the merged update to preserve the task vector of model m, while the second rewards alignment with the shared basis $\boldsymbol{U}$ in Eq. (2). The third term regularizes excessive contribution requests.

Let $\boldsymbol{\sigma}_{-m}(\boldsymbol{s}_{-m}, \boldsymbol{o})$ denote the joint actions of the other models under their respective strategies. Given a candidate action b_m and the strategies of the other models, model m evaluates its conditional expected utility under the correlated joint belief:

$$\bar{u}_m\left(b_m, \boldsymbol{\sigma}_{-m}; \boldsymbol{s}_m, \boldsymbol{o}\right) = \mathbb{E}_{\boldsymbol{s}_{-m} \sim \pi_m(\cdot \mid \boldsymbol{s}_m, \boldsymbol{o})} \left[u_m\left(\left(b_m, \boldsymbol{\sigma}_{-m}(\boldsymbol{s}_{-m}, \boldsymbol{o})\right); \boldsymbol{s}_m, \boldsymbol{o} \right) \right]. \tag{15}$$

Thus, uncertainty over $\boldsymbol{s}_{-m}$ propagates through the strategies of the other models to their actions and, consequently, to the merged model.

Let $\boldsymbol{\sigma}^\star = (\sigma_1^\star, \ldots, \sigma_M^\star)$ denote a strategy profile. It forms a Bayesian Nash Equilibrium if, for every model m, its strategy is a best response to the equilibrium strategies of the other models under its posterior belief:

$$\sigma_m^\star(\boldsymbol{s}_m, \boldsymbol{o}) \in \underset{\boldsymbol{b}_m \in \mathcal{B}_m}{\arg\max}\, \bar{u}_m(\boldsymbol{b}_m; \sigma_{-m}^\star, \boldsymbol{s}_m, \boldsymbol{o}), \tag{16}$$

At equilibrium, no model can improve its conditional expected utility by changing its contribution while strategies of other models remain fixed. Under the sufficient conditions as established in Appendix D[3], the resulting Bayesian game admits a unique pure-strategy Bayesian Nash Equilibrium, with exact best-response updates guaranteed to converge to this equilibrium.

[3] Appendix D states the sufficient conditions, proves equilibrium uniqueness and convergence of cyclic exact best-response updates, and describes the optimization scheme and model construction.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmarks and metrics. Following prior work (Crisostomi et al., 2026), the proposed `BeliefMerge` is evaluated on CLIP (Radford et al., 2021) with ViT-B-32, ViT-B-16, and ViT-L-14 encoders across nested suites of 8, 14, and 20 vision tasks (Ilharco et al., 2023; Wang et al., 2024; Dosovitskiy et al., 2021), as well as on merging 8 Flan-T5-Base models (Chung et al., 2024) fine-tuned on GLUE (Wang et al., 2019). Performance is evaluated using both average absolute and normalized accuracy[4].

Baselines. Public-information baselines include **Weight Averaging** (Wortsman et al., 2022), **Task Arithmetic** (Ilharco et al., 2023), **TIES-Merging** (Yadav et al., 2023), **Consensus TA** (Wang et al., 2024), **TSV-M** (Gargiulo et al., 2025), and **Iso-C/Iso-CTS** (Marczak et al., 2025). Private-information baselines include **RegMean++** (Nguyen et al., 2026b), **AdaMerging** (Yang et al., 2024b), **WeMoE** (Tang et al., 2024), **SMILE** (Tang et al., 2026), and **MASS** (Crisostomi et al., 2026). All baseline results are directly sourced from their original papers.

Implementation details. Experiments are conducted on NVIDIA A100 GPUs with 80 GB memory. The model is optimized using AdamW with a learning rate of 2×10^{-5}, a warmup ratio of 5%, gradient clipping at 1.0. We implement f in Eq. (8) as an MLP predicting layerwise private information s_m from public observations o_m. For `BeliefMerge`, the shared subspace rank is set to $r = 4$, while the hyperparameters λ_t, λ_o and λ_b are set to 0.5,0.3 and 0.2, respectively. All results from `BeliefMerge` are averaged over five independent runs.

4.2 MAIN RESULTS

Table 1 shows that `BeliefMerge` achieves the highest average accuracy across all nine CLIP configurations with ViT-B-32, ViT-B-16, and ViT-L-14, outperforming the strongest baseline (*i.e.*, MASS) by 2.6% points on average. Averaged across backbones, the gains increase from 1.4 to 2.3 and 4.0 points for 8, 14, and 20 tasks, respectively. Notably, the advantage becomes more evident with larger task collections (namely 20), suggesting that coordinating model contributions through joint beliefs becomes increasingly beneficial as cross-task interactions grow more complex.

Table 1: Performance comparison on **CLIP** in terms of average absolute accuracy and normalized accuracy (shown as subscripts). The **best** and second-best results are highlighted.

Method	ViT-B-32			ViT-B-16			ViT-L-14		
	8	14	20	8	14	20	8	14	20
Zero-shot	$48.1_{(54.8)}$	$56.9_{(64.5)}$	$57.5_{(65.2)}$	$55.3_{(60.6)}$	$61.9_{(67.9)}$	$62.5_{(68.3)}$	$64.9_{(69.2)}$	$69.1_{(73.8)}$	$68.2_{(72.7)}$
Fine-tuned	$90.3_{(100)}$	$89.0_{(100)}$	$89.5_{(100)}$	$92.4_{(100)}$	$91.3_{(100)}$	$91.9_{(100)}$	$94.2_{(100)}$	$93.4_{(100)}$	$94.0_{(100)}$
Weight Averaging	$67.1_{(74.6)}$	$65.5_{(73.8)}$	$64.4_{(72.6)}$	$73.0_{(78.8)}$	$70.8_{(77.3)}$	$69.2_{(75.3)}$	$80.5_{(85.2)}$	$78.5_{(83.8)}$	$76.1_{(80.9)}$
Task Arithmetic	$68.8_{(75.7)}$	$64.6_{(72.5)}$	$64.0_{(71.9)}$	$73.0_{(78.3)}$	$70.6_{(77.0)}$	$69.0_{(75.0)}$	$84.4_{(89.3)}$	$80.4_{(85.8)}$	$76.9_{(81.7)}$
Consensus TA	$72.6_{(80.1)}$	$70.3_{(78.9)}$	$68.5_{(76.8)}$	$75.9_{(81.7)}$	$74.9_{(81.7)}$	$72.2_{(78.4)}$	$85.5_{(90.5)}$	$82.0_{(87.6)}$	$78.9_{(83.8)}$
TSV-M	$83.2_{(91.8)}$	$78.6_{(88.0)}$	$75.6_{(84.3)}$	$85.5_{(92.2)}$	$81.4_{(88.8)}$	$78.8_{(85.5)}$	$91.2_{(96.7)}$	$88.8_{(94.9)}$	$87.5_{(93.0)}$
Iso-C	$82.8_{(91.7)}$	$78.4_{(88.0)}$	$73.2_{(81.9)}$	$87.5_{(94.4)}$	$79.8_{(87.0)}$	$75.3_{(81.6)}$	$92.6_{(98.2)}$	$89.6_{(95.8)}$	$86.8_{(92.3)}$
Iso-CTS	$82.0_{(90.9)}$	$80.6_{(90.4)}$	$77.0_{(86.2)}$	$88.7_{(95.9)}$	$84.1_{(91.8)}$	$80.7_{(87.7)}$	$92.8_{(98.5)}$	$\underline{91.1}_{(97.4)}$	$89.2_{(94.9)}$
AdaMerging	$80.1_{(88.7)}$	$-_{(-)}$	$69.3_{(77.4)}$	$84.9_{(91.9)}$	$-_{(-)}$	$72.8_{(79.2)}$	$90.8_{(96.4)}$	$-_{(-)}$	$79.8_{(84.9)}$
RegMean++	$84.4_{(93.5)}$	$-_{(-)}$	$74.2_{(82.9)}$	$87.2_{(94.4)}$	$-_{(-)}$	$76.9_{(83.7)}$	$91.0_{(96.6)}$	$-_{(-)}$	$82.7_{(88.0)}$
WeMoE	$\underline{88.8}_{(97.5)}$	$74.3_{(82.8)}$	$68.2_{(76.3)}$	$89.1_{(96.4)}$	$76.6_{(83.2)}$	$65.0_{(70.5)}$	$88.7_{(94.2)}$	$72.3_{(76.8)}$	$65.0_{(69.4)}$
SMILE-1	$83.2_{(92.1)}$	$75.4_{(84.5)}$	$72.8_{(82.3)}$	$87.8_{(94.9)}$	$81.7_{(89.5)}$	$79.5_{(86.7)}$	$91.2_{(96.7)}$	$86.6_{(92.7)}$	$84.9_{(90.5)}$
SMILE-2	$84.4_{(93.5)}$	$76.4_{(85.6)}$	$74.1_{(83.8)}$	$89.0_{(96.2)}$	$82.7_{(90.7)}$	$80.4_{(87.7)}$	$92.0_{(97.6)}$	$87.1_{(93.4)}$	$85.5_{(91.1)}$
MASS	$87.0_{(96.5)}$	$\underline{82.9}_{(93.2)}$	$\underline{81.1}_{(90.9)}$	$\underline{90.6}_{(98.0)}$	$\underline{87.8}_{(96.1)}$	$\underline{81.1}_{(88.7)}$	$\underline{92.9}_{(98.6)}$	$90.9_{(97.3)}$	$\underline{90.8}_{(96.6)}$
`BeliefMerge`	$\mathbf{89.2}_{(98.8)}$	$\mathbf{86.5}_{(97.2)}$	$\mathbf{86.4}_{(96.5)}$	$\mathbf{91.7}_{(99.2)}$	$\mathbf{89.6}_{(98.1)}$	$\mathbf{86.9}_{(94.6)}$	$\mathbf{93.8}_{(99.6)}$	$\mathbf{92.4}_{(98.9)}$	$\mathbf{91.7}_{(97.6)}$
Std.	$\pm$ 0.06	$\pm$ 0.22	$\pm$ 0.23	$\pm$ 0.37	$\pm$ 0.45	$\pm$ 0.40	$\pm$ 0.19	$\pm$ 0.38	$\pm$ 0.44

In addition, Table 2 reports the average merging performance of task-specific Flan-T5-Base models on GLUE. Similarly, `BeliefMerge` achieves competitive performance across all tasks, obtaining the best results on five of the eight tasks and the second-best results on the remaining three.

[4]Training, validation, and test splits, together with detailed evaluation metrics, are provided in Appendix E.

Table 2: Performance comparison on **GLUE** in terms of average absolute accuracy and normalized accuracy (shown as subscripts). The **best** and second-best results are highlighted.

Method	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST-2	STS-B	Avg.
Fine-tuned	$75.0_{(100)}$	$83.4_{(100)}$	$87.5_{(100)}$	$91.5_{(100)}$	$85.4_{(100)}$	$85.9_{(100)}$	$93.6_{(100)}$	$88.7_{(100)}$	$86.4_{(100)}$
Weight Averaging	$69.1_{(92.1)}$	$62.6_{(75.1)}$	$79.4_{(90.7)}$	$89.8_{(98.1)}$	$83.9_{(98.2)}$	$81.2_{(94.5)}$	$91.7_{(97.9)}$	$73.2_{(82.5)}$	$78.9_{(91.3)}$
Task Arithmetic	$70.5_{(94.0)}$	$57.8_{(69.3)}$	$78.4_{(89.6)}$	$90.2_{(98.6)}$	$83.6_{(97.9)}$	$80.5_{(93.7)}$	$92.3_{(98.6)}$	$77.8_{(87.7)}$	$78.9_{(91.3)}$
TIES-Merging	$70.3_{(93.7)}$	$65.0_{(77.9)}$	$78.9_{(90.2)}$	$90.2_{(98.6)}$	$83.5_{(97.8)}$	$81.6_{(95.0)}$	$91.7_{(97.9)}$	$78.3_{(88.3)}$	$79.9_{(92.5)}$
WeMoE	$72.5_{(96.6)}$	$79.0_{(94.7)}$	$51.9_{(59.3)}$	$89.3_{(97.5)}$	$69.6_{(81.4)}$	$81.5_{(94.8)}$	$88.6_{(94.6)}$	$82.1_{(92.5)}$	$76.8_{(88.9)}$
SMILE-1	$72.0_{(96.0)}$	$\mathbf{84.2}_{(101.0)}$	$84.3_{(96.3)}$	$\mathbf{91.3}_{(99.8)}$	$84.7_{(99.2)}$	$84.1_{(97.9)}$	$93.3_{(99.7)}$	$87.0_{(98.1)}$	$85.1_{(98.5)}$
SMILE-2	$73.2_{(97.6)}$	$\mathbf{84.2}_{(101.0)}$	$85.0_{(97.1)}$	$\mathbf{91.3}_{(99.8)}$	$84.9_{(99.4)}$	$84.8_{(98.7)}$	$\underline{93.5}_{(99.9)}$	$87.3_{(98.4)}$	$85.5_{(99.0)}$
MASS	$\underline{74.1}_{(98.8)}$	$83.2_{(99.8)}$	$\underline{85.8}_{(98.1)}$	$\underline{90.9}_{(99.3)}$	$\underline{85.1}_{(99.6)}$	$\underline{84.9}_{(98.8)}$	$\mathbf{94.2}_{(100.6)}$	$\mathbf{88.9}_{(100.2)}$	$\underline{85.9}_{(99.4)}$
BeliefMerge	$\mathbf{74.7}_{(99.6)}$	$\underline{83.3}_{(99.9)}$	$\mathbf{87.1}_{(99.5)}$	$\mathbf{91.3}_{(99.8)}$	$\mathbf{85.3}_{(99.9)}$	$\mathbf{85.5}_{(99.5)}$	$\underline{93.5}_{(99.9)}$	$\underline{88.6}_{(99.9)}$	$\mathbf{86.2}_{(99.8)}$
Std.	$\pm$ 0.11	$\pm$ 0.05	$\pm$ 0.1	$\pm$ 0.06	$\pm$ 0.02	$\pm$ 0.15	$\pm$ 0.09	$\pm$ 0.07	$\pm$ 0.08

Together with the CLIP results, these findings support the generality of BeliefMerge across different model architectures and task domains.

4.3 Ablation Study

To analyze the effectiveness of key components in BeliefMerge, a series of ablation studies is conducted on the 20-task CLIP setting using ViT-B-32 and ViT-B-16, covering two encoders under the largest task-merging scenario.

Effect of Public and Private Information. This ablation evaluates the contribution of each information component in BeliefMerge. Five *w/o* variants are constructed by individually removing one component from the public or private information defined in Eqs. (3) and (4): g_m (update magnitude), $\boldsymbol{z}_m$ (update direction), $\boldsymbol{s}_{m,\text{sens}}$ (task sensitivity), $\boldsymbol{s}_{m,\text{repr}}$ (representation relevance), and $\boldsymbol{s}_{m,\text{perf}}$ (performance benefit). In addition, two augmented variants are considered: $+c_m$, which adds similarity between task vectors to $\boldsymbol{o}_m$, and $+s_{m,\text{stab}}$, which adds gradient coherence to $\boldsymbol{s}_m$. Their definitions are provided in Appendix C.3. Fig. 4 shows that removing $\boldsymbol{z}_m$ causes the largest accuracy drops, by 4.1% and 3.7% on ViT-B-32 and ViT-B-16, respectively, suggesting that task-vector direction is critical for capturing cross-task compatibility and avoiding conflicting updates. Meanwhile, adding either public or private information further improves performance, indicating that the added factors provide complementary cues and demonstrating the extensibility of BeliefMerge to additional information sources.

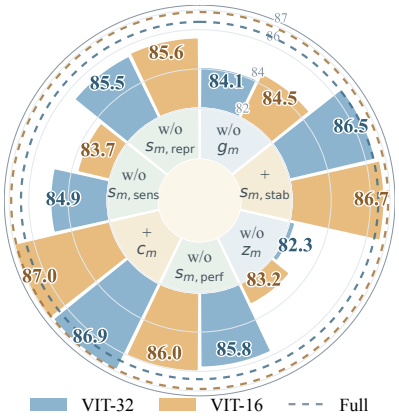


Figure 4: Public and Private Information ablation.

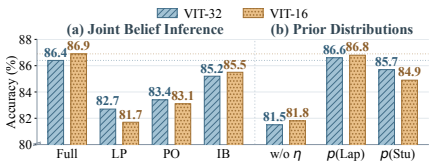


Figure 5: Effect of beliefs and priors.

Effect of Joint Belief Inference and Prior Distributions. This section evaluates the components of joint belief inference and whether strong merging performance can be retained with alternative priors. Given $(\boldsymbol{s}_m, \boldsymbol{o})$, **Full** uses an MLP predictor $f(\boldsymbol{o}_j)$ and a Gaussian prior $p(\boldsymbol{\eta})$ to form $\pi_m(\boldsymbol{s}_{-m} \mid \boldsymbol{s}_m, \boldsymbol{o})$ and fit contribution strategies through conditional expected utility. We consider (i) *Linear Predictor* (***LP***), replacing f with $\boldsymbol{W}\boldsymbol{o}_j + \boldsymbol{c}$ to test nonlinear prediction; (ii) *Prior Only* (***PO***), replacing $p(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o})$ with $p(\boldsymbol{\eta})$ to test posterior updating while retaining $\boldsymbol{s}_m$ in strategy inputs; and (iii) *Independent Beliefs* (***IB***), replacing the joint belief Eq. (10) with $\prod_{j \neq m} \pi_m(\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o})$ to test cross-model dependence while preserving conditional marginals. Additionally, $p(\text{Lap})$ and $p(\text{Stu})$ use multivariate Laplace and Student's t priors respectively, both matching the Gaussian mean μ_η and covariance Σ_η, with 3 degrees of freedom for the latter. Model-specific noise remains Gaussian; inference follows each prior, and strategies are refitted with other components and budgets

unchanged. Fig. 5 shows that PO, IB, and w/o η (removing the shared latent variable) reduce accuracy by up to 3.8, 1.4, and 5.1 percentage points, respectively, supporting posterior updating and correlated joint beliefs. Laplace remains close to Full, whereas Student-t performs worse, supporting Gaussian as an effective default while demonstrating flexibility across priors.

Effect of Bayesian Inference and Nash Coordination. This section examines Bayesian inference and Nash coordination through four variants: (i) *w/o* **BPI** removes posterior inference and applies classical Nash bargaining over utility gains relative to uniform averaging; (ii) *w/o* **NO** retains joint beliefs but replaces individual best responses with joint expected-utility maximization, $\max_\sigma \mathbb{E}_{\boldsymbol{s}\sim\pi(\cdot|\boldsymbol{o})}[\frac{1}{M}\sum_m u_m(\sigma(\boldsymbol{s},\boldsymbol{o});\boldsymbol{s}_m,\boldsymbol{o})]$; (iii) *Gaussian Posterior Merging* (***GPM***), inspired by BMM (Li et al., 2026a), fits task-specific residual outputs from local representations H_m using Gaussian likelihoods and a Gaussian prior, and uses the posterior mean of the merged update to obtain $\theta^\star = \theta_0 + \mathbb{E}[\Delta\theta \mid \{(H_m, \Delta\theta_m)\}_{m=1}^M]$; and (iv) *Gradient Nash Bargaining* (***GNB***) replaces belief inference and strategy fitting with Nash bargaining over first-order utility gains directly updating $\boldsymbol{b}$. The first two variants and GNB retain the original utilities and merging rule, whereas GPM estimates merged parameters. Together, these comparisons assess component contributions and alternative merging mechanisms. Fig. 6 shows that removing either component consistently degrades performance, with average absolute accuracy drops of 2.9-3.5%, 2.2-1.9%, 4.1-4.3% and 0.7-1.8% on ViT-B-32 and ViT-B-16, respectively. Bayesian inference helps account for the unobserved characteristics of other models, while Nash optimization coordinates their interdependent contributions. Their complementary effects support the joint use of posterior beliefs and equilibrium-based coordination in `BeliefMerge`.

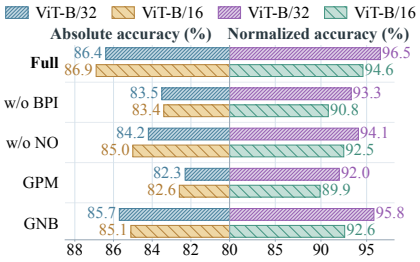


Figure 6: Effect of Bayesian inference and Nash coordination.

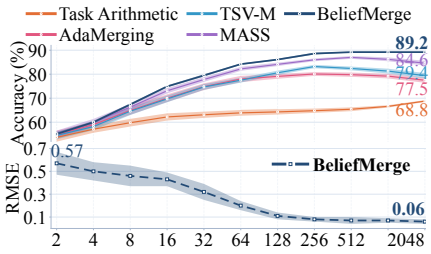


Figure 7: Performance comparison on evolving private tasks.

Evaluation under evolving tasks. The preliminary analysis in Sec. 2 shows that the value of a model contribution depends on the states of other models. To further evaluate `BeliefMerge` under dynamically evolving task states, all task-specific models are fine-tuned simultaneously for up to 2,048 steps, and merging is performed at successive checkpoints. This setting evaluates both whether `BeliefMerge` maintains effective merging as task states evolve and whether its Bayesian beliefs accurately recover the underlying private information, measured by $\mathrm{RMSE}(\boldsymbol{s}_j^\star, \widehat{\boldsymbol{s}}_{j|m})$, where $\boldsymbol{s}_j^\star$ and $\widehat{\boldsymbol{s}}_{j|m}$ denote the ground-truth and estimated private information of model j, respectively. As shown in Fig. 7, `BeliefMerge` reaches 89.2% accuracy at 512 steps and maintains it through 2,048 steps, outperforming the strongest baseline by 4.6% at the final steps. RMSE decreases from 0.57 to 0.06, with smaller reductions after 256 steps, supporting improved private-information estimation and sustained merging performance.

Other Analyses. Appendix D.3 details belief calibration and strategy optimization. Appendices F and G further analyze computational efficiency and hyperparameter sensitivity, respectively.

5 Conclusion

This paper studies model merging under incomplete information, where task-specific models share public information but cannot directly observe private task characteristics of others. To address this challenge, `BeliefMerge` is proposed to form correlated joint beliefs through a shared base model and formulates contribution selection as a Bayesian game. Theoretical analysis establishes sufficient conditions for equilibrium uniqueness and convergence of exact best-response updates. Extensive experiments demonstrate the superiority of `BeliefMerge` over strong baselines across CLIP and GLUE benchmarks. Future work will explore more expressive belief models and extend the framework to larger-scale and heterogeneous model merging settings.

AI USE STATEMENT

In this work, generative AI tools are not used for research ideation, method design, experimental design, or result analysis, and the remaining disclosure categories are not applicable to this work. We take responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work focuses on model merging methodology and does not involve human subjects, personal data, or sensitive user information. All experiments are conducted on publicly available benchmarks and pretrained models under their respective usage terms. The proposed framework is intended to improve coordination among task-specific models and does not introduce mechanisms for collecting or inferring personal information. Potential risks mainly concern downstream misuse of merged models, which should be assessed according to the deployment context. The authors take responsibility for ensuring that the reported methodology and experiments follow applicable research and data-use standards.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide detailed formulations of `BeliefMerge`, including public/private information construction, correlated joint-belief inference, Bayesian Nash optimization, and the associated utility design. Experiments are conducted on publicly available CLIP models with ViT-B-32, ViT-B-16, and ViT-L-14 encoders across 8-, 14-, and 20-task settings, as well as Flan-T5-Base models fine-tuned on GLUE. Training configurations, hyperparameters, evaluation metrics, and additional implementation details are reported in the main paper and Appendix E and G. All reported results are averaged over five independent runs, and the source code is released through the anonymous repository provided in https://anonymous.4open.science/r/Beliefmerge.

REFERENCES

Utkarsh Agarwal, Vamshi Bonagiri, Raul Astudillo, and Monojit Choudhury. Multi-objective bayesian optimization for model merging. *arXiv preprint arXiv:2608.14264*, 2026.

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023.

Long Minh Bui, Tuan Anh Le Van, Tung Phi Duc, Phi Le Nguyen, Jana Doppa, and Trong Nghia Hoang. Model merging as probabilistic inference in fine-tuning parameter space. In *Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence*, 2026.

Anda Cao, Zhuo Gou, Yi Wang, Kaixuan Chen, Yu Wang, Can Wang, Mingli Song, and Jie Song. Evolutionary negative module pruning for better LoRA merging. In *the Annual Meeting of the Association for Computational Linguistics*, 2026.

Levy Chaves, Chao Zhou, Rebekka Burkholz, Eduardo Valle, and Sandra Avila. Bridging domains through subspace-aware model merging. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Hao Mark Chen, Shell Xu Hu, Wayne Luk, Timothy Hospedales, and Hongxiang Fan. FW-merging: Scaling model merging with frank-wolfe optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Weiyu Chen and James Kwok. Pareto merging: Multi-objective optimization for preference-aware model merging. In *International Conference on Machine Learning*, 2025.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *International Conference on Machine Learning*, 2025.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, pp. 1–53, 2024.

Donato Crisostomi, Alessandro Zirilli, Antonio Andrea Gargiulo, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, Iacopo Masi, and Emanuele Rodolà. MASS: MoErging through adaptive subspace selection. In *The International Conference on Learning Representations*, 2026.

Akash Dhasade, Divyansh Jhunjhunwala, Milos Vujasinovic, Gauri Joshi, and Anne-Marie Kermarrec. Navigating the accuracy-size trade-off with flexible model merging. In *International Conference on Learning Representations*, 2026.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Guodong Du, Junlin Lee, Jing Li, Runhua Jiang, Yifei Guo, Shuyang Yu, Hanting Liu, Sim Kuan Goh, Ho-Kin Tang, Daojing He, and Min Zhang. Parameter competition balancing for model merging. In *Advances in Neural Information Processing Systems*, 2024.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. Task singular vectors: Reducing task interference in model merging. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

John C. Harsanyi. Games with incomplete information played by "bayesian" players, i-iii. part i. the basic model. *Management Science*, 14(3):159–182, 1967.

Amin Heyrani Nobari, Kaveh Alimohammadi, Ali ArjomandBigdeli, Akash Srivastava, Faez Ahmed, and Navid Azizan. Activation-informed merging of large language models. In *Advances in Neural Information Processing Systems*, 2025.

Stefan Horoi, Albert Manuel Orozco Camacho, Eugene Belilovsky, and Guy Wolf. Harmony in diversity: Merging neural networks with canonical correlation analysis. In *International Conference on Machine Learning*, 2024.

Stefan Horoi, Guy Wolf, Eugene Belilovsky, and Gintare Karolina Dziugaite. From memorization to parameter interference: How overtraining experts harms model merging. In *International Conference on Machine Learning*, 2026.

Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. EMR-merging: Tuning-free high-performance model merging. In *Advances in Neural Information Processing Systems*, 2024.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *The International Conference on Learning Representations*, 2023.

Youngeun Kim, Seunghwan Lee, Aecheon Jung, Bogon Ryu, and Sungeun Hong. Task vector quantization for memory-efficient model merging. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Chanhyuk Lee, Jiho Choi, Chanryeol Lee, Donggyun Kim, and Seunghoon Hong. Adarank: Adaptive rank pruning for enhanced model merging. In *International Conference on Learning Representations*, 2026.

Sanwoo Lee, Jiahao Liu, Qifan Wang, Jingang Wang, Xunliang Cai, and Yunfang Wu. Dynamic fisher-weighted model merging via Bayesian optimization. In *Proceedings of Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2025a.

Yu-Ang Lee, Ching-Yun Ko, Tejaswini Pedapati, I-Hsin Chung, Mi-Yen Yeh, and Pin-Yu Chen. STAR: Spectral truncation and rescale for model merging. In *Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics*, 2025b.

Kaiyang Li, Shaobo Han, Qing Su, and Shihao Ji. Bayesian Model Merging. *arXiv preprint arXiv:2605.12843*, 2026a.

Longhua Li, Lei Qi, Qi Tian, and Xin Geng. Model merging in the essential subspace. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026b.

Lujun Li, Qiyuan Zhu, Jiacheng Wang, Xiaoyu Qin, Wei Li, Hao Gu, Sirui Han, and Yike Guo. Sub-moe: Efficient mixture-of-expert llms compression via subspace expert merging. In *the AAAI Conference on Artificial Intelligence*, 2026c.

Mei Li, Yuxiang Lu, Qinyan Dai, Suizhi Huang, Yue Ding, and Hongtao Lu. BECAME: Bayesian continual learning with adaptive model merging. In *Forty-second International Conference on Machine Learning*, 2025.

Yayuan Li, Ze Peng, Jian Zhang, Jintao Guo, Yue Duan, and Yinghuan Shi. When shared knowledge hurts: Spectral over-accumulation in model merging. In *International Conference on Machine Learning*, 2026d.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, 2021.

Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. Twin-merging: Dynamic integration of modular expertise in model merging. In *Advances in Neural Information Processing Systems*, 2024.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost Van De Weijer. No task left behind: isotropic model merging with common and task-specific subspaces. In *International Conference on Machine Learning*, 2025.

Michael S Matena and Colin Raffel. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems*, 2022.

Tommaso Mencattini, Robert Adrian Minut, Donato Crisostomi, Andrea Santilli, and Emanuele Rodolà. MERGE^3: Efficient evolutionary merging on consumer-grade GPUs. In *Proceedings of the International Conference on Machine Learning*, 2025.

John F. Nash. The bargaining problem. *Econometrica*, pp. 155–162, 1950.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *International Conference on Machine Learning*, 2022.

Dung Viet Nguyen, Anh Thi, Minh Hoang Nguyen, Luc Nguyen, Shiqi Jiang, Ethan Fetaya, Duy Linh Tran, Gal Chechik, and Tan Nguyen. Expert merging in sparse mixture of experts with nash bargaining. In *International Conference on Learning Representations*, 2026a.

The-Hai Nguyen, Dang Huu-Tien, Takeshi Suzuki, and Le-Minh Nguyen. RegMean++: Enhancing effectiveness and generalization of regression mean for model merging. *Transactions on Machine Learning Research*, 2026b.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems*, 2023.

Aniello Panariello, Daniel Marczak, Simone Magistri, Angelo Porrello, Bartł omiej Twardowski, Andrew Bagdanov, SIMONE CALDERARA, and Joost van de Weijer. Accurate and efficient low-rank model merging in core space. In *Advances in Neural Information Processing Systems*, 2025.

Haiquan Qiu, You Wu, Dong Li, Jianmin Guo, and Quanming Yao. Superpose task-specific features for model merging. In *Conference on Empirical Methods in Natural Language Processing*, 2025.

Zihuan Qiu, Lei Wang, Yang Cao, Runtong Zhang, Bing Su, Yi Xu, Fanman Meng, Linfeng Xu, Qingbo Wu, and Hongliang Li. Null-space filtering for data-free continual model merging: Preserving stability, promoting plasticity. In *International Conference on Learning Representations*, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Adir Rahamim, Asaf Yehudai, Boaz Carmeli, Leshem Choshen, Yosi Mass, and Yonatan Belinkov. Will it merge? on the causes of model mergeability. In *Findings of the Association for Computational Linguistics*, 2026.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, 2018.

Bedionita Soro, Bruno Andreis, Hayeon Lee, Wonyong Jeong, Song Chong, Frank Hutter, and Sung Ju Hwang. Diffusion-based neural network weights generation. In *International Conference on Learning Representations*, 2025.

George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. ZipIt! merging models from different tasks without training. In *International Conference on Learning Representations*, 2024.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the knots. In *International Conference on Learning Representations*, 2025.

Wenju Sun, Qingyong Li, Yangliao Geng, and Boyang Li. CAT merging: A training-free approach for resolving conflicts in model merging. In *International Conference on Machine Learning*, 2025a.

Wenju Sun, Qingyong Li, Wen Wang, Yang Liu, Yangliao Geng, and Boyang Li. Towards minimizing feature drift in model merging: Layer-wise task vector fusion for adaptive knowledge integration. In *Advances in Neural Information Processing Systems*, 2025b.

Anke Tang, Li Shen, Yong Luo, Nan Yin, Lefei Zhang, and Dacheng Tao. Merging multi-task models via weight-ensembling mixture of experts. In *International Conference on Machine Learning*, 2024.

Anke Tang, Li Shen, Yong Luo, Enneng Yang, Han Hu, Lefei Zhang, Bo Du, and Dacheng Tao. FusionBench: A unified library and comprehensive benchmark for deep model fusion. *Journal of Machine Learning Research*, pp. 1–38, 2025a.

Anke Tang, Enneng Yang, Li Shen, Yong Luo, Han Hu, Lefei Zhang, Bo Du, and Dacheng Tao. Merging on the fly without retraining: A sequential approach to scalable continual model merging. In *Advances in Neural Information Processing Systems*, 2025b.

Anke Tang, Li Shen, Yong Luo, Shuai Xie, Han Hu, Lefei Zhang, Bo Du, and Dacheng Tao. Zero-shot sparse mixture of low-rank experts construction from pre-trained foundation models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1145–1157, 2026.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019.

Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *International Conference on Machine Learning*, 2024.

Tiantong Wang, Yiyang Duan, Haoyu Chen, Tiantong Wu, and Wei Yang Bryan Lim. M-Loss: Quantifying model merging compatibility with limited unlabeled data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Qi Wei, Shuo He, Enneng Yang, Tingcong Liu, Haobo Wang, Lei Feng, and Bo An. Representation surgery in model merging with probabilistic modeling. In *International Conference on Machine Learning*, 2025a.

Yongxian Wei, Anke Tang, Li Shen, Zixuan Hu, Chun Yuan, and Xiaochun Cao. Modeling multi-task model merging as adaptive projective gradient descent. In *International Conference on Machine Learning*, 2025b.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, 2022.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, 2023.

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. Representation surgery for multi-task model merging. In *International Conference on Machine Learning*, 2024a.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. In *International Conference on Learning Representations*, 2024b.

Zongzhen Yang, Binhang Qi, Hailong Sun, Wenrui Long, Ruobing Zhao, and Xiang Gao. CABS: Conflict-aware and balanced sparsification for enhancing model merging. In *International Conference on Machine Learning*, 2025.

Yuxuan Yao, Shuqi LIU, Zehua Liu, Qintong Li, Mingyang LIU, Xiongwei Han, Zhijiang Guo, Han Wu, and Linqi Song. Activation-guided consensus merging for large language models. In *Advances in Neural Information Processing Systems*, 2025.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning*, 2024.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, 2020.

Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Robustmerge: Parameter-efficient model merging for mllms with direction robustness. In *Advances in Neural Information Processing Systems*, 2025.

Frederic Z. Zhang, Paul Albert, Cristian Rodriguez-Opazo, Anton van den Hengel, and Ehsan Abbasnejad. Knowledge composition using task vectors with learned anisotropic scaling. In *Advances in Neural Information Processing Systems*, 2024.

Luca Zhou, Bo Zhao, Rose Yu, and Emanuele Rodola. Demystifying mergeability: Interpretable properties to predict model merging success. In *International Conference on Machine Learning*, 2026.

BeliefMerge: Bayesian Game Theoretic Model Merging with Joint Beliefs

(Appendix)

CONTENTS

Appendices **15**

A Notation and Terminology **16**

B Related Work **17**

B.1 Model Merging with Public Information 17

B.2 Model Merging with Private Information 17

B.3 Joint Beliefs for Model Merging 18

C Private Information and Joint Belief Inference **19**

C.1 Private-Information Diagnostics 19

C.2 Posterior Derivations and Joint Sampling 22

C.3 Additional Evidence for Information Ablation 23

D Merging Strategies and Bayesian Nash Equilibrium **24**

D.1 Merging Strategies 24

D.2 Best Response 24

D.3 Optimization 25

D.4 Nash Equilibrium 25

E Dataset Splits and Metrics **27**

E.1 CLIP 27

E.2 GLUE 27

F Time Complexity Analysis **29**

G Hyperparameter Analysis **31**

G.1 Balance Among the Utility Terms 31

G.2 Shared-Subspace Rank 32

A NOTATION AND TERMINOLOGY

Table 3 summarizes the main notation used in `BeliefMerge`. The symbols are organized according to the main components of `BeliefMerge`: task models, public and private information, joint belief inference, and contribution coordination under the Bayesian Nash equilibrium criterion.

Table 3: Main notation used in `BeliefMerge`.

Task models	
M, L, d	Number of task models (M), number of layers (L), and parameter dimension (d).
$\boldsymbol{\theta}_0, \boldsymbol{\theta}_m$	Parameters of the base model ($\boldsymbol{\theta}_0$) and task model m ($\boldsymbol{\theta}_m$).
$\Delta\boldsymbol{\theta}_m$	Task vector $\boldsymbol{\theta}_m - \boldsymbol{\theta}_0$, fixed throughout merging.
$\mathcal{D}_m$	Local task dataset available to model m.
Public information	
g_m	Task-vector magnitude relative to the base parameter norm.
$\boldsymbol{U}, r$	Shared singular basis ($\boldsymbol{U}$) and subspace dimension (r).
$\boldsymbol{z}_m$	Coordinates of the normalized task vector in the shared singular basis.
$\boldsymbol{o}_m, \boldsymbol{o}$	Public observation for model m ($\boldsymbol{o}_m = [g_m; \boldsymbol{z}_m]$) and all public observations ($\boldsymbol{o}$).
Private information	
$\boldsymbol{s}_m$	Private information comprising three layerwise score vectors.
$\boldsymbol{s}_{m,\text{sens}}$	Layerwise task sensitivity scores.
$\boldsymbol{s}_{m,\text{repr}}$	Layerwise representation relevance scores.
$\boldsymbol{s}_{m,\text{perf}}$	Layerwise performance benefit scores.
Joint belief inference	
$f(\boldsymbol{o}_m)$	Prediction of $\boldsymbol{s}_m$ from public observation $\boldsymbol{o}_m$.
$\boldsymbol{\eta}, \boldsymbol{\varepsilon}_m$	Shared latent variable ($\boldsymbol{\eta}$) and model-specific Gaussian noise ($\boldsymbol{\varepsilon}_m$).
$\boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta, \boldsymbol{\Sigma}_\varepsilon$	Prior mean ($\boldsymbol{\mu}_\eta$) and covariance ($\boldsymbol{\Sigma}_\eta$) of $\boldsymbol{\eta}$; noise covariance ($\boldsymbol{\Sigma}_\varepsilon$).
$\boldsymbol{\mu}_{\eta\|m}, \boldsymbol{\Sigma}_{\eta\|m}$	Posterior mean ($\boldsymbol{\mu}_{\eta\|m}$) and covariance ($\boldsymbol{\Sigma}_{\eta\|m}$) of $\boldsymbol{\eta}$ given $(\boldsymbol{s}_m, \boldsymbol{o})$.
$\pi_m(\boldsymbol{s}_{-m} \mid \boldsymbol{s}_m, \boldsymbol{o})$	Correlated joint belief of model m over the private information of the other models.
Merging strategies and Bayesian Nash equilibrium	
$\sigma_m(\boldsymbol{s}_m, \boldsymbol{o})$	Strategy mapping private information and public observations to a contribution action.
$b_m, \boldsymbol{b}$	Contribution action of model m (b_m) and joint action profile ($\boldsymbol{b}$).
$\mathcal{B}_m$	Feasible action space of model m.
$\alpha_m(\boldsymbol{b})$	Contribution weight for model m, obtained by softmax across models.
$\Delta\boldsymbol{\theta}(\boldsymbol{b})$	Merged task vector assembled from task vectors using model-specific contribution.
$\lambda_t, \lambda_o, \lambda_b$	Coefficients for task-vector preservation (λ_t), shared-subspace alignment (λ_o), and action regularization (λ_b).
$u_m(\boldsymbol{b}; \boldsymbol{s}_m, \boldsymbol{o})$	Utility combining task preservation, subspace alignment, and action regularization.
$\bar{u}_m$	Conditional expected utility under π_m and the strategies of the other models.
$\sigma_m^\star$	Strategy of model m in a Bayesian Nash equilibrium.

B Related Work

B.1 Model Merging with Public Information

Shared model parameters and quantities derived solely from them provide public information for model merging. We organize these approaches around two complementary perspectives: combining and refining task vectors, and exploiting singular subspaces to structure interactions among updates.

Task vectors. Uniform parameter averaging combines compatible models (Wortsman et al., 2022), while task arithmetic represents task-specific adaptations as parameter updates from the same pre-trained model (Ilharco et al., 2023). Tangent-space formulations further study task composition through linearized models (Ortiz-Jimenez et al., 2023). Directly combining updates can introduce cancellation and interference. To mitigate these effects, sign-aware selection, sparsification, and rescaling regulate individual updates (Yadav et al., 2023; Yu et al., 2024; Huang et al., 2024), while competition balancing and structured sparsification control overlapping contributions (Du et al., 2024; Yang et al., 2025). Complementary approaches address storage and parameter alignment: task-vector quantization reduces storage costs (Kim et al., 2025), and the weight-matching variant of Re-Basin aligns models through permutations using only parameters (Ainsworth et al., 2023).

Singular subspaces. Singular-subspace methods examine the structure within parameter updates to identify shared and task-specific directions. To reduce interference, KnOTS aligns updates through a joint singular decomposition (Stoica et al., 2025), while TSV-M decorrelates task singular vectors (Gargiulo et al., 2025). Common and task-specific subspaces distinguish shared from specialized contributions (Marczak et al., 2025). To regulate these contributions, spectral truncation and rescaling control retained components (Lee et al., 2025b), singular-value calibration reduces over-accumulation along shared directions (Li et al., 2026d), and projection-based methods reshape incompatible updates (Wei et al., 2025b; Cheng et al., 2025). Core-space representations improve computational efficiency when merging low-rank updates (Panariello et al., 2025). Further work extends subspace-based merging to sequential arrivals, improves direction robustness, preserves task-specific features through superposition, and addresses cross-domain conflicts (Tang et al., 2025b; Zeng et al., 2025; Qiu et al., 2025; Chaves et al., 2026). Parameter geometry characterizes relations among updates but does not directly measure their effects on individual task objectives.

B.2 Model Merging with Private Information

To assess the task-specific importance of updates beyond parameter geometry, existing merging methods incorporate task data and derived auxiliary information. Such information constitutes private information when it remains local to the corresponding model and is unavailable to other models. These approaches draw on three complementary forms of task evidence: task sensitivity, representation relevance, and performance benefit, with access protocols varying across methods.

Task sensitivity. Sensitivity statistics connect parameter changes to task objectives. Fisher-weighted merging estimates parameter importance from task-data gradients through Fisher information (Matena & Raffel, 2022). DF-Merge combines Fisher-based importance with merging coefficients optimized through Bayesian optimization on validation performance (Lee et al., 2025a). These sensitivity estimates help identify parameters whose changes may strongly affect the local task objective, providing a basis for prioritizing the preservation of corresponding update components.

Representation relevance. Representation statistics characterize model responses to task inputs. Activation-informed merging uses these statistics to guide parameter combination (Heyrani Nobari et al., 2025). RegMean uses representation statistics to minimize discrepancies between merged and task-specific layer outputs (Jin et al., 2023), while RegMean++ further accounts for dependencies across layers (Nguyen et al., 2026b). To address representation discrepancies, activation matching and cross-model feature correlations align representations (Ainsworth et al., 2023; Stoica et al., 2024; Horoi et al., 2024), while representation surgery and feature-drift minimization correct changes introduced by merging (Yang et al., 2024a; Sun et al., 2025b). CAT and essential-subspace merging use input representations to identify update directions to retain (Sun et al., 2025a; Li et al., 2026b). Input-dependent approaches further adapt contributions during prediction (Tang et al., 2024; Lu et al., 2024); MASS selects task subspaces from current input representations (Crisostomi et al., 2026). These signals connect parameter updates to their effects on task representations, guiding which contributions to retain, align, or adjust to preserve task-specific relevance.

Performance benefit. Task performance, empirical losses, and output-based proxies provide feedback on merging outcomes. Validation performance guides greedy model selection in Model Soups (Wortsman et al., 2022). Prediction entropy guides coefficient learning and rank selection (Yang et al., 2024b; Lee et al., 2026), while task losses support anisotropic scaling and constrained contribution optimization (Zhang et al., 2024; Chen et al., 2025). Evolutionary merging uses task evaluations to compare candidate configurations (Mencattini et al., 2025), and merging–ensembling disagreement estimates compatibility from limited unlabeled data (Wang et al., 2026). Such feedback helps determine whether candidate merging choices improve measured task performance or its proxies, providing an empirical basis for selecting and optimizing contributions. When task evidence remains local, each model must coordinate its contribution using public information and its own private information while accounting for the unobserved private characteristics of other models.

B.3 Joint Beliefs for Model Merging

Coordinating contributions under incomplete information involves assessing model compatibility, reasoning about uncertainty, and balancing task objectives. Related work provides three complementary perspectives: mergeability analysis, probabilistic inference, and contribution coordination.

Mergeability. Recent studies examine how mergeability depends on the models, tasks, and merging procedure (Zhou et al., 2026; Rahamim et al., 2026). Continued fine-tuning can improve standalone performance while degrading merged performance, motivating early stopping (Horoi et al., 2026). These findings motivate assessing model contributions beyond standalone performance. The training dynamics in Section 2 further illustrate that the value of a model's contribution depends on its interactions with the states of other models, motivating coordination of interdependent contributions.

Probabilistic inference. Probabilistic approaches connect available evidence to distributions over different inference targets. Product-of-experts merging defines energy-based distributions over merged parameters (Bui et al., 2026). Bayesian Model Merging estimates merged parameters through anchor-informed Bayesian regression, using activation statistics or parameter-derived approximations (Li et al., 2026a). Probabilistic representation surgery models uncertainty in representation discrepancies (Wei et al., 2025a), while dataset-conditioned weight generation learns distributions over neural parameters as a related parameter-generation approach (Soro et al., 2025). These formulations use probabilistic assumptions to connect observed evidence with parameter estimation or representation correction, providing a framework for inference beyond directly observed quantities.

Contribution coordination. Multi-objective learning studies trade-offs among task losses (Sener & Koltun, 2018), while gradient projection and conflict-averse optimization address competing update directions (Yu et al., 2020; Liu et al., 2021). In model merging, Pareto optimization exposes preference-dependent task trade-offs (Chen & Kwok, 2025), and flexible merging explores the accuracy–size trade-off (Dhasade et al., 2026). Nash bargaining selects cooperative agreements based on utility gains relative to disagreement outcomes (Nash, 1950). Nash-MTL applies this criterion to multi-task optimization (Navon et al., 2022), while NAMEx uses it for expert merging in sparse mixture-of-experts models (Nguyen et al., 2026a). These approaches provide explicit criteria for balancing competing objectives, allowing contribution to account for effects across tasks.

The remaining challenge is how locally observed task evidence can inform joint beliefs about the unobserved private characteristics of other models and support interdependent contribution decisions. `BeliefMerge` formulates this problem as a Bayesian game under incomplete information (Harsanyi, 1967). Each model combines public information with its own private information to form correlated joint beliefs over the unobserved private information of other models. Conditional expected utility evaluates contribution actions while accounting for joint uncertainty in the actions induced by the other models' strategies. Bayesian Nash equilibrium defines contribution balance: no unilateral change in a contribution action improves conditional expected utility when the other strategies remain fixed (Harsanyi, 1967). The resulting strategies jointly determine the merged model.

C PRIVATE INFORMATION AND JOINT BELIEF INFERENCE

This appendix details the three private-information diagnostics and derives the posterior distributions and joint sampling procedure used to infer the unobserved private information of other models.

C.1 PRIVATE-INFORMATION DIAGNOSTICS

Information source and visibility. Public information consists of shared model parameters and quantities derived solely from them. Task data and derived auxiliary information constitute private information when unavailable to other models. Model m observes public information $\boldsymbol{o}$ and its own measured private information $\boldsymbol{s}_m$, while measured private information $\boldsymbol{s}_{-m}$ from the other models remains unobserved.

Information sources. Table 4 summarizes parameter-based information and task-derived evidence used by existing merging methods and related approaches. Task evidence is organized into task sensitivity, representation relevance, and performance benefit. Categories may overlap, and entries distinguish the information used by specific components or variants, with access protocols varying across methods.

Private information in `BeliefMerge`. For task model $\boldsymbol{\theta}_m$, let $\Delta\boldsymbol{\theta}_m = \boldsymbol{\theta}_m - \boldsymbol{\theta}_0$ and let L denote the number of layers. Layer updates are vectorized unless a matrix form is specified. The diagnostics define the private information

$$\boldsymbol{s}_m = [\boldsymbol{s}_{m,\text{sens}}; \boldsymbol{s}_{m,\text{repr}}; \boldsymbol{s}_{m,\text{perf}}]. \tag{17}$$

Each diagnostic contributes one score per layer. Scores are clipped to $[\epsilon, 1-\epsilon]$, with $0 < \epsilon < 1/2$, for numerical stability. The same symbols denote the stabilized scores. All scale and stabilization constants below are positive.

Task sensitivity. Let $F_{m,j}^{(\ell)}$ denote the empirical Fisher diagonal entry for parameter j in layer ℓ, computed from $\mathcal{D}_m$ at $\boldsymbol{\theta}_m$. This entry is the mean squared log-likelihood derivative. For a layer containing d_ℓ parameters, the score is

$$s_{m,\text{sens}}^{(\ell)} = \frac{\sum_{j=1}^{d_\ell} F_{m,j}^{(\ell)} ([\Delta\boldsymbol{\theta}_m]_{\ell,j})^2}{\epsilon + \sum_{j=1}^{d_\ell} F_{m,j}^{(\ell)} ([\Delta\boldsymbol{\theta}_m]_{\ell,j})}. \tag{18}$$

The score measures normalized sensitivity-weighted update energy. By incorporating task-data gradients through Fisher information (Matena & Raffel, 2022), it distinguishes updates with similar magnitudes but different local task sensitivity.

Representation relevance. Representation relevance characterizes how an update acts on local task representations. Let $\boldsymbol{H}_m^{(\ell)}$ contain, as rows, the inputs to layer ℓ obtained by passing local task inputs through the task-specific model $\boldsymbol{\theta}_m$. Writing $[\Delta\boldsymbol{\theta}_m]_\ell$ in input-by-output matrix form gives

$$s_{m,\text{repr}}^{(\ell)} = \frac{\|\boldsymbol{H}_m^{(\ell)} [\Delta\boldsymbol{\theta}_m]_\ell\|_F^2}{\|\boldsymbol{H}_m^{(\ell)}\|_F^2 \|[\Delta\boldsymbol{\theta}_m]_\ell\|_F^2 + \epsilon}. \tag{19}$$

This expression applies to matrix-valued layer updates with compatible input representations. The score measures the normalized energy of representation changes induced by the layer update, characterizing its relevance to local task inputs.

Performance benefit. Performance benefit measures the local loss change caused by removing a layer update. Let $\mathcal{E}_m(\boldsymbol{\theta})$ denote the mean task loss on $\mathcal{D}_m$, and let $\boldsymbol{\theta}_{m,-\ell}$ denote $\boldsymbol{\theta}_m$ with layer ℓ replaced by its pretrained counterpart. Applying the sigmoid function to the normalized loss difference gives

$$s_{m,\text{perf}}^{(\ell)} = \left[1 + \exp\left(-\frac{\mathcal{E}_m(\boldsymbol{\theta}_{m,-\ell}) - \mathcal{E}_m(\boldsymbol{\theta}_m)}{|\mathcal{E}_m(\boldsymbol{\theta}_m)| + \epsilon}\right)\right]^{-1}. \tag{20}$$

Both losses use the same local examples and loss function. Scores above 0.5 indicate a positive performance benefit of retaining the layer update in the task model, with larger scores indicating a greater normalized loss increase upon removal.

Together, the three diagnostics summarize task-dependent private characteristics. Each model combines its own private information with public observations to form correlated joint beliefs over the unobserved private information of other models. Performance benefit is a diagnostic score, while the utility u_m evaluates candidate joint contributions.

Table 4: Public information and task-derived evidence used by existing merging methods.

Category	Information used	References
Public information	Shared model parameters (uniform averaging)	Wortsman et al. (2022)
Public information	Task vectors	Ilharco et al. (2023)
Public information	Task vectors	Yu et al. (2024)
Public information	Shared model parameters and task vectors	Huang et al. (2024)
Public information	Task vectors for quantization	Kim et al. (2025)
Public information	Task vectors and parameter-derived importance statistics	Du et al. (2024)
Public information	Task vectors	Yadav et al. (2023)
Public information	Task vectors	Yang et al. (2025)
Public information	Task vectors and parameter-derived subspaces	Wei et al. (2025b)
Public information	Task vectors and parameter-derived subspaces	Cheng et al. (2025)
Public information	LoRA updates and a joint singular decomposition (KnOTS)	Stoica et al. (2025)
Public information	Task vectors and singular subspaces	Gargiulo et al. (2025)
Public information	Task vectors and singular subspaces	Marczak et al. (2025)
Public information	LoRA parameters and parameter-derived subspaces	Panariello et al. (2025)
Public information	Task vectors and singular subspaces	Tang et al. (2025b)
Public information	LoRA parameters	Zeng et al. (2025)
Public information	Task vectors and singular subspaces	Qiu et al. (2025)
Public information	Task vectors and singular subspaces	Li et al. (2026d)
Public information	Task vectors and singular subspaces	Chaves et al. (2026)
Public information	Task vectors and singular subspaces	Lee et al. (2025b)
Public information	Expert domain vectors and their Gram matrix (NAMEx bargaining step)	Nguyen et al. (2026a)
Public information	Task vectors and directional residuals (PoE-EBM merging step)	Bui et al. (2026)
Public information	Shared model parameters (weight matching)	Ainsworth et al. (2023)
Public information	Task vectors and parameter-derived Gram estimates (BMM)	Li et al. (2026a)
Public information	Shared model parameters for blockwise merging	Dhasade et al. (2026)
Task sensitivity	Task-data gradients and Fisher information	Matena & Raffel (2022)
Task sensitivity	Task-data gradients and Fisher information	Lee et al. (2025a)
Representation relevance	Input representations and activation Gram statistics	Jin et al. (2023)
Representation relevance	Input representations and regression statistics accounting for cross-layer dependencies	Nguyen et al. (2026b)

Category	Information used	References
Representation relevance	Input representations and activation statistics	Heyrani Nobari et al. (2025)
Representation relevance	Activation similarities (activation matching)	Ainsworth et al. (2023)
Representation relevance	Input representations and activation correlations	Stoica et al. (2024)
Representation relevance	Input representations and activation covariance statistics	Horoi et al. (2024)
Representation relevance	Task-model and merged-model representations	Yang et al. (2024a)
Representation relevance	Task-model and merged-model representations	Wei et al. (2025a)
Representation relevance	Task-model and merged-model representations	Sun et al. (2025b)
Representation relevance	Input representations and activation statistics	Sun et al. (2025a)
Representation relevance	Input representations and update-induced activation changes	Li et al. (2026b)
Representation relevance	Input representations for input-dependent contributions	Tang et al. (2024)
Representation relevance	Input representations for input-dependent contributions	Lu et al. (2024)
Representation relevance	Input representations for input-dependent contributions	Crisostomi et al. (2026)
Representation relevance	Input-dependent expert outputs and routing frequencies	Li et al. (2026c)
Representation relevance	Input representations and activation Gram statistics (BMM)	Li et al. (2026a)
Representation relevance	Dataset embeddings (dataset-conditioned weight generation)	Soro et al. (2025)
Performance benefit	Validation performance (greedy model selection)	Wortsman et al. (2022)
Performance benefit	Validation performance for merging-coefficient optimization	Lee et al. (2025a)
Performance benefit	Task losses	Zhang et al. (2024)
Performance benefit	Model predictions and prediction entropy	Yang et al. (2024b)
Performance benefit	Model predictions and merging–ensembling losses	Wang et al. (2026)
Performance benefit	Model predictions and prediction entropy	Lee et al. (2026)
Performance benefit	Validation performance	Cao et al. (2026)
Performance benefit	Task losses and task-data gradients	Chen et al. (2025)
Performance benefit	Task losses	Chen & Kwok (2025)
Performance benefit	Task-performance evaluations	Mencattini et al. (2025)
Performance benefit	Task-performance evaluations	Rahamim et al. (2026)
Performance benefit	Validation performance for hyperparameter selection	Bui et al. (2026)
Performance benefit	Validation performance for Bayesian hyperparameter optimization (BMM)	Li et al. (2026a)

C.2 Posterior Derivations and Joint Sampling

The shared latent variable $\boldsymbol{\eta}$ captures shared variation in private information beyond public predictions. Each model uses its own private information to update the posterior over $\boldsymbol{\eta}$ and form correlated joint beliefs about the unobserved private information of other models. Following the main text, we use the Gaussian approximation $\boldsymbol{s}_m = f(\boldsymbol{o}_m) + \boldsymbol{\eta} + \boldsymbol{\varepsilon}_m$, where $\boldsymbol{\eta} \sim \mathcal{N}(\boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta)$ and $\boldsymbol{\varepsilon}_m \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\varepsilon)$. The noise terms are mutually independent and independent of $\boldsymbol{\eta}$. Public observations and the map f are fixed during inference. The following posterior expressions are exact under this Gaussian belief model.

Common prior. Marginalizing $\boldsymbol{\eta}$ gives the common prior

$$\pi(\boldsymbol{s} \mid \boldsymbol{o}) = \int p(\boldsymbol{\eta}) \prod_{m=1}^{M} p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m)\, d\boldsymbol{\eta}, \tag{21}$$

where $p(\boldsymbol{\eta}) = \mathcal{N}(\boldsymbol{\eta} \mid \boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta)$ and $p(\boldsymbol{s}_m \mid \boldsymbol{\eta}, \boldsymbol{o}_m) = \mathcal{N}(\boldsymbol{s}_m \mid f(\boldsymbol{o}_m) + \boldsymbol{\eta}, \boldsymbol{\Sigma}_\varepsilon)$. Although the private information is conditionally independent across models given $\boldsymbol{\eta}$, marginalizing $\boldsymbol{\eta}$ induces cross-model dependencies.

Posterior update from private information. For $\boldsymbol{\Sigma}_\eta, \boldsymbol{\Sigma}_\varepsilon \succ 0$, Bayes' rule gives $\pi_m(\boldsymbol{\eta} \mid \boldsymbol{s}_m, \boldsymbol{o}) = \mathcal{N}(\boldsymbol{\eta} \mid \boldsymbol{\mu}_{\eta|m}, \boldsymbol{\Sigma}_{\eta|m})$, where

$$\boldsymbol{\Sigma}_{\eta|m} = \left(\boldsymbol{\Sigma}_\eta^{-1} + \boldsymbol{\Sigma}_\varepsilon^{-1}\right)^{-1}, \quad \boldsymbol{\mu}_{\eta|m} = \boldsymbol{\Sigma}_{\eta|m}\left[\boldsymbol{\Sigma}_\eta^{-1}\boldsymbol{\mu}_\eta + \boldsymbol{\Sigma}_\varepsilon^{-1}\big(\boldsymbol{s}_m - f(\boldsymbol{o}_m)\big)\right]. \tag{22}$$

Its unique mode is $\boldsymbol{\mu}_{\eta|m}$, and its covariance satisfies $0 \prec \boldsymbol{\Sigma}_{\eta|m} \prec \boldsymbol{\Sigma}_\eta$. For $j \neq m$, the predictive distribution is

$$\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o} \sim \mathcal{N}\big(f(\boldsymbol{o}_j) + \boldsymbol{\mu}_{\eta|m},\, \boldsymbol{\Sigma}_{\eta|m} + \boldsymbol{\Sigma}_\varepsilon\big). \tag{23}$$

The change in the predicted mean is

$$\mathbb{E}[\boldsymbol{s}_j \mid \boldsymbol{s}_m, \boldsymbol{o}] - \mathbb{E}[\boldsymbol{s}_j \mid \boldsymbol{o}] = \boldsymbol{\Sigma}_\eta(\boldsymbol{\Sigma}_\eta + \boldsymbol{\Sigma}_\varepsilon)^{-1}\big(\boldsymbol{s}_m - f(\boldsymbol{o}_m) - \boldsymbol{\mu}_\eta\big). \tag{24}$$

The term in parentheses measures the deviation of model m's observed private information from its prior mean. Through the shared latent variable, local evidence adjusts predictions of other models' private information.

Derivation. Let $\boldsymbol{r}_m = \boldsymbol{s}_m - f(\boldsymbol{o}_m)$. Up to an additive constant, the negative log-posterior density is

$$\frac{1}{2}(\boldsymbol{\eta} - \boldsymbol{\mu}_\eta)^\top \boldsymbol{\Sigma}_\eta^{-1}(\boldsymbol{\eta} - \boldsymbol{\mu}_\eta) + \frac{1}{2}(\boldsymbol{r}_m - \boldsymbol{\eta})^\top \boldsymbol{\Sigma}_\varepsilon^{-1}(\boldsymbol{r}_m - \boldsymbol{\eta}). \tag{25}$$

The two terms correspond to the prior and the likelihood of the locally observed private information. The Hessian $\boldsymbol{\Sigma}_\eta^{-1} + \boldsymbol{\Sigma}_\varepsilon^{-1} \succ 0$ ensures a unique minimizer, yielding the posterior mean in Eq. (22). The posterior precision exceeds the prior precision in the positive-definite order, yielding the covariance bound. Adding independent noise $\boldsymbol{\varepsilon}_j$ gives Eq. (23). Finally, the identity

$$\boldsymbol{\Sigma}_{\eta|m}\boldsymbol{\Sigma}_\varepsilon^{-1} = \boldsymbol{\Sigma}_\eta(\boldsymbol{\Sigma}_\eta + \boldsymbol{\Sigma}_\varepsilon)^{-1}$$

gives the mean shift in Eq. (24).

Joint sampling for contribution decisions. The unobserved private information remains dependent across models through the shared latent variable. For distinct $j, k \neq m$, the centered predictive variables satisfy

$$\mathbb{E}\big[(\boldsymbol{s}_j - f(\boldsymbol{o}_j) - \boldsymbol{\mu}_{\eta|m})(\boldsymbol{s}_k - f(\boldsymbol{o}_k) - \boldsymbol{\mu}_{\eta|m})^\top \mid \boldsymbol{s}_m, \boldsymbol{o}\big] = \boldsymbol{\Sigma}_{\eta|m}. \tag{26}$$

Each joint sample uses one posterior draw of $\boldsymbol{\eta}$ for all other models and independent draws of their model-specific noise, giving $\boldsymbol{s}_j = f(\boldsymbol{o}_j) + \boldsymbol{\eta} + \boldsymbol{\varepsilon}_j$ for each $j \neq m$. This procedure samples the correlated joint belief $\pi_m(\boldsymbol{s}_{-m} \mid \boldsymbol{s}_m, \boldsymbol{o})$. Sampling models independently from their marginal predictive distributions would discard this dependence. The sampled private information is passed through the strategies of the other models to estimate the conditional expected utility defined in the main text, which guides contribution actions under the Bayesian Nash equilibrium criterion.

C.3 Additional Evidence for Information Ablation

We define the two additional information sources used in the information ablation study.

Layerwise directional compatibility. Motivated by task-vector analysis (Ortiz-Jimenez et al., 2023; Ilharco et al., 2023), we measure the directional agreement between each model's task vector and the other task vectors at layer ℓ:

$$c_m^{(\ell)} = \frac{1}{M-1} \sum_{j \neq m} \frac{\langle [\Delta \boldsymbol{\theta}_m]_\ell, [\Delta \boldsymbol{\theta}_j]_\ell \rangle}{\|[\Delta \boldsymbol{\theta}_m]_\ell\|_2 \|[\Delta \boldsymbol{\theta}_j]_\ell\|_2 + \epsilon}. \tag{27}$$

Layer updates are vectorized for this calculation, and $\epsilon > 0$ stabilizes the denominator. Since these scores depend only on shared parameters, they constitute public information. The $+\boldsymbol{c}_m$ variant concatenates $\boldsymbol{c}_m = [c_m^{(1)}, \ldots, c_m^{(L)}]^\top$ with the original public observation, giving $\boldsymbol{o}_m = [g_m; \boldsymbol{z}_m; \boldsymbol{c}_m]$.

Local gradient stability. Motivated by gradient agreement in multi-task optimization (Liu et al., 2021), we define a score that measures gradient consistency across local mini-batches:

$$s_{m,\mathrm{stab}}^{(\ell)} = \frac{\left\| \mathbb{E}_B \left[\nabla_{[\boldsymbol{\theta}_m]_\ell} \mathcal{E}_B(\boldsymbol{\theta}_m) \right] \right\|_2^2}{\mathbb{E}_B \left[\left\| \nabla_{[\boldsymbol{\theta}_m]_\ell} \mathcal{E}_B(\boldsymbol{\theta}_m) \right\|_2^2 \right] + \epsilon}. \tag{28}$$

B is a mini-batch drawn from the local dataset $\mathcal{D}_m$, $\mathcal{E}_B$ is its mean loss, and the expectations are over local mini-batches. The score compares the squared norm of the mean gradient with the mean squared gradient norm. Since its computation requires local task data, it constitutes private information. The $+\boldsymbol{s}_{m,\mathrm{stab}}$ variant augments $\boldsymbol{s}_m$ with these layerwise scores.

D MERGING STRATEGIES AND BAYESIAN NASH EQUILIBRIUM

D.1 MERGING STRATEGIES

Fix the public information $\boldsymbol{o}$ and task vectors. Given observed or sampled private information, each strategy produces $b_m = \sigma_m(\boldsymbol{s}_m, \boldsymbol{o})$, determining the merging weights and model:

$$\alpha_m(\boldsymbol{b}) = \frac{\exp(b_m)}{\sum_{j=1}^{M} \exp(b_j)}, \qquad \boldsymbol{\theta}(\boldsymbol{b}) = \boldsymbol{\theta}_0 + \sum_{m=1}^{M} \alpha_m(\boldsymbol{b}) \Delta\boldsymbol{\theta}_m. \tag{29}$$

Let $\boldsymbol{P} = \boldsymbol{U}\boldsymbol{U}^\top$ and cache

$$K_{ij} = \Delta\boldsymbol{\theta}_i^\top \Delta\boldsymbol{\theta}_j, \qquad K_{ij}^{\mathrm{svd}} = \Delta\boldsymbol{\theta}_i^\top \boldsymbol{P} \Delta\boldsymbol{\theta}_j. \tag{30}$$

With $\boldsymbol{\alpha} = \boldsymbol{\alpha}(\boldsymbol{b})$, the main-text utility Eq. (14) is exactly

$$u_m(\boldsymbol{b}) = -\lambda_t \frac{\boldsymbol{\alpha}^\top \boldsymbol{K}\boldsymbol{\alpha} - 2(\boldsymbol{K}\boldsymbol{\alpha})_m + K_{mm}}{K_{mm} + \epsilon} + \lambda_o \frac{\boldsymbol{\alpha}^\top \boldsymbol{K}^{\mathrm{svd}} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^\top \boldsymbol{K}\boldsymbol{\alpha} + \epsilon} - \lambda_b b_m^2. \tag{31}$$

Dependence on fixed public information is suppressed. Conditional utilities average this expression over the joint posterior of $\boldsymbol{s}_{-m}$, with $b_j = \sigma_j(\boldsymbol{s}_j, \boldsymbol{o})$ for $j \neq m$.

Evaluating the utilities at one action profile costs $O(M^2)$ using the cached matrices, without accessing high-dimensional task vectors or running task models. With other actions fixed, both quadratic forms are scalar quadratics in α_m, allowing $O(1)$ candidate evaluations after caching their coefficients. Evaluating T candidates per model against a common fixed profile therefore costs $O(M^2 + MT)$. These costs exclude belief inference and strategy-network computation; preprocessing and final assembly remain dependent on the parameter dimension d.

D.2 BEST RESPONSE

Fix the belief model, $\boldsymbol{s}_m$, $\boldsymbol{o}$, and the other strategies σ_{-m}. The conditional best-response set is

$$\mathrm{BR}_m(\sigma_{-m}; \boldsymbol{s}_m, \boldsymbol{o}) = \arg\max_{a \in \mathbb{R}} \bar{u}_m(a, \sigma_{-m}; \boldsymbol{s}_m, \boldsymbol{o}). \tag{32}$$

a is a candidate value of b_m and $\bar{u}_m$ is the main-text conditional expected utility. The fixed arguments are suppressed below.

Write $h_m(\boldsymbol{b}) = u_m(\boldsymbol{b}) + \lambda_b b_m^2$. For $\lambda_t, \lambda_o \geq 0$ and $\epsilon > 0$, the convex combination of task vectors and the orthogonal projection give $|h_m(\boldsymbol{b})| \leq \lambda_o$ and equality requires $\boldsymbol{U}$ is square matrix and $\varepsilon = 0$. This action-independent bound also bounds the conditional expectation of h_m. For $\lambda_b > 0$,

$$\bar{u}_m(a) \leq \lambda_o - \lambda_b a^2 \longrightarrow -\infty \qquad \text{as } |a| \longrightarrow \infty. \tag{33}$$

Continuity and this tail bound ensure a best response exists. For any best response $a^\star$, optimality relative to the feasible action 0 gives

$$-\lambda_o \leq \bar{u}_m(0) \leq \bar{u}_m(a^\star) \leq \lambda_o - \lambda_b (a^\star)^2. \tag{34}$$

Thus every best response lies in $[-B, B]$, where $B = \sqrt{2\lambda_o/\lambda_b}$. The comparison does not set $a^\star = 0$; the bound concerns optimal actions, not arbitrary candidates, and requires no concavity assumption.

Uniqueness of the conditional best response. Define

$$C_{mm} = \sup_{\boldsymbol{b} \in \mathbb{R}^M} \left| \frac{\partial^2 h_m(\boldsymbol{b})}{\partial b_m^2} \right| < \infty. \tag{35}$$

Let $D_\theta^2 = \max_{i,j}(K_{ii} + K_{jj} - 2K_{ij})$. Softmax derivative bounds and the orthogonal projection property of $\boldsymbol{P}$ give, by the chain rule,

$$|\partial^2_{b_i b_j} h_m(\boldsymbol{b})| \leq \overline{C}_m := \frac{9\lambda_t D_\theta^2}{8(K_{mm} + \epsilon)} + \lambda_o \left(\frac{D_\theta^2}{4\epsilon} + \frac{D_\theta}{2\sqrt{\epsilon}} \right), \tag{36}$$

uniformly over all actions and indices i, j. Thus $C_{mm} \le \overline{C}_m$, and the same bound holds for C_{mj} in Eq. (43). Since

$$\bar{u}''_m(a) = \mathbb{E}_{\pi_m}[\partial^2_{b_m} h_m] - 2\lambda_b, \tag{37}$$

it gives instruction of the choice of λ_b. We require

$$\lambda_b > \frac{1}{2} C_{mm}, \forall m = 1, \dots, M \tag{38}$$

so the conditional best response is unique as $\bar{u}''_m(a)$ in Eq. (37) is then strictly negative. A tighter one, $\lambda_b > \frac{1}{2}\mathbb{E}_{\pi_m}[\partial^2_{b_m} h_m]$, also works apparently. This is a sufficient condition on the utility coefficients and task vectors. Consequently, $2\lambda_b > M\overline{C}_m$ for every m is a computable sufficient condition for Eq. (44).

Numerical responses. Scalar response search can be restricted to $[-B, B]$, with conditional utilities estimated from joint posterior samples and Eq. (31). Write $\mu_m := 2\lambda_b - C_{mm} > 0$. Therefore, any approximate response $\widehat{a}_m$ whose true conditional utility gap, defined as

$$\bar{u}_m(a^\star_m; \sigma_{-m}, s_m, o) - \bar{u}_m(\hat{a}_m; \sigma_{-m}, s_m, o)$$

is at most δ_m satisfies

$$|\widehat{a}_m - a^\star| \le \sqrt{2\delta_m/\mu_m}. \tag{39}$$

D.3 Optimization

Belief calibration. For observed calibration pairs $\mathcal{D}_{\text{cal}}$ (M pairs; f_ψ is a small MLP shared across models and trained with weight decay to mitigate overfitting) and fixed Gaussian parameters $\boldsymbol{\mu}_\eta$, $\boldsymbol{\Sigma}_\eta$, and $\boldsymbol{\Sigma}_\epsilon$, fit the predictor f_ψ by minimizing the marginal negative log-likelihood

$$\mathcal{L}_{\text{belief}}(\psi) = -\frac{1}{|\mathcal{D}_{\text{cal}}|} \sum_{(\boldsymbol{o}_m, \boldsymbol{s}_m) \in \mathcal{D}_{\text{cal}}} \log \mathcal{N}(\boldsymbol{s}_m; f_\psi(\boldsymbol{o}_m) + \boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta + \boldsymbol{\Sigma}_\epsilon)\,. \tag{40}$$

The fitted predictor and Gaussian parameters are then fixed; conditional beliefs follow from the main-text posterior formulas.

Cyclic strategy optimization. Parameterize each strategy as $\sigma_m(\boldsymbol{s}_m, \boldsymbol{o}; \boldsymbol{w}_m)$ where $\boldsymbol{w}_m$ collects the parameters to be optimized. Let q_m be the marginal distribution of $\boldsymbol{s}_m$ under the calibrated common prior. Writing $b_j = \sigma_j(\boldsymbol{s}_j, \boldsymbol{o}; \boldsymbol{w}_j)$, the objective is

$$J_m(\boldsymbol{w}_m; \boldsymbol{w}_{-m}) = \mathbb{E}_{q_m}[\bar{u}_m(b_m, \sigma_{-m}; \boldsymbol{s}_m, \boldsymbol{o})] = \mathbb{E}_{\pi(\cdot|\boldsymbol{o})}[u_m(\boldsymbol{b})]. \tag{41}$$

The equality follows from total expectation. Cycle through models $1, \dots, M$, fixing all other strategy parameters during each update. For model m, sample $\boldsymbol{s}_m \sim q_m$ and $\boldsymbol{s}_{-m} \sim \pi_m(\cdot \mid \boldsymbol{s}_m, \boldsymbol{o})$. Each joint posterior sample uses one shared latent draw and independent model-specific noise. With fresh batches, take a prescribed number of AdamW steps on $-\widehat{J}_m$ with respect to $\boldsymbol{w}_m$ only, where

$$\widehat{J}_m = \frac{1}{S} \sum_{r=1}^{S} u_m(\boldsymbol{b}^{(r)}), \qquad b_j^{(r)} = \sigma_j(\boldsymbol{s}_j^{(r)}, \boldsymbol{o}; \boldsymbol{w}_j). \tag{42}$$

Later models use the latest updated strategies. These updates seek approximate conditional best responses; finite network training need not attain exact responses.

Model construction. After the prescribed training cycles, evaluate $\widehat{b}_m = \sigma_m(\boldsymbol{s}_m, \boldsymbol{o}; \widehat{\boldsymbol{w}}_m)$ at each model's observed private information and construct $\widehat{\boldsymbol{\theta}} = \boldsymbol{\theta}(\widehat{\boldsymbol{b}})$ using Eq. (29).

D.4 Nash Equilibrium

Fix the public information, task vectors, and calibrated common prior. Let $\mathcal{S} = \prod_m L^2(q_m)$ be the Hilbert space of square-integrable functions under the distribution q_m, with norm $\|\sigma\|_{\mathcal{S}} =$

$\max_m \|\sigma_m\|_{L^2(q_m)}$. All conditional beliefs come from this prior; correlated private types are allowed. Define

$$C_{mj} = \sup_{\boldsymbol{b}\in\mathbb{R}^M} \left| \frac{\partial^2 h_m(\boldsymbol{b})}{\partial b_m \, \partial b_j} \right| < \infty \qquad (j \neq m). \tag{43}$$

By proper choice of the value of λ_b, we ensure the utility coefficients and task vectors satisfy

$$\mu_m = 2\lambda_b - C_{mm} > 0 \quad \text{for all } m, \qquad \kappa = \max_m \frac{\sum_{j\neq m} C_{mj}}{\mu_m} < 1. \tag{44}$$

Define

$$(\mathcal{T}_m\sigma)(\boldsymbol{s}_m, \boldsymbol{o}) = \arg\max_{a\in\mathbb{R}} \bar{u}_m(a, \sigma_{-m}; \boldsymbol{s}_m, \boldsymbol{o}). \tag{45}$$

The unique conditional maximizer is measurable and bounded by B, so $\mathcal{T}$ maps $\mathcal{S}$ into itself.

Existence and uniqueness. Optimality, strong concavity from Eq. (37), and the cross-action derivative bounds imply, with fixed $\boldsymbol{o}$ suppressed,

$$|(\mathcal{T}_m\sigma)(\boldsymbol{s}_m) - (\mathcal{T}_m\widetilde{\sigma})(\boldsymbol{s}_m)| \leq \sum_{j\neq m} \frac{C_{mj}}{\mu_m} \mathbb{E}[|\sigma_j(\boldsymbol{s}_j) - \widetilde{\sigma}_j(\boldsymbol{s}_j)| \mid \boldsymbol{s}_m, \boldsymbol{o}]\,. \tag{46}$$

In above, we use $\sigma_j^t = (1-t)\sigma_j + t\widetilde{\sigma}_j$, $t \in [0,1]$, for $j \neq m$ in conjunction with implicit function theorem for the $\arg\max$ in Eq. (45). Writing $\|\cdot\|_m = \|\cdot\|_{L^2(q_m)}$, conditional Jensen's inequality and the common-prior identity give

$$\|\mathcal{T}_m\sigma - \mathcal{T}_m\widetilde{\sigma}\|_m \leq \sum_{j\neq m} \frac{C_{mj}}{\mu_m} \|\sigma_j - \widetilde{\sigma}_j\|_j \leq \kappa \|\sigma - \widetilde{\sigma}\|_{\mathcal{S}}. \tag{47}$$

Taking the maximum over m makes $\mathcal{T}$ a κ-contraction on $\mathcal{S}$. Banach's theorem yields a fixed point satisfying the conditional best-response inequalities, hence a pure-strategy BNE. Every pure-strategy BNE is bounded by B and belongs to $\mathcal{S}$. For any two equilibria,

$$\|\sigma^\star - \widetilde{\sigma}^\star\|_{\mathcal{S}} \leq \kappa \|\sigma^\star - \widetilde{\sigma}^\star\|_{\mathcal{S}}. \tag{48}$$

Since $\kappa < 1$, they coincide almost surely, establishing uniqueness.

Cyclic exact updates. Replace each strategy in order by its conditional best response to the latest other strategies. Each update is nonexpansive and reduces its coordinate difference to at most κ times the difference at the start of the cycle. After all models update, the full cycle is a contraction. Thus, with k counting complete cycles,

$$\|\sigma^{(k)} - \sigma^\star\|_{\mathcal{S}} \leq \kappa^k \|\sigma^{(0)} - \sigma^\star\|_{\mathcal{S}}. \tag{49}$$

Approximate updates. Here we concern the update from our cyclic strategy optimization, which is not exact $\arg\max$ but an approximate, *i.e.* suboptimal. Let $\delta_{m,k}$ be the true conditional utility gap, averaged over q_m, between model m's approximate update and its exact best response to the other strategies present at that update. Squaring Eq. (39) and averaging bounds the $L^2(q_m)$ strategy error by $\sqrt{2\delta_{m,k}/\mu_m}$. Set $D_k = \|\widehat{\sigma}^{(k)} - \sigma^\star\|_{\mathcal{S}}$ and $e_k = \sum_m \sqrt{2\delta_{m,k}/\mu_m}$. Nonexpansiveness of individual updates then gives

$$D_{k+1} \leq \kappa D_k + e_k. \tag{50}$$

Consequently, $e_k \to 0$ implies convergence, while $e_k \leq \bar{e}$ gives $\limsup_k D_k \leq \bar{e}/(1-\kappa)$, where $\bar{e}$ is the uniform upper bound on the per-iteration approximation error e_k. These guarantees require the stated curvature and error conditions; finite-step AdamW and small empirical training loss do not establish them.

Under Eq. (44), constant strategies form a closed invariant subspace of $\mathcal{T}$. The contraction restricted to this subspace has a fixed point, which must coincide with the unique BNE. Hence this exact equilibrium is constant in private information almost surely, although equilibrium actions may differ across models. This conclusion concerns the exact equilibrium limit under the stated sufficient conditions. The finite-step strategy outputs described in Section D.3 may depend on private information; their approximation error is governed by Eq. (50) when its curvature and true utility-gap conditions hold.

E Dataset Splits and Metrics

This section details the dataset splits and evaluation metrics for the CLIP and GLUE experiments. Average absolute performance is computed as the unweighted mean of the task scores. For each task, the normalized score is the merged-model score divided by the corresponding fine-tuned model score and multiplied by 100.

E.1 CLIP

We evaluate CLIP with ViT-B-32, ViT-B-16, and ViT-L-14 on nested suites of 8, 14, and 20 vision tasks, yielding nine configurations (Gargiulo et al., 2025; Crisostomi et al., 2026). The 14-task suite extends the eight-task suite with six additional datasets, and the 20-task suite adds another six. These configurations evaluate merging performance as the number and diversity of tasks increase.

Dataset splits. Table 5 reports the benchmark partitions documented by MASS (Crisostomi et al., 2026). Except for the prescribed EuroSAT split, validation sets contain $\min(\lfloor 0.1N \rfloor, 5000)$ examples, where N is the designated training-pool size before the validation holdout. The counts refer to the adopted benchmark partitions. For DTD, Flowers-102, PCAM, and Rendered SST-2, validation examples are held out from the designated training pools. EMNIST refers to the Digits subset, and Rendered SST-2 contains rendered text images for sentiment classification.

Evaluation metrics. Each task is evaluated using classification accuracy on its test split. Average absolute accuracy is the unweighted mean across tasks in the corresponding suite. Task-wise normalized accuracy expresses performance relative to the corresponding fine-tuned model.

Table 5: CLIP dataset splits. The 14- and 20-task suites include all datasets in the preceding groups.

	Dataset	Train	Validation	Test
8 tasks	Cars	7,330	814	8,041
	DTD	1,692	188	1,880
	EuroSAT	21,600	2,700	2,700
	GTSRB	23,976	2,664	12,630
	MNIST	55,000	5,000	10,000
	RESISC45	17,010	1,890	6,300
	SUN397	17,865	1,985	19,850
	SVHN	68,257	5,000	26,032
14 tasks	CIFAR-100	45,000	5,000	10,000
	STL-10	4,500	500	8,000
	Flowers-102	918	102	6,149
	Oxford-IIIT Pet	3,312	368	3,669
	PCAM	257,144	5,000	32,768
	FER2013	25,839	2,870	7,178
20 tasks	EMNIST-Digits	235,000	5,000	40,000
	CIFAR-10	45,000	5,000	10,000
	Food-101	70,750	5,000	25,250
	Fashion-MNIST	55,000	5,000	10,000
	Rendered SST-2	6,228	692	1,821
	KMNIST	55,000	5,000	10,000

E.2 GLUE

We merge eight task-specific models independently fine-tuned from the same Flan-T5-Base model on CoLA, MNLI, MRPC, QNLI, QQP, RTE, SST-2, and STS-B (Wang et al., 2019; Crisostomi et al., 2026). STS-B is a regression task, while the remaining seven are classification tasks.

Dataset splits. Table 6 reports the training and validation split sizes. Following the Flan-T5 evaluation protocol in FusionBench, evaluation uses the matched validation split for MNLI and the respective validation splits for the other tasks (Tang et al., 2025a). Thus, the validation column lists the evaluation-set sizes.

Evaluation metrics. Following FusionBench, the seven classification tasks use exact-match accuracy on generated labels, while STS-B uses Spearman rank correlation ρ (Tang et al., 2025a). All scores are multiplied by 100 for reporting, and their unweighted mean gives the average task score across the eight tasks. Each task-wise normalized score uses the corresponding fine-tuned model as its reference. This evaluation follows the model-merging benchmark protocol and differs from the official GLUE leaderboard convention.

Table 6: GLUE dataset splits.

Task	Train	Validation
CoLA	8,551	1,043
MNLI	392,702	9,815
MRPC	3,668	408
QNLI	104,743	5,463
QQP	363,846	40,430
RTE	2,490	277
SST-2	67,349	872
STS-B	5,749	1,500

F TIME COMPLEXITY ANALYSIS

We analyze the cost of merging M task models with L compatible matrix layers of size $h \times h$. These layers are assumed to dominate the parameter count, giving $d = \Theta(Lh^2)$. We assume $r \leq M \leq h^2$ and use standard dense matrix operations, including $\mathcal{O}(h^3)$ for an $h \times h$ SVD. For `BeliefMerge`, the analysis includes preprocessing, cyclic strategy optimization, response assessment, and final model assembly. Task-data processing and belief-model calibration are excluded; posterior sampling and strategy-network computation are accounted for separately.

Task Arithmetic. Task Arithmetic forms the merged update by summing weighted task vectors (Ilharco et al., 2023). Accumulating contributions from M models costs $\mathcal{O}(Mh^2)$ per layer. Computing the task vectors has the same order of cost, yielding

$$\boxed{\text{Task Arithmetic}: \quad \mathcal{O}(MLh^2).} \tag{51}$$

TSV-M. TSV-M decomposes task matrices, processes the retained singular vectors, and reconstructs the merged update (Gargiulo et al., 2025). The M task-matrix SVDs cost $\mathcal{O}(Mh^3)$ per layer. If $k \leq Mh$ components are retained in total, the concatenated singular-vector matrices have size $h \times k$. Their economy-size SVDs cost $\mathcal{O}(hk \min\{h, k\})$, and reconstruction costs $\mathcal{O}(h^2 k)$. Both are bounded by $\mathcal{O}(Mh^3)$, giving

$$\boxed{\text{TSV-M}: \quad \mathcal{O}(MLh^3).} \tag{52}$$

Iso-CTS. Iso-CTS sums the task matrices and decomposes the result to identify common directions (Marczak et al., 2025), costing $\mathcal{O}(Mh^2 + h^3)$ per layer. For a common basis with $c \leq h$ directions, constructing task-specific residuals costs $\mathcal{O}(Mh^2 c)$. Their M dense SVDs cost $\mathcal{O}(Mh^3)$. Under the prescribed rank allocation, the retained factors have at most h columns, so subsequent orthogonalization and reconstruction cost $\mathcal{O}(h^3)$. The residual decompositions determine the overall order:

$$\boxed{\text{Iso-CTS}: \quad \mathcal{O}(MLh^3).} \tag{53}$$

MASS. The offline preprocessing of MASS includes task-matrix decomposition, redundancy filtering, and construction of fixed merged components (Crisostomi et al., 2026). Dense decompositions cost $\mathcal{O}(MLh^3)$. Redundancy filtering compares candidate updates with previously accepted updates using cosine similarity, with $\mathcal{O}(h^2)$ operations per comparison. In the worst case, $M(M-1)/2$ comparisons are performed per layer, giving $\mathcal{O}(LM^2h^2)$. TSV-M assembly over the accepted updates remains within $\mathcal{O}(MLh^3)$. Thus,

$$\boxed{\text{MASS}: \quad \mathcal{O}(MLh^3 + LM^2h^2).} \tag{54}$$

This bound covers offline merging and excludes input-dependent routing and adaptive merging during prediction.

`BeliefMerge`. Following Appendix D.1, let $\boldsymbol{V} = [\Delta\boldsymbol{\theta}_1, \ldots, \Delta\boldsymbol{\theta}_M] \in \mathbb{R}^{d \times M}$. The reduced utility uses the global matrices

$$\boldsymbol{K} = \boldsymbol{V}^\top \boldsymbol{V}, \qquad \boldsymbol{Q} = \boldsymbol{U}^\top \boldsymbol{V}, \qquad \boldsymbol{K}^{\text{svd}} = \boldsymbol{Q}^\top \boldsymbol{Q}. \tag{55}$$

Constructing $\boldsymbol{K}$ costs $\mathcal{O}(dM^2)$. The Gram matrix of normalized task vectors is obtained by diagonal rescaling of $\boldsymbol{K}$, and its eigendecomposition costs $\mathcal{O}(M^3)$. Recovering the shared basis and computing $\boldsymbol{Q}$ cost $\mathcal{O}(dMr)$. Forming $\boldsymbol{K}^{\text{svd}}$ costs $\mathcal{O}(M^2 r)$, which is absorbed by $\mathcal{O}(M^3)$ because $r \leq M$. The full $d \times d$ projection matrix is never constructed.

After preprocessing, utilities and their action derivatives are evaluated using these cached statistics, costing $\mathcal{O}(M^2)$ per joint action profile. With the other models' actions fixed, both quadratic forms in the reduced utility are scalar quadratics in the selected model's softmax weight. Caching their coefficients therefore reduces subsequent candidate-action evaluations to $\mathcal{O}(1)$. This saving applies to scalar response search with fixed opponents.

Let N_{eval} count the total joint action profiles evaluated during strategy optimization and response assessment, including posterior batches and repeated evaluations. Let C_{post} denote the total cost of

posterior sampling and its setup, and C_{strat} the total cost of strategy-network forward and backward passes and optimizer updates. Utility evaluation and differentiation cost $\mathcal{O}(N_{\text{eval}}M^2)$. Final model assembly requires $\mathcal{O}(dM)$, giving

$$\mathcal{O}\big(dM^2 + M^3 + dMr + N_{\text{eval}}M^2 + dM\big) + C_{\text{post}} + C_{\text{strat}}. \tag{56}$$

Thus, utility evaluation and differentiation do not access high-dimensional model parameters after preprocessing. High-dimensional task vectors are required for preprocessing and final assembly.

For fixed optimization and evaluation budgets per model, $N_{\text{eval}} = \mathcal{O}(M)$. Using $d = \Theta(Lh^2)$ and $r \le M \le h^2$, the parameter-algebra cost simplifies to

$$\boxed{\texttt{BeliefMerge}: \quad \mathcal{O}(LM^2h^2).} \tag{57}$$

This simplification describes parameter-algebra scaling under fixed budgets. Posterior sampling, strategy-network computation, and increases in optimization or response-assessment budgets remain explicit in Eq. (56).

Memory and prediction. Let S_{aux} denote storage for belief and strategy networks, optimizer states, posterior batches, and peak temporary workspace, including activations and numerical buffers. Resident task updates, the shared basis, reduced matrices, and auxiliary storage require

$$\mathcal{O}(dM + dr + M^2) + S_{\text{aux}} = \mathcal{O}(dM + M^2) + S_{\text{aux}} \tag{58}$$

scalars. Cyclic optimization updates the current strategies without storing an action history. Final assembly produces a single model with the original architecture and per-input inference cost.

Empirical runtime and memory. Following prior efficiency evaluations (Nguyen et al., 2026b; Qiu et al., 2026), Table 7 summarizes the reported merging time and peak GPU memory on the eight-task CLIP ViT-B-32 benchmark using a single NVIDIA A100 GPU with 80 GB memory. In these measurements, `BeliefMerge` takes 50.14 s and uses 4.45 GB of peak GPU memory. Compared with MASS, this corresponds to 1.81 s (3.7%) additional merging time and 29.0% lower peak memory. The reported merging time is also 20.3% lower than that of Iso-CTS.

Table 7: **Merging efficiency.**

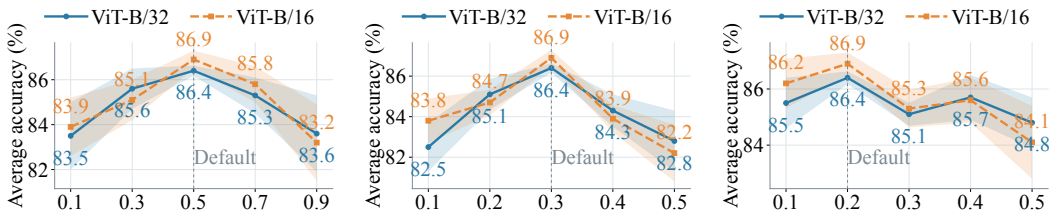
Method	Merging Time (s)	GPU Memory (GB)
Weight Averaging	0.06	3.26
Task Arithmetic	0.08	3.91
TIES-Merging	1.20	3.64
TSV-M	40.64	3.59
Iso-C	4.29	3.92
Iso-CTS	62.92	3.92
MASS	48.33	6.27
`BeliefMerge`	**50.14**	**4.45**

G HYPERPARAMETER ANALYSIS

We analyze the utility coefficients and shared-subspace rank of `BeliefMerge` on the 20-task CLIP benchmark with ViT-B-32 and ViT-B-16. The positive utility coefficients satisfy $\lambda_t + \lambda_o + \lambda_b = 1$, with reference values $(\lambda_t, \lambda_o, \lambda_b) = (0.5, 0.3, 0.2)$. In each coefficient sweep, the coefficient under investigation is fixed to the specified value, while the remaining two are adjusted accordingly to maintain the sum-to-one constraint. Contribution strategies are reoptimized for every setting, with the shared-subspace rank fixed at $r = 4$. In the rank sweep, the three utility coefficients are kept at their reference values. All other training settings are unchanged. Curves report mean absolute accuracy across the 20 tasks, averaged over five independent runs, with shaded regions indicating one standard deviation across runs.

G.1 BALANCE AMONG THE UTILITY TERMS

The three coefficients control task-vector preservation, shared-subspace alignment, and action regularization in Eq. (14). Each sweep evaluates the effect of varying one coefficient while the remaining two are adjusted accordingly under the sum-to-one constraint and the contribution strategies are reoptimized for each setting. Public and private information inputs are retained throughout, complementing the information ablations in Section 4.3.



(a) Effect of λ_t. (b) Effect of λ_o. (c) Effect of λ_b.

Figure 8: Effect of the utility weights in BeliefMerge.

Task-vector preservation. The first utility term penalizes the normalized distance between the merged update and each model's task vector. We vary $\lambda_t \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Both backbones attain their highest tested mean accuracy at $\lambda_t = 0.5$, reaching 86.4% and 86.9%, respectively (Fig. 8(a)). Reducing λ_t to 0.1 incurs losses of 2.9 and 3.0 percentage points, while increasing it to 0.9 incurs losses of 2.8 and 3.7 points. Under the sum constraint, a larger λ_t leaves less total weight for alignment and action regularization. The interior optimum is consistent with balancing task-vector fidelity against these complementary objectives.

Shared-subspace alignment. The second utility term rewards the fraction of merged-update energy captured by the shared basis U. We vary $\lambda_o \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. Both backbones peak at $\lambda_o = 0.3$ (Fig. 8(b)). At $\lambda_o = 0.1$, accuracy falls to 82.5% and 83.8%; at $\lambda_o = 0.5$, it falls to 82.8% and 82.2%. These results favor a moderate allocation to shared geometry while preserving the balance among the remaining utility terms. The sweep retains z_m in the public observations, assessing the emphasis placed on the alignment reward separately from removing public-direction information.

Action regularization. The third utility term penalizes the squared actions b_m^2 that parameterize the softmax weights in Eq. (12). Larger penalties favor actions closer to zero; when all actions approach zero, the merging weights approach $1/M$. Regularization therefore balances limiting extreme contribution requests with retaining flexible model-specific weights. Both backbones achieve their highest tested mean accuracy at $\lambda_b = 0.2$ (Fig. 8(c)). Using $\lambda_b = 0.1$ reduces accuracy by 0.9 and 0.7 percentage points, whereas $\lambda_b = 0.5$ reduces it by 1.6 and 2.8 points. These results are consistent with a moderate action penalty.

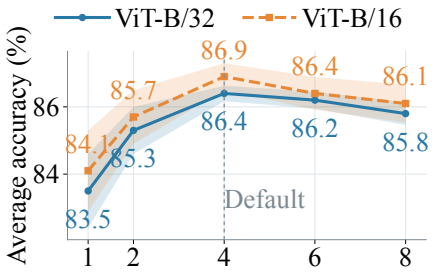


Figure 9: Effect of the shared subspace rank r.

G.2 Shared-Subspace Rank

The rank r determines the retained singular directions in U, affecting both the public coordinates z_m used for belief inference and the shared-subspace alignment reward. We evaluate $r \in \{1, 2, 4, 6, 8\}$, while reoptimizing the contribution strategies for each rank and keeping the utility coefficients at their reference values. Increasing r from 1 to 4 improves accuracy by 2.9 and 2.8 percentage points on ViT-B-32 and ViT-B-16, respectively (Fig. 9). Increasing it further to 8 brings no additional gain, with decreases of 0.6 and 0.8 points. Accuracy remains within 0.8 points of the best tested mean over $r \in \{4, 6, 8\}$, supporting $r = 4$ as a compact and effective choice.

The rank also controls the directional selectivity of the alignment reward. If U spans all task vectors, every merged update in Eq. (13) lies in that span, and the reward becomes $\|\Delta\theta(b)\|_2^2/(\|\Delta\theta(b)\|_2^2+\epsilon)$, which is approximately one when the update energy dominates ϵ. A larger rank therefore need not provide a more selective geometric preference. The rank sweep therefore measures the combined effect of the public representation and the alignment reward while the contribution strategies are reoptimized for each rank.