CS8695 - Research in Computer Science
Research & Academic Writing

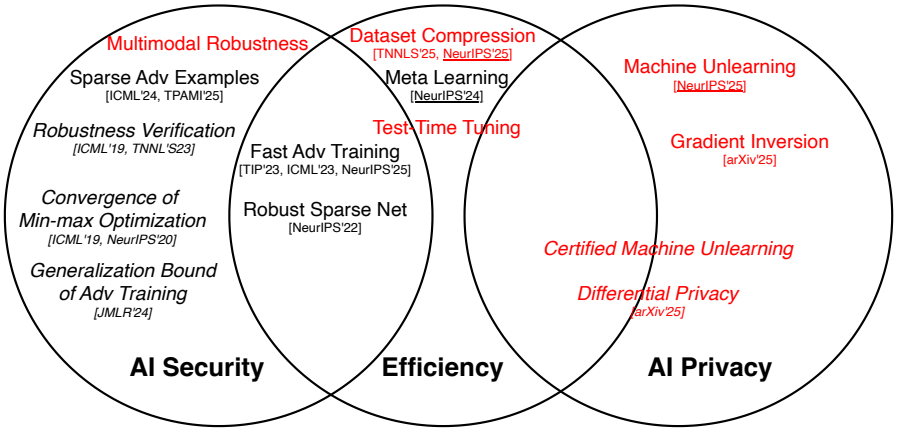
Chen Liu

Department of Computer Science
City University of Hong Kong

14, November, 2025

Outline

Overview of Research



Preliminary: Machine Unlearning

Machine unlearning (MU) targets the need to remove specific data influences from pretrained models, while complying with privacy requirements.

Preliminary: Machine Unlearning

Machine unlearning (MU) targets the need to remove specific data influences from pretrained models, while complying with privacy requirements.

Examples:

- Some training data is updated or no longer correct.
- The copyright of some training data expired.
- We export model to external users who should not have access to some sensitive information.

Preliminary: Exact and Approximate Machine Unlearning

Exact machine unlearning

- Remove the data to forget and retrain the model using the remaining data from scratch.
- Golden standard but expensive. Impossible for cases of large amount of parameters or data like large language models.

Preliminary: Exact and Approximate Machine Unlearning

Exact machine unlearning

- Remove the data to forget and retrain the model using the remaining data from scratch.
- Golden standard but expensive. Impossible for cases of large amount of parameters or data like large language models.

Approximate machine unlearning

- Finetune the pretrained models to remove the effect of data to forget while maintaining the performance of the remaining data.
- Inaccurate but efficient. Suffer from issues like unstable performance and catastrophic forgetting.

Preliminary: Exact and Approximate Machine Unlearning

Exact machine unlearning

- Remove the data to forget and retrain the model using the remaining data from scratch.
- Golden standard but expensive. Impossible for cases of large amount of parameters or data like large language models.

Approximate machine unlearning

- Finetune the pretrained models to remove the effect of data to forget while maintaining the performance of the remaining data.
- Inaccurate but efficient. Suffer from issues like unstable performance and catastrophic forgetting.

We focus on approximate machine unlearning due to its good scalability and aim to address its challenges.

Preliminary: Terminologies

- Forget set $\mathcal{D}_f$: the set of data to forget.
- Retain set $\mathcal{D}_r$: the remaining data to remember.
- Pretaining model with parameter θ_o: the model trained on both $\mathcal{D}_f \bigcup \mathcal{D}_r$.
- Retrained model with parameter θ_u: the model trained only on $\mathcal{D}_r$.

Preliminary: Terminologies

- Forget set $\mathcal{D}_f$: the set of data to forget.
- Retain set $\mathcal{D}_r$: the remaining data to remember.
- Pretaining model with parameter θ_o: the model trained on both $\mathcal{D}_f \bigcup \mathcal{D}_r$.
- Retrained model with parameter θ_u: the model trained only on $\mathcal{D}_r$.

In approximate machine unlearning, we aim to design an algorithm $\mathcal{A}$ such that $\mathcal{A}(\theta_o, \mathcal{D}_f, \mathcal{D}_r) \simeq \theta_u$.

Preliminary: Terminologies

- Forget set $\mathcal{D}_f$: the set of data to forget.
- Retain set $\mathcal{D}_r$: the remaining data to remember.
- Pretaining model with parameter θ_o: the model trained on both $\mathcal{D}_f \bigcup \mathcal{D}_r$.
- Retrained model with parameter θ_u: the model trained only on $\mathcal{D}_r$.

In approximate machine unlearning, we aim to design an algorithm $\mathcal{A}$ such that $\mathcal{A}(\theta_o, \mathcal{D}_f, \mathcal{D}_r) \simeq \theta_u$.

- $\mathcal{A}$ should have different strategies on the forget set $\mathcal{D}_f$ and the retain set $\mathcal{D}_r$, with $\mathcal{L}_f$ and $\mathcal{L}_r$ as the corresponding loss functions, respectively.

$$\min_{\theta} \mathcal{L}_f(\theta) + \mathcal{L}_r(\theta)$$

Preliminary: Terminologies

- Forget set $\mathcal{D}_f$: the set of data to forget.
- Retain set $\mathcal{D}_r$: the remaining data to remember.
- Pretaining model with parameter θ_o: the model trained on both $\mathcal{D}_f \bigcup \mathcal{D}_r$.
- Retrained model with parameter θ_u: the model trained only on $\mathcal{D}_r$.

In approximate machine unlearning, we aim to design an algorithm $\mathcal{A}$ such that $\mathcal{A}(\theta_o, \mathcal{D}_f, \mathcal{D}_r) \simeq \theta_u$.

- $\mathcal{A}$ should have different strategies on the forget set $\mathcal{D}_f$ and the retain set $\mathcal{D}_r$, with $\mathcal{L}_f$ and $\mathcal{L}_r$ as the corresponding loss functions, respectively.

$$\min_{\theta} \mathcal{L}_f(\theta) + \mathcal{L}_r(\theta)$$

- $\mathcal{L}_f$ and $\mathcal{L}_r$ are usually opposite functions.

Preliminary: Evaluation Criteria

- The accuracy on the retrain set (RA).
- The accuracy on the forget set (FA).
- The accuracy on the test set (TA).
- The accuracy on the membership inference attack on the forget set (MIA).

Ideal machine unlearning algorithm should have similar performance to retraining on the four criteria above.

Challenges for Current MU Methods

Let's review the machine unlearning problem below.

$$\min_{\theta} \mathcal{L}_f(\theta) + \mathcal{L}_r(\theta)$$

Existing methods may (1) jointly minimize $\mathcal{L}_f$ and $\mathcal{L}_r$; (2) alternatively minimize $\mathcal{L}_f$ and $\mathcal{L}_r$. However, they suffer from either *suboptimal performance* or *prohibitively large performance variance*.

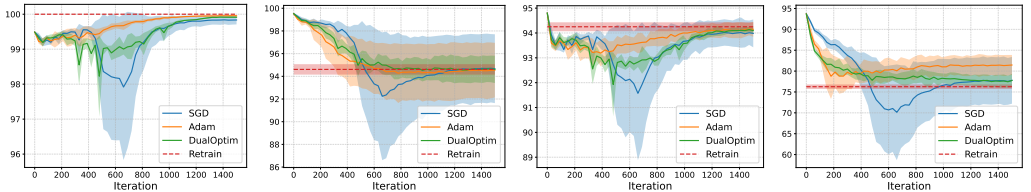


Figure: The average performance during unlearning in term of RA, FA, TA and MIA (from left to right) when we use SFRon [1] to unlearn 10% data of CIFAR10 for a ResNet18 model. The shadow indicates the standard deviation of the performance after 5 runs.

[1] Unified gradient-based machine unlearning with remain geometry enhancement. NeurIPS 2024.

Recipe 1: Adaptive Learning Rate

- Observation 1: the gradient magnitudes vary a lot during unlearning.
- Observation 2: there is a big discrepancy between the gradients on $\mathcal{L}_f$ and the ones on $\mathcal{L}_r$.

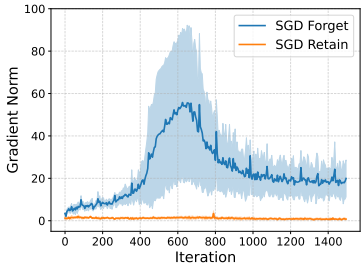


Figure: The gradient norms on $\mathcal{L}_f$ and $\mathcal{L}_r$, respectively. Left: SGD; .

Recipe 1: Adaptive Learning Rate

- Observation 1: the gradient magnitudes vary a lot during unlearning.
- Observation 2: there is a big discrepancy between the gradients on $\mathcal{L}_f$ and the ones on $\mathcal{L}_r$.

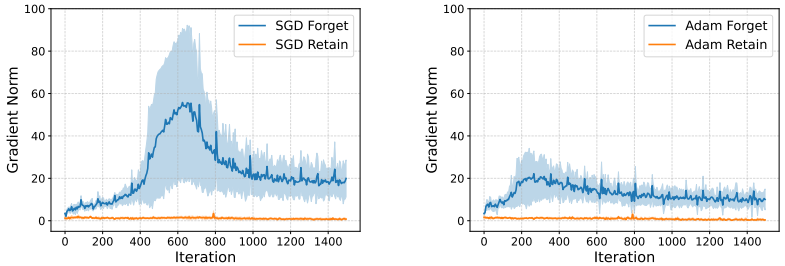


Figure: The gradient norms on $\mathcal{L}_f$ and $\mathcal{L}_r$, respectively. Left: SGD; Right: Adam.

Both observations indicate challenges when using a *unified learning rate*, which is the case of optimizers like SGD. We need to *adaptively* adjust the learning rate.

Recipe 2: Decoupled Statistics in Optimizers

- Observation: there is a big discrepancy between the gradients on $\mathcal{L}_f$ and the ones on $\mathcal{L}_r$.

Recipe 2: Decoupled Statistics in Optimizers

- Observation: there is a big discrepancy between the gradients on $\mathcal{L}_f$ and the ones on $\mathcal{L}_r$.

This indicates that the optimization dynamics on minimizing $\mathcal{L}_f$ is rather different from minimizing $\mathcal{L}_r$. Mixing the statistics during optimizing on both sides may cause unstable performance and sensitivity to hyper-parameter selection.

Recipe 2: Decoupled Statistics in Optimizers

- Observation: there is a big discrepancy between the gradients on $\mathcal{L}_f$ and the ones on $\mathcal{L}_r$.

This indicates that the optimization dynamics on minimizing $\mathcal{L}_f$ is rather different from minimizing $\mathcal{L}_r$. Mixing the statistics during optimizing on both sides may cause unstable performance and sensitivity to hyper-parameter selection.

Therefore, we use different factors to denote the optimization statistics, such as momentum factors, for $\mathcal{L}_f$ and $\mathcal{L}_r$.

Recipe 2: Decoupled Statistics in Optimizers

If we use $\widehat{\boldsymbol{g}}_{f,t}$ and $\widehat{\boldsymbol{g}}_{r,t}$ to represent the stochastic gradient from $\mathcal{L}_f$ and $\mathcal{L}_r$ at the time stamp t, respectively.

$$
\begin{aligned}
&\text{(Shared Momentum)} \begin{cases} \boldsymbol{m}_{f,t}^{S} &= \alpha \boldsymbol{m}_{r,t-1}^{S} + \widehat{\boldsymbol{g}}_{f,t}^{S}, \quad \theta_{f,t}^{S} = \theta_{r,t-1}^{S} - \eta \boldsymbol{m}_{f,t}^{S} \\ \boldsymbol{m}_{r,t}^{S} &= \alpha \boldsymbol{m}_{f,t}^{S} + \widehat{\boldsymbol{g}}_{r,t}^{S}, \quad \theta_{r,t}^{S} = \theta_{f,t}^{S} - \eta \boldsymbol{m}_{r,t}^{S} \end{cases} \\
&\text{(Decoupled Momentum)} \begin{cases} \boldsymbol{m}_{f,t}^{D} &= \alpha \boldsymbol{m}_{f,t-1}^{D} + \widehat{\boldsymbol{g}}_{f,t}^{D}, \quad \theta_{f,t}^{D} = \theta_{r,t-1}^{D} - \eta \boldsymbol{m}_{f,t}^{D} \\ \boldsymbol{m}_{r,t}^{D} &= \alpha \boldsymbol{m}_{r,t-1}^{D} + \widehat{\boldsymbol{g}}_{r,t}^{D}, \quad \theta_{r,t}^{D} = \theta_{f,t}^{D} - \eta \boldsymbol{m}_{r,t}^{D} \end{cases}
\end{aligned} \tag{1}
$$

By induction, the variance of the model parameters by decoupled momentum is *theoretically guaranteed smaller* compared with shared momentum after the same number of iterations.

Theoretical Guarantees

Assumptions:

(Stochastic Gradient Condition) For all time steps $t = 0, \ldots, T-1$, the stochastic gradients of the forget loss $\widehat{\boldsymbol{g}}_{f,t}$ and retain loss $\widehat{\boldsymbol{g}}_{r,t}$ satisfy:

$$\widehat{\boldsymbol{g}}_{f,t} = \boldsymbol{g}_{f,t} + \boldsymbol{\epsilon}_{f,t}, \quad \widehat{\boldsymbol{g}}_{r,t} = \boldsymbol{g}_{r,t} + \boldsymbol{\epsilon}_{r,t},$$

where $\boldsymbol{g}_{f,t} \coloneqq \nabla_{\theta_t} \mathcal{L}_f(\mathcal{D}_f, \theta_t)$ and $\boldsymbol{g}_{r,t} \coloneqq \nabla_{\theta_t} \mathcal{L}_r(\mathcal{D}_r, \theta_t)$ are the full-batch gradients with model parameter θ_t at the time stamp t. $\boldsymbol{\epsilon}_{f,t}$ and $\boldsymbol{\epsilon}_{r,t}$ are batch noises with zero mean and a bounded variance: there exists a minimal $\sigma^2 \geq 0$ such that $\mathrm{Var}(\boldsymbol{\epsilon}_{f,t}) \leq \sigma^2$, $\mathrm{Var}(\boldsymbol{\epsilon}_{r,t}) \leq \sigma^2$ for all t.

(Correlation Bounds) The correlation between the stochastic gradients from the same function in different time steps is bounded while the correlation between stochastic gradients from different functions can be ignored. That is to say, $\exists \tau \in [0, 1]$ such that:

$$\forall t_1 \neq t_2, , s.t.\ \rho(\widehat{\boldsymbol{g}}_{f,t_1}, \widehat{\boldsymbol{g}}_{f,t_2}) \leq \tau, \rho(\widehat{\boldsymbol{g}}_{r,t_1}, \widehat{\boldsymbol{g}}_{r,t_2}) \leq \tau, \quad \forall t_1, t_2, \rho(\widehat{\boldsymbol{g}}_{f,t_1}, \widehat{\boldsymbol{g}}_{r,t_2}) \leq o(\tau) \simeq 0$$

(Lipschitz Smoothness) The loss functions $\mathcal{L}_f$ and $\mathcal{L}_r$ are both L-smooth:

$$\forall \theta_1, \theta_2, \|\nabla_{\theta_1} \mathcal{L}_f(\mathcal{D}_f, \theta_1) - \nabla_{\theta_2} \mathcal{L}_f(\mathcal{D}_f, \theta_2)\| \leq L\|\theta_1 - \theta_2\|, \tag{2}$$

$$\forall \theta_1, \theta_2, \|\nabla_{\theta_1} \mathcal{L}_r(\mathcal{D}_r, \theta_1) - \nabla_{\theta_2} \mathcal{L}_r(\mathcal{D}_r, \theta_2)\| \leq L\|\theta_1 - \theta_2\|. \tag{3}$$

Theoretical Guarantees

$$\text{(Shared Momentum)} \begin{cases} \boldsymbol{m}^S_{f,t} &= \alpha \boldsymbol{m}^S_{r,t-1} + \widehat{\boldsymbol{g}}^S_{f,t}, \quad \theta^S_{f,t} = \theta^S_{r,t-1} - \eta \boldsymbol{m}^S_{f,t} \\ \boldsymbol{m}^S_{r,t} &= \alpha \boldsymbol{m}^S_{f,t} + \widehat{\boldsymbol{g}}^S_{r,t}, \quad \theta^S_{r,t} = \theta^S_{f,t} - \eta \boldsymbol{m}^S_{r,t} \end{cases}$$

$$\text{(Decoupled Momentum)} \begin{cases} \boldsymbol{m}^D_{f,t} &= \alpha \boldsymbol{m}^D_{f,t-1} + \widehat{\boldsymbol{g}}^D_{f,t}, \quad \theta^D_{f,t} = \theta^D_{r,t-1} - \eta \boldsymbol{m}^D_{f,t} \\ \boldsymbol{m}^D_{r,t} &= \alpha \boldsymbol{m}^D_{r,t-1} + \widehat{\boldsymbol{g}}^D_{r,t}, \quad \theta^D_{r,t} = \theta^D_{f,t} - \eta \boldsymbol{m}^D_{r,t} \end{cases}$$

Lemma

(Variance of Gradients) If the loss function $\mathcal{L}$ is Lipschitz smooth with a constant L, and $\mathrm{Var}(\theta) \leq \sigma_\theta^2$, then we have $\mathrm{Var}(\nabla_\theta \mathcal{L}(\theta)) \leq L^2 \sigma_\theta^2$.

Theorem

(Variance Bound Comparison for Decoupled vs. Shared Momentum) For the shared and decoupled schemes using the same hyperparameters (η, α), and we use $\overline{\mathrm{Var}}(\cdot)$ to denote the maximum variance of a variable, if the function $\mathcal{L}_f$, $\mathcal{L}_r$ and the stochastic gradient $\{(\widehat{\boldsymbol{g}}^S_{f,i}, \widehat{\boldsymbol{g}}^S_{r,i})\}_{i=0}^{T-1}$, $\{(\widehat{\boldsymbol{g}}^D_{f,i}, \widehat{\boldsymbol{g}}^D_{r,i})\}_{i=0}^{T-1}$ satisfy the assumptions, then

$$\forall t, \overline{\mathrm{Var}}(\theta^D_{f,t}) \leq \overline{\mathrm{Var}}(\theta^S_{f,t}), \quad \overline{\mathrm{Var}}(\theta^D_{r,t}) \leq \overline{\mathrm{Var}}(\theta^S_{r,t}),$$

DualOptim

Algorithm 1: Machine Unlearning with Shared Optimizer / Dual Optimizers

Input: Model: f_θ; Forget set: $\mathcal{D}_f$; Retain set: $\mathcal{D}_r$; Iterations for outer loop: T_o; Iterations for forgetting: T_f; Iterations for retaining: T_r; Step sizes: η, η_f, η_r.

Optim is the same optimizer as in pretraining with step size η.

Optim_f is $\text{Adam}(\theta, \eta_f)$, Optim_r is the same optimizer as in pretraining with step size η_r.

for $t = 1, ..., T_o$ **do**
  for $t' = 1, ..., T_f$ **do**
    Fetch mini-batch data from the forget set $B_f \sim \mathcal{D}_f$
    Calculate the forget loss $\mathcal{L}_f$ on B_f and get the gradient
    Use Optim / Optim_f to update θ
  end for
  for $t' = 1, ..., T_r$ **do**
    Fetch mini-batch data from the retain set $B_r \sim \mathcal{D}_r$
    Calculate the retain loss $\mathcal{L}_r$ on B_r and get the gradient
    Use Optim / Optim_r to update θ
  end for
end for
Output: Model f_θ

Experiments: Image Classification

Table 1: Performance summary of MU methods for image classification. Experiments are conducted on (a) 10% random subset of **CIFAR-10** using **ResNet-18** and **(b)** 10% random subset of **TinyImageNet** using **Swin-T**. All results are presented as mean and standard deviation across 5 trials with different random forget data. Performance gaps with *RT* are indicated in blue. The average gap (**Gap**) and average standard deviation (**Std**) metrics are calculated by the average of the gaps and standard deviation measured in FA, RA, TA, and MIA, respectively. All the numbers are in percentage.

(a) **CIFAR-10 Random Subset Unlearning (10%)**

Method	FA	RA	TA	MIA	Gap ↓	Std ↓
RT	$94.61_{\pm 0.46}$ (0.00)	$100.00_{\pm 0.00}$ (0.00)	$94.25_{\pm 0.18}$ (0.00)	$76.26_{\pm 0.54}$ (0.00)	0.00	0.30
FT	$99.16_{\pm 0.10}$ (4.55)	$99.84_{\pm 0.06}$ (0.16)	$94.10_{\pm 0.09}$ (0.15)	$88.77_{\pm 0.38}$ (12.51)	4.34	0.16
GA	$98.76_{\pm 0.39}$ (4.15)	$99.10_{\pm 0.90}$ (0.90)	$93.89_{\pm 0.41}$ (0.36)	$92.58_{\pm 0.55}$ (16.32)	5.43	0.44
RL	$97.19_{\pm 0.21}$ (2.58)	$99.67_{\pm 0.08}$ (0.33)	$94.03_{\pm 0.27}$ (0.22)	$68.19_{\pm 0.95}$ (8.43)	2.80	0.38
SCRUB	$92.88_{\pm 0.25}$ (1.73)	$99.62_{\pm 0.10}$ (0.38)	$93.54_{\pm 0.22}$ (0.71)	$82.78_{\pm 0.86}$ (6.52)	2.33	**0.36**
+DualOptim	$94.90_{\pm 0.42}$ (0.29)	$99.52_{\pm 0.09}$ (0.48)	$93.50_{\pm 0.20}$ (0.75)	$78.26_{\pm 0.79}$ (2.00)	**0.88**	0.38
SalUn	$96.99_{\pm 0.31}$ (2.38)	$99.40_{\pm 0.28}$ (0.60)	$93.84_{\pm 0.36}$ (0.41)	$65.76_{\pm 1.05}$ (10.50)	3.47	0.50
+DualOptim	$95.47_{\pm 0.22}$ (0.86)	$99.06_{\pm 0.94}$ (0.60)	$92.47_{\pm 0.29}$ (1.78)	$76.14_{\pm 0.70}$ (0.12)	**0.93**	**0.35**
SFRon	$94.67_{\pm 3.03}$ (0.06)	$99.83_{\pm 0.13}$ (0.17)	$93.98_{\pm 0.56}$ (0.27)	$77.80_{\pm 5.61}$ (1.54)	0.51	2.33
+DualOptim	$94.69_{\pm 1.13}$ (0.02)	$99.92_{\pm 0.01}$ (0.08)	$94.11_{\pm 0.11}$ (0.14)	$77.77_{\pm 1.39}$ (1.51)	**0.44**	**0.66**

(b) **TinyImageNet Random Subset Unlearning (10%)**

Method	FA	RA	TA	MIA	Gap ↓	Std ↓
RT	$85.29_{\pm 0.09}$ (0.00)	$99.55_{\pm 0.03}$ (0.00)	$85.49_{\pm 0.15}$ (0.00)	$69.30_{\pm 0.20}$ (0.00)	0.00	0.12
FT	$96.50_{\pm 0.10}$ (11.21)	$98.23_{\pm 0.08}$ (1.32)	$82.67_{\pm 0.21}$ (2.82)	$79.85_{\pm 0.13}$ (10.55)	6.48	0.13
GA	$90.02_{\pm 3.26}$ (4.73)	$90.84_{\pm 3.29}$ (8.71)	$75.64_{\pm 2.67}$ (9.85)	$78.97_{\pm 2.07}$ (9.67)	8.24	2.82
RL	$94.66_{\pm 0.26}$ (9.37)	$98.02_{\pm 0.14}$ (1.53)	$82.73_{\pm 0.27}$ (2.76)	$54.45_{\pm 1.04}$ (15.15)	7.13	0.43
SCRUB	$97.80_{\pm 0.16}$ (12.51)	$98.13_{\pm 0.08}$ (1.42)	$82.64_{\pm 0.19}$ (2.85)	$79.62_{\pm 0.41}$ (10.32)	6.78	**0.21**
+DualOptim	$97.20_{\pm 0.20}$ (11.91)	$98.30_{\pm 0.10}$ (1.25)	$83.17_{\pm 0.19}$ (2.32)	$79.10_{\pm 0.63}$ (9.80)	**6.32**	0.28
SalUn	$97.69_{\pm 0.14}$ (12.40)	$98.89_{\pm 0.03}$ (0.66)	$84.02_{\pm 0.32}$ (1.47)	$61.87_{\pm 0.97}$ (7.43)	5.49	0.37
+DualOptim	$91.68_{\pm 0.28}$ (6.39)	$95.13_{\pm 0.18}$ (4.42)	$80.16_{\pm 0.34}$ (5.33)	$72.48_{\pm 0.33}$ (3.18)	**4.83**	**0.28**
SFRon	$96.41_{\pm 0.74}$ (11.12)	$98.95_{\pm 0.22}$ (0.60)	$83.40_{\pm 0.51}$ (2.09)	$70.40_{\pm 3.15}$ (1.10)	3.73	1.16
+DualOptim	$92.26_{\pm 1.44}$ (6.97)	$98.27_{\pm 0.12}$ (1.28)	$83.12_{\pm 0.21}$ (2.37)	$69.19_{\pm 2.27}$ (0.11)	**2.68**	**1.01**

Experiments: Image Generation

Table 2: Class-wise unlearning performance on **CIFAR-10** with **DDPM** and **ImageNet** with **DiT**. The best unlearning performance for each forgetting class is highlighted in **bold** for FA (in %) and FID. Note that the results of SA, SalUn and SFRon are those reported in [7].

Method	**CIFAR-10 Class-wise Unlearning**										**ImageNet Class-wise Unlearning**									
	Automobile		Cat		Dog		Horse		Truck		Cockatoo		Golden Retriever		White Wolf		Arctic Fox		Otter	
	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓	FA ↓	FID ↓
SA	**0.00**	23.56	14.20	21.34	8.60	21.19	**0.00**	21.13	**0.00**	29.04	**0.00**	348.75	**0.00**	298.97	**0.00**	45.89	**0.00**	393.91	29.8	321.21
SalUn	0.20	21.23	1.40	20.29	**0.00**	20.18	0.60	20.70	0.80	20.45	91.21	18.47	46.09	25.28	**0.00**	15.16	45.90	408.07	87.5	19.69
SFRon	**0.00**	20.70	7.40	**18.44**	0.20	18.89	**0.00**	19.93	**0.00**	20.61	**0.00**	**13.59**	**0.00**	17.76	**0.00**	23.28	**0.00**	16.12	**0.00**	16.43
+DO	0.20	**19.72**	**1.00**	19.36	**0.00**	**18.58**	**0.00**	**18.91**	**0.00**	**17.26**	**0.00**	17.46	**0.00**	**14.63**	**0.00**	**14.72**	**0.00**	**14.91**	**0.00**	**14.55**

Experiments: Large Language Models

Table 4: Performance comparison of different MU methods on TOFU-finetuned **Phi-1.5** and **LLaMA 2**. The results include Model Capability (MC), Forget Efficacy (FE), and the average metric (Avg.) for forget 1%, 5% data, and 10% data.

Method	Phi-1.5									LLaMA 2								
	forget 1% data			forget 5% data			forget 10% data			forget 1% data			forget 5% data			forget 10% data		
	MC ↑	FE ↑	Avg. ↑	MC ↑	FE ↑	Avg. ↑	MC↑	FE ↑	Avg. ↑	MC ↑	FE ↑	Avg. ↑	MC ↑	FE ↑	Avg. ↑	MC ↑	FE ↑	Avg. ↑
GA+GD	0.4934	0.4493	0.4714	0.4360	0.5084	0.4722	0.4471	0.5246	0.4859	0.6696	0.5908	0.6302	0.0000	0.8772	0.4386	0.5592	0.9346	0.7469
ME+GD	**0.4944**	0.3938	0.4441	0.4559	0.4480	0.4520	0.4594	0.4564	0.4579	0.7271	0.9204	0.8237	**0.7472**	0.9313	0.8392	**0.7357**	0.9489	0.8423
+DO	0.4866	**0.6913**	**0.5889**	**0.4676**	**0.8200**	**0.6438**	**0.5009**	**0.7732**	**0.6370**	**0.7425**	**0.9612**	**0.8519**	0.7316	**0.9602**	**0.8459**	0.7315	**0.9625**	**0.8470**
DPO+GD	0.2410	0.6831	0.4621	0.4105	0.6334	0.5219	0.3517	0.6302	0.4910	0.7564	0.5335	0.6450	0.0000	0.8243	0.4122	0.0000	0.8041	0.4021
IDK+AP	**0.4403**	0.5723	0.5063	**0.4800**	0.5112	0.4956	**0.4614**	0.6003	0.5308	**0.7580**	0.7625	0.7603	**0.7529**	0.7479	0.7504	**0.7471**	0.7433	0.7452
+DO	0.4221	**0.7037**	**0.5629**	0.4633	**0.6974**	**0.5804**	0.4422	**0.7193**	**0.5807**	0.7412	**0.8075**	**0.7743**	0.7354	**0.7958**	**0.7656**	0.7362	**0.7855**	**0.7609**

Sparse Portfolio Optimization

Problem definition

- Select at most m assets from n candidates to maximize investment performance.

Sparse Portfolio Optimization

Problem definition

- Select at most m assets from n candidates to maximize investment performance.

Why sparsity matters?

- better interpretability of selected assets.
- lower transaction costs and easier implementation.

Sparse Portfolio Optimization

Problem definition

- Select at most m assets from n candidates to maximize investment performance.

Why sparsity matters?

- better interpretability of selected assets.
- lower transaction costs and easier implementation.

Evaluation metrics

- Cumulative Wealth (CW): total portfolio return over the horizon.
- Sharpe Ratio (SR): risk-adjusted return per unit volatility.
- Maximum Drawdown (MD): worst-case loss from peak to trough.

Alpha Factor

Definition

- An alpha factor is a mathematical expression that maps historical features (e.g. price, volume, volatility) to a score for each asset.
- Higher alpha factor score $\rightarrow$ More attractive asset under the investment objective.

Evaluation

- The correlation between the ranking by alpha factors and the real ranking.

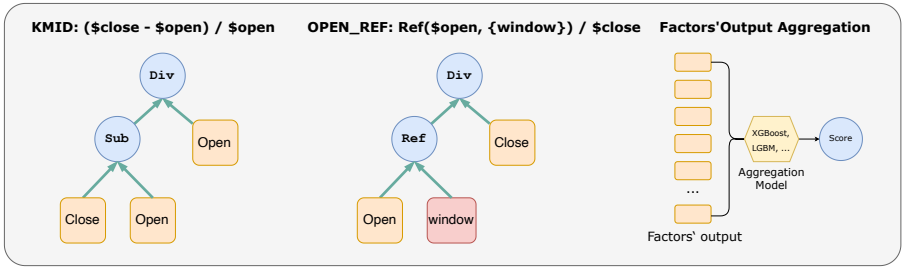


Figure: Example of alpha factors with their tree-structures in Alpha158 (Left) and how multiple factors' outputs are aggregated using models such as XGBoost or LightGBM to produce a final score (Right).

Alpha Factor

Traditional Alpha Factor Pool

Alpha#74: ((rank(correlation(close, sum(adv30, 37.4843), 15.1365)) < rank(correlation(rank(((high * 0.0261661) + (vwap * (1 - 0.0261661)))), rank(volume), 11.4791))) * -1)

Alpha#75: (rank(correlation(vwap, volume, 4.24304)) < rank(correlation(rank(low), rank(adv50), 12.4413)))

Alpha#76: (max(rank(decay_linear(delta(vwap, 1.24383), 11.8259)), Ts_Rank(decay_linear(Ts_Rank(correlation(IndNeutralize(low, IndClass.sector), adv81, 8.14941), 19.569), 17.1543), 19.383)) * -1)

Alpha#77: min(rank(decay_linear(((((high + low) / 2) + high) - (vwap + high)), 20.0451)), rank(decay_linear(correlation(((high + low) / 2), adv40, 3.1614), 5.64125)))

Alpha#78: (rank(correlation(sum(((low * 0.352233) + (vwap * (1 - 0.352233))), 19.7428), sum(adv40, 19.7428), 6.83313))^rank(correlation(rank(vwap), rank(volume), 5.77492)))

Alpha#79: (rank(delta(IndNeutralize(((close * 0.60733) + (open * (1 - 0.60733))), IndClass.sector), 1.23438)) < rank(correlation(Ts_Rank(vwap, 3.60973), Ts_Rank(adv150, 9.18637), 14.6644)))

Limitations of Current Methods

Factor-based Methods

- Interpretable but require heavy manual design.
- Degrade quickly in live trading.
- Sparse decay: performance drops sharply when selecting only few assets.

Traditional Optimization Methods

- Competitive but uninterpretable results.
- Computationally expensive and sensitive to hyper-parameters.

Limitations of Current Methods

Factor-based Methods

- Interpretable but require heavy manual design.
- Degrade quickly in live trading.
- Sparse decay: performance drops sharply when selecting only few assets.

Traditional Optimization Methods

- Competitive but uninterpretable results.
- Computationally expensive and sensitive to hyper-parameters.

Let's design an **adaptive, interpretable and robust** algorithm to find competitive alphas efficiently.

Overall Framework

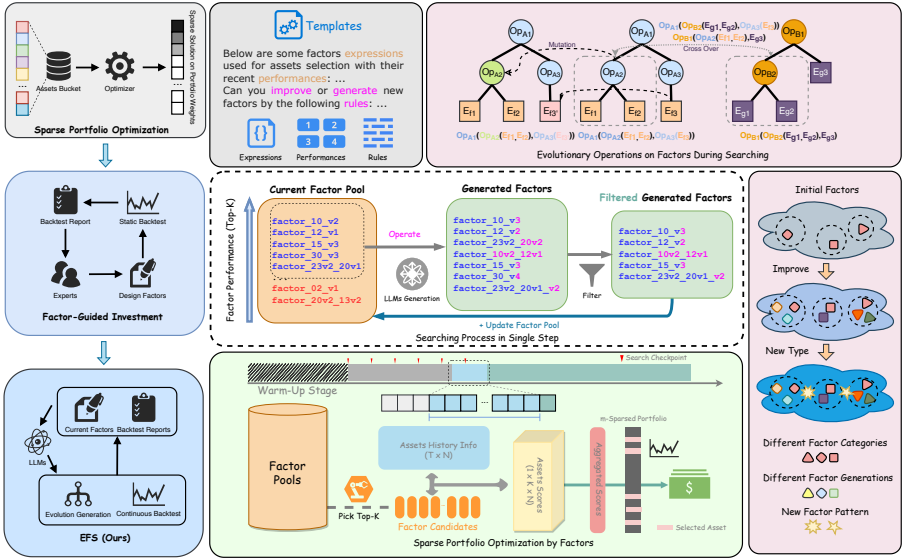


Figure: Evolutionary Factor Search (EFS) framework.

Prompt to Improve Alpha Factor

```
You are a world-class quantitative researcher and Python programmer specializing
in alpha factor design for asset ranking.
Your task is to generate high-quality Python factor functions that are evolved
versions of provided factors.
STRICT REQUIREMENTS:
1. Output ONLY a Python list of function strings - no comments or explanations
2. Each function MUST:
  - Be bug-free and executable
  - Maintain identical input signature: prices, window  - Use only numpy (as np),
don't depend on any external function or variable, you need to do computation all
inside function
  - Handle edge cases (short series, NaNs)
  - Clearly indicate if combining or modifying existing factors
3. Absolute prohibitions:
  - No external functions
  - No hardcoded values that should be parameters
  - No pandas or other libraries
  - No comments in output code
4. Factor name rules: [factor_name_part]_[window_size]_v[version number], the
window_size can only be the following value: 3, 7, 14, 21
5. Value of output factor: For factors, higher value means better asset, please
make sure the output value is positive related to performance of assets.
ACTION SPACE:
1. Improve existing factors by mutation:
  - Modifying parameters (e.g., inner parameters)
  - Adjusting logic
  - Updating inside operators for factors
2. Improve existing factors by crossover:
  - Combining two existing factors to create a new one if you think they can work
together
  - Restart version number from v1 for new factors
IMPROVEMENT CRITERIA:
1. Version increments must show clear:
  - if you improve from a given version, increase 1 to version number, the
```

Figure: Prompt fed to LLMs in the evolution of alpha factors.

Evolution of Alpha Factor

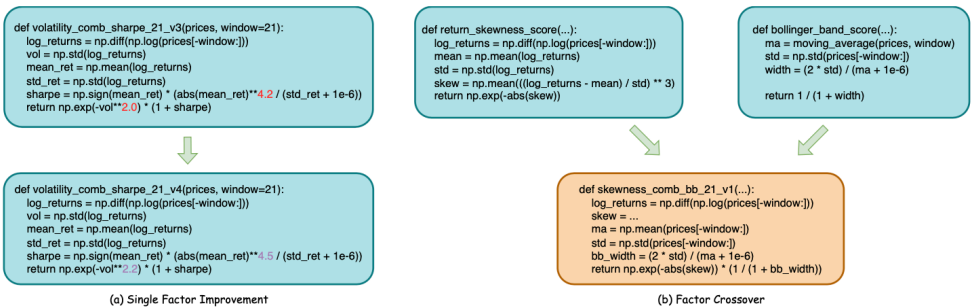


Figure: How alpha factor evolves in LLM-enabled search.

Interpret Alpha Factors

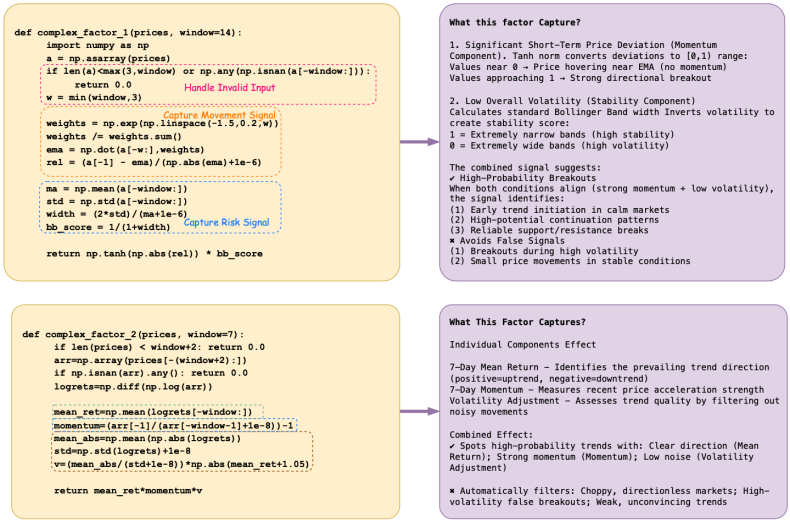


Figure: Use LLMs to interpret alpha factors.

Overall Framework

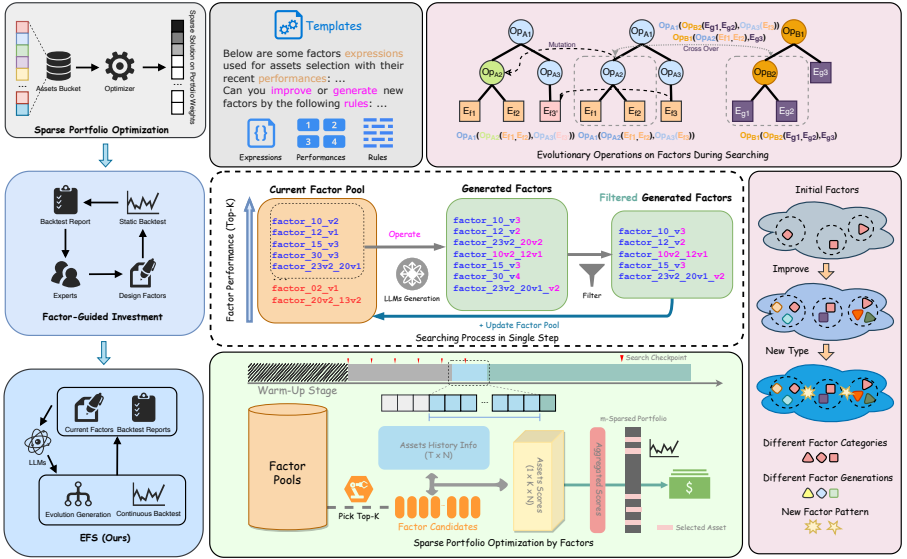


Figure: Evolutionary Factor Search (EFS) framework.

Evolutionary Factor Search (EFS)

Key Features of EFS:

- **LLM-driven factor generation:** prompt LLMs to create *executable* alpha factor formulas.
- **Evolutionary Search:** refine factors iteratively via *mutation* and *crossover*, guided by backtest results.
- **Closed-Loop Feedback:** use performance metrics (CW, SR) to update and prune the factor pool.
- **Sparse Portfolio Construction:** select top-m asset to construct portfolio.
- **Transparency:** factors are human-readable, interpretability and directly deployable.

Results on Real Market

Table: Evaluation of Cumulative Wealth (CW↑), Sharpe Ratio (SR↑), and Maximum Drawdown (MDD↓) on real-market datasets (US50, HSI45 and CSI300) for different model variants. The time frame is from 2019 to 2024 for US50, from 2022 to 2025 for HSI45 and CSI300.

		US50			HSI45			CSI300		
Group	Method	CW↑	SR↑	MDD↓	CW↑	SR↑	MDD↓	CW↑	SR↑	MDD↓
Baseline	1/N	4.562	0.072	0.344	1.333	0.029	0.409	1.087	0.014	0.214
	Min-cVaR	1.779	0.038	0.314	1.628	0.063	0.244	0.992	0.003	0.286
	Max-Sharpe	4.495	0.061	0.461	1.428	0.043	0.300	1.008	0.007	0.333
m=10	LGBM	4.182	0.063	0.332	1.611	0.038	0.367	2.334	0.072	0.225
	XGBoost	6.313	0.077	0.328	1.581	0.035	0.440	1.420	0.032	0.345
	mSSRM-PGA	5.121	0.059	0.569	0.766	-0.003	0.547	0.881	0.002	0.399
	ASMCVaR	10.259	0.073	0.582	2.481	0.052	0.453	1.453	0.030	0.462
	EFS-DeepSeek	**25.101**	0.132	0.288	**3.463**	0.080	0.385	3.437	0.079	0.327
	EFS-GPT	22.905	0.130	0.260	2.789	0.067	0.292	**4.962**	0.098	0.301
m=15	LGBM	3.899	0.062	0.328	1.588	0.037	0.387	1.812	0.055	0.250
	XGBoost	5.607	0.076	0.319	1.586	0.036	0.420	1.348	0.029	0.344
	mSSRM-PGA	4.976	0.062	0.477	0.766	-0.003	0.547	0.787	-0.010	0.384
	ASMCVaR	11.124	0.074	0.566	**2.647**	0.054	0.434	1.658	0.035	0.424
	EFS-DeepSeek	13.978	0.114	0.298	2.364	0.061	0.406	2.510	0.067	0.298
	EFS-GPT	**14.707**	0.117	0.278	2.277	0.058	0.307	**3.218**	0.082	0.246

Portfolio Performance

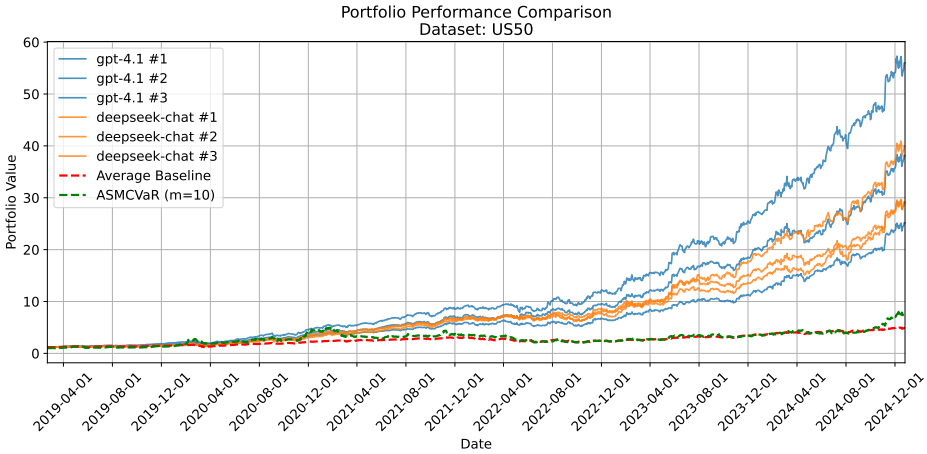


Figure: Portfolio performance comparison across US50, HSI45, and CSI300 datasets. Each plot shows the evolution of LLM-generated portfolios versus baselines and the ASMCVaR benchmark over time.

Portfolio Performance

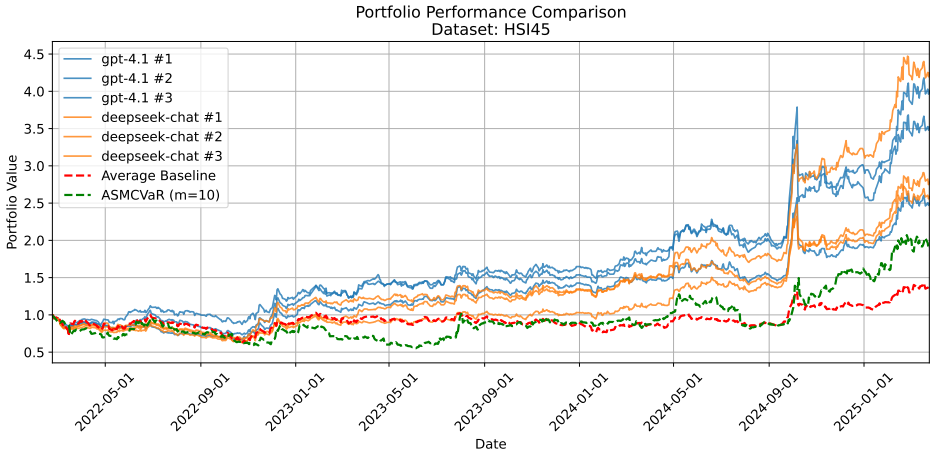


Figure: Portfolio performance comparison across US50, HSI45, and CSI300 datasets. Each plot shows the evolution of LLM-generated portfolios versus baselines and the ASMCVaR benchmark over time.

Portfolio Performance

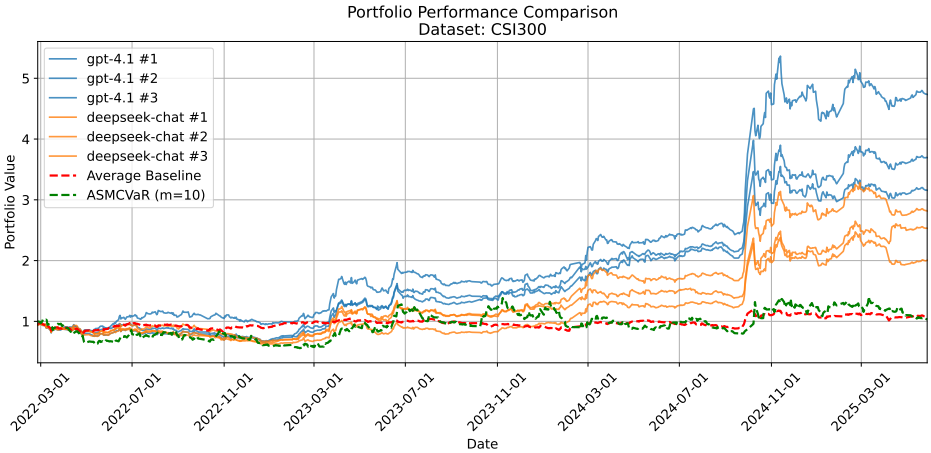
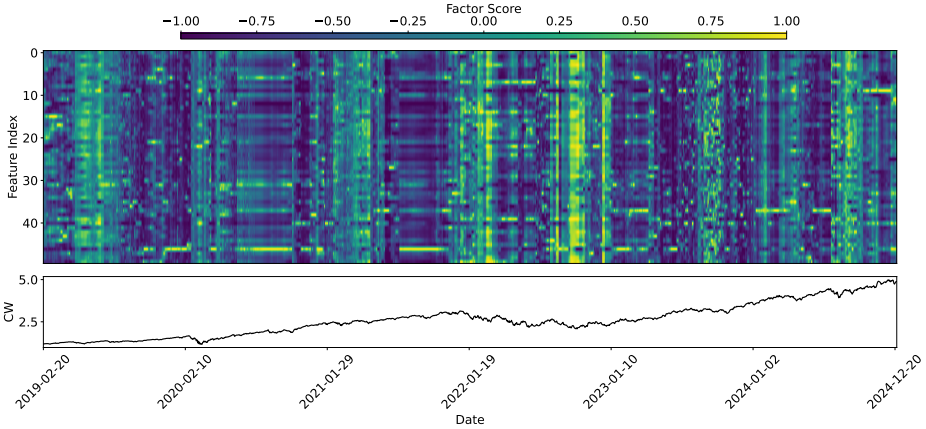


Figure: Portfolio performance comparison across US50, HSI45, and CSI300 datasets. Each plot shows the evolution of LLM-generated portfolios versus baselines and the ASMCVaR benchmark over time.

Portfolio Analysis

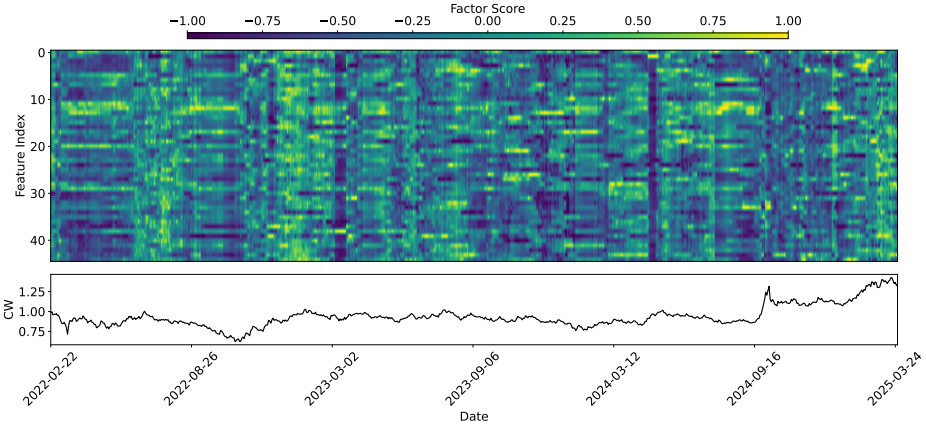


(a) US50 dataset

Figure: Factor score heatmaps and corresponding baseline curves across three datasets.

In general, portfolio holdings are more consistent in bull market.

Portfolio Analysis

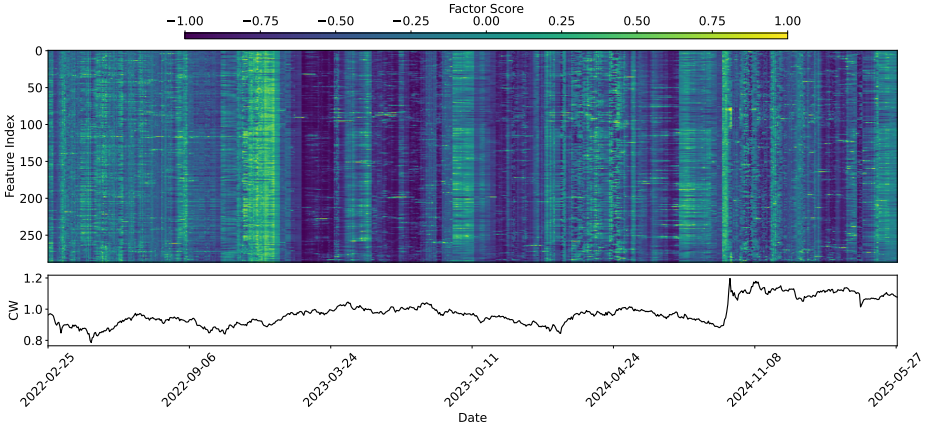


(a) HSI45 dataset

Figure: Factor score heatmaps and corresponding baseline curves across three datasets.

In general, portfolio holdings are more consistent in bull market.

Portfolio Analysis



(a) CSI300 dataset

Figure: Factor score heatmaps and corresponding baseline curves across three datasets.

In general, portfolio holdings are more consistent in bull market.

Transaction Fees

Table: Backtest results under transaction costs $c = 0.1\%$ and $c = 0.2\%$. Metrics shown are Cumulative Wealth (CW), Sharpe Ratio (SR), and Maximum Drawdown (MDD) across datasets US50, HSI45, and CSI300.

c	Method	US50			HSI45			CSI300		
		CW	SR	MDD	CW	SR	MDD	CW	SR	MDD
-	EFS-GPT 4.1	39.746±15.484	0.154±0.013	0.252±0.021	3.338±0.770	0.076±0.012	0.324±0.041	3.862±0.802	0.086±0.011	0.290±0.079
	EFS-DeepSeek	32.709±6.244	0.149±0.003	0.261±0.014	3.203±0.906	0.076±0.015	0.385±0.024	2.451±0.412	0.060±0.010	0.356±0.022
0.1%	EFS-GPT 4.1	25.293±10.802	0.135±0.016	0.256±0.021	2.710±0.571	0.065±0.011	0.340±0.049	2.624±0.425	0.064±0.008	0.354±0.093
	EFS-DeepSeek	22.058±4.053	0.133±0.003	0.265±0.015	2.643±0.775	0.065±0.015	0.410±0.023	1.668±0.227	0.039±0.008	0.434±0.012
0.2%	EFS-GPT 4.1	16.114±7.464	0.117±0.018	0.260±0.022	2.201±0.422	0.054±0.011	0.358±0.056	1.785±0.208	0.043±0.006	0.412±0.103
	EFS-DeepSeek	14.876±2.646	0.117±0.003	0.270±0.016	2.182±0.665	0.054±0.016	0.433±0.022	1.136±0.130	0.018±0.007	0.502±0.009

Different LLMs

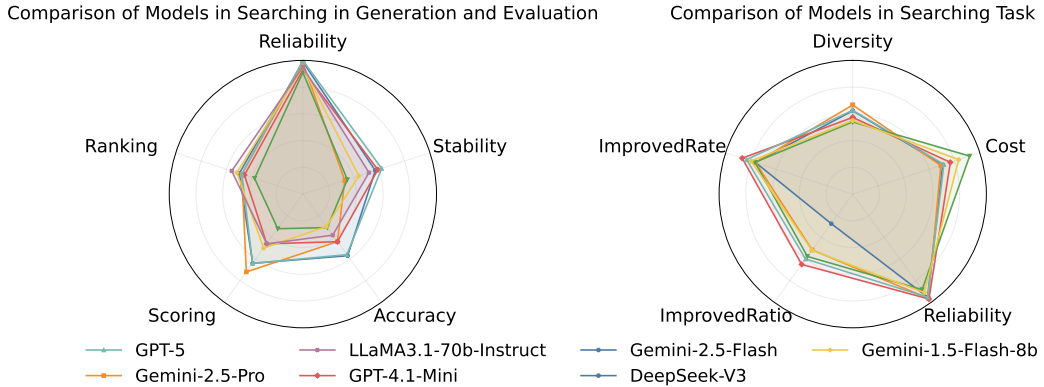


Figure: Radar chart comparison of model performance in alpha factor searching. (Left) Models evaluated on generation and evaluation tasks across reliability, stability, accuracy, scoring, and ranking dimensions. (Right) Models compared on search efficiency metrics including diversity, cost, reliability, improvement rate, and improvement ratio.

Summary

Contributions:

- We propose a unified evolutionary factor search (EFS) framework.
- We adaptively adjust alpha factors in different market regimes.
- The alpha factors found demonstrate good interpretability and competitive performance.

Current Limitations:

- Backtesting is computationally expensive.
- Sensitivity to prompt design and hyper-parameters.
- Limited coverage of multimodal signals (e.g. news, macro events).

Future Works

- Multimodal factor mining using news and alternative textual data.
- Enhance prompt robustness with automatic filtering and tuning.

Outline

Why Academic Writing Matters

- A good research should be presented in a readable and coherent way.
- An academic manuscript is also a good way to systematically organize and summarize your current work.
  - To figure out how current results are organized in a logical way.
  - To figure our what you should do next.

Why Academic Writing Matters

- A good research should be presented in a readable and coherent way.
- An academic manuscript is also a good way to systematically organize and summarize your current work.
  - To figure out how current results are organized in a logical way.
  - To figure our what you should do next.

- **Do not start writing your paper in the last few weeks or even days.**
- Ask a peer to proof read your paper before submission.

Common Structure

1. Introduction: background, motivation, method overview and major contributions.
2. Related works: literature review.
3. Preliminary: background concepts and motivation.
4. Methodology: details about your proposed methods and theoretical results.
5. Experiments: details about the experimental results.
6. Conclusion: concluding remarks.
7. Appendix: proofs, pseudo codes, additional experiment settings and results.

Put the right content in the right position.

Abstract

- Abstract is to facilitate the reader to quickly capture the content of your work and is usually used for paper bidding.
- Abstract should be more concise than the introduction and should not be too long, especially for a conference paper.

Abstract

- Abstract is to facilitate the reader to quickly capture the content of your work and is usually used for paper bidding.
- Abstract should be more concise than the introduction and should not be too long, especially for a conference paper.
- In most cases, background and motivation are not included in the abstract.

Abstract

- Abstract is to facilitate the reader to quickly capture the content of your work and is usually used for paper bidding.
- Abstract should be more concise than the introduction and should not be too long, especially for a conference paper.
- In most cases, background and motivation are not included in the abstract.
- You just need to think about *"We propose XXX, XXX solves the YYY problem and have several benefits, we demonstrate its benefits through ZZZ way."*

Introduction

- Introduction is an extension of the abstract, including background, motivation, method overview and major contributions.
- You should not include too many related works here, you should only discuss the most important ones.

Introduction

- Introduction is an extension of the abstract, including background, motivation, method overview and major contributions.
- You should not include too many related works here, you should only discuss the most important ones.
- Contribution overview typically includes 3 or 4 points, when writing major contributions, think about the following. **This is also what you should think about throughout the project**.
  - What is unique about your proposed method?
  - What is the scope of the problems that your method can solve?
  - Does your work have some theoretical results or insights?
  - How strong and comprehensive are your experimental results?

Introduction

- Introduction is an extension of the abstract, including background, motivation, method overview and major contributions.
- You should not include too many related works here, you should only discuss the most important ones.
- Contribution overview typically includes 3 or 4 points, when writing major contributions, think about the following. **This is also what you should think about throughout the project**.
  - What is unique about your proposed method?
  - What is the scope of the problems that your method can solve?
  - Does your work have some theoretical results or insights?
  - How strong and comprehensive are your experimental results?
- You can include an overview figure in the introduction.
- Sometimes, you can introduction some key notations and terminology you may use in the end of the introduction.

Related Works

- Try to cite as many works as possible if this section is not overlength (typically half to one page).
- Summarize the related work in a structured way.
- More related to your work, more details you should include.
  - Examples w/o details: *many recent works (Ref1, Ref2, Ref3) investigate this problem.*
  - Example w/ details: *many recent works investigate this problem, Ref1 works on XXX, Ref2 works on YYY, but has issues of ZZZ ...*
- Properly use "citep" and "citet".
- If a paper is published, do not use the preprint version in the reference.

Preliminary & Methodology

- More detailed description on the basis of this work.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.
- Use **formal** and **accurate** words to describe your methods.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.
- Use **formal** and **accurate** words to describe your methods.
- Except common knowledge, all claims you make should be supported by either theoretical derivation, empirical observations and cited references.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.
- Use **formal** and **accurate** words to describe your methods.
- Except common knowledge, all claims you make should be supported by either theoretical derivation, empirical observations and cited references.
- If you would include some theorems from other papers, you should only cite the theorems that **directly** help build your theoretical contributions.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.
- Use **formal** and **accurate** words to describe your methods.
- Except common knowledge, all claims you make should be supported by either theoretical derivation, empirical observations and cited references.
- If you would include some theorems from other papers, you should only cite the theorems that **directly** help build your theoretical contributions.
- For theoretical work, you should include major lemmas, theorems, propositions in the logical chain.
- For empirical work, provide the pseudo-code of your algorithm when necessary.

Preliminary & Methodology

- More detailed description on the basis of this work.
- Notation and terminology should be consistent.
- Use **formal** and **accurate** words to describe your methods.
- Except common knowledge, all claims you make should be supported by either theoretical derivation, empirical observations and cited references.
- If you would include some theorems from other papers, you should only cite the theorems that **directly** help build your theoretical contributions.
- For theoretical work, you should include major lemmas, theorems, propositions in the logical chain.
- For empirical work, provide the pseudo-code of your algorithm when necessary.
- Consider using diagrams or geometric interpretations when necessary.

Experiments

- Major settings should be in the main text so that the reader will not be confused about your **experiment design**.
- Minor settings can be put in the appendix, these minor settings are mainly **implementation details**.

Experiments

- Major settings should be in the main text so that the reader will not be confused about your **experiment design**.
- Minor settings can be put in the appendix, these minor settings are mainly **implementation details**.
- Experiments on the most popular and the most challenging datasets should be in the main text, this will facilitate the reader to compare your method with baselines and understand the scalability of your algorithm.
  - Common settings for baselines: same hyper-parameters as in the original paper or optimal hyper-parameters.
- Ablation studies are expected if you algorithm includes some unique hyper-parameters.

Experiments

- Major settings should be in the main text so that the reader will not be confused about your **experiment design**.
- Minor settings can be put in the appendix, these minor settings are mainly **implementation details**.
- Experiments on the most popular and the most challenging datasets should be in the main text, this will facilitate the reader to compare your method with baselines and understand the scalability of your algorithm.
  - Common settings for baselines: same hyper-parameters as in the original paper or optimal hyper-parameters.
- Ablation studies are expected if you algorithm includes some unique hyper-parameters.
- Wisely use tables or figures, table highlights the numerical comparison, while figures are appropriate to demonstrate the trends.

Conclusion

- The conclusion should emphasize your contribution again.
- Mention your potential future works in the last sentence if possible.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.
- The curve width, the font of the axis labels and the legend should be properly set so that it can **be printed clearly**.
  - The width is typically set 2 or 3 in matplotlib.
  - The font should be set bigger than what your perceived in the screen.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.
- The curve width, the font of the axis labels and the legend should be properly set so that it can **be printed clearly**.
  - The width is typically set 2 or 3 in matplotlib.
  - The font should be set bigger than what your perceived in the screen.
- If labels or legends involves mathematical notation, use latex to make sure it is rendered properly. Do not use latex source code in the figure.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.
- The curve width, the font of the axis labels and the legend should be properly set so that it can **be printed clearly**.
  - The width is typically set 2 or 3 in matplotlib.
  - The font should be set bigger than what your perceived in the screen.
- If labels or legends involves mathematical notation, use latex to make sure it is rendered properly. Do not use latex source code in the figure.
- If the figure has several sub-figures, these sub-figures should be aligned.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.
- The curve width, the font of the axis labels and the legend should be properly set so that it can **be printed clearly**.
  - The width is typically set 2 or 3 in matplotlib.
  - The font should be set bigger than what your perceived in the screen.
- If labels or legends involves mathematical notation, use latex to make sure it is rendered properly. Do not use latex source code in the figure.
- If the figure has several sub-figures, these sub-figures should be aligned.
- The caption of the figures should be informative enough so that the reader can understand the *rough meaning* of your figure without reading the main text.

Figures

- Figures must be in PDF format. Do not use JPG or PNG format.
- The curve width, the font of the axis labels and the legend should be properly set so that it can **be printed clearly**.
  - The width is typically set 2 or 3 in matplotlib.
  - The font should be set bigger than what your perceived in the screen.
- If labels or legends involves mathematical notation, use latex to make sure it is rendered properly. Do not use latex source code in the figure.
- If the figure has several sub-figures, these sub-figures should be aligned.
- The caption of the figures should be informative enough so that the reader can understand the *rough meaning* of your figure without reading the main text.
- More about how to design figures.

Tables

- The width of each column and each row should not vary a lot.

Tables

- The width of each column and each row should not vary a lot.
- Do not use too many horizontal or vertical lines in the table, but the headers should be separate from the main content.
- Properly use bold lines, solid lines and dashed lines.

Tables

- The width of each column and each row should not vary a lot.
- Do not use too many horizontal or vertical lines in the table, but the headers should be separate from the main content.
- Properly use bold lines, solid lines and dashed lines.
- If the table involves comparisons with baselines, make sure to highlight your proposed method and the best results.

Tables

- The width of each column and each row should not vary a lot.
- Do not use too many horizontal or vertical lines in the table, but the headers should be separate from the main content.
- Properly use bold lines, solid lines and dashed lines.
- If the table involves comparisons with baselines, make sure to highlight your proposed method and the best results.
- The caption of the tables should be informative enough so that the reader can understand the *rough meaning* of your figure without reading the main text.

Tables

- The width of each column and each row should not vary a lot.
- Do not use too many horizontal or vertical lines in the table, but the headers should be separate from the main content.
- Properly use bold lines, solid lines and dashed lines.
- If the table involves comparisons with baselines, make sure to highlight your proposed method and the best results.
- The caption of the tables should be informative enough so that the reader can understand the *rough meaning* of your figure without reading the main text.
- More about how to make nice tables.

Notation and Terminology

Try to make your notation in different papers as consistent as possible, otherwise it would be troublesome to write your thesis.

- Thin Latin letters to represent scalar.
  - Uppercase letters to represent constants, e.g., C, N, M.
  - Lowercase letters to represent variables, e.g., x, y, z.
- Bold lowercase Latin letters to represent vectors, e.g., $\boldsymbol{x}$, $\boldsymbol{y}$.
- Bold uppercase Latin letters to represent matrices or tensors, e.g., $\mathbf{X}$, $\mathbf{Y}$.
- Some Geek letters to represent specific values, e.g., Λ, ϵ, δ.

Notation and Terminology

- Functions are usually represented by thin Latin letters, e.g., f, g, h.
- Sets are usually represented by calligraphic letters, e.g. $\mathcal{S}$, $\mathcal{T}$, $\mathcal{U}$, $\mathcal{V}$.
- Use tilde or hat to represent the approximation or estimation of some variable, e.g. $\widehat{\mathbf{X}}$, $\widetilde{\mathbf{x}}$.
  - Use widehat and widetilde in Latex to make the notation nicer.
- Use overline and underline to represent the upper bound and the lower bound of some variable, e.g. $\overline{\mathbf{X}}$, $\underline{\mathbf{X}}$.

Lemma, Theorem, Proposition, Corollary

- **Lemma**: a helper fact that leads to a more significant results.
- **Theorem**: significant results, major results.
- **Proposition**: less important, minor results.
- **Corollary**: short and easy results derived from theorems.

Thank You!