Meta Architecture Search

Albert Shaw[1], Wei Wei[2], Weiyang Liu[1], Le Song[1,3], Bo Dai[1,2]

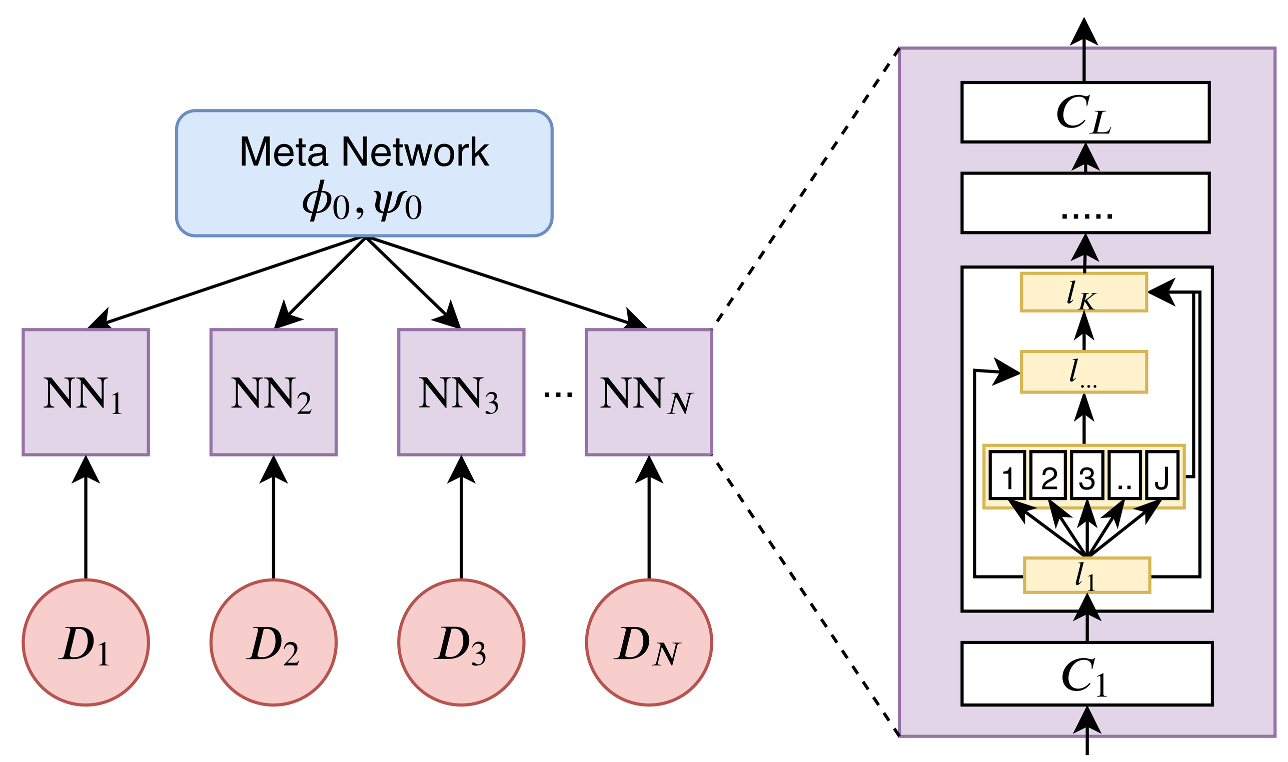
Motivation

Recent work has indicated that the optimal Neural Network architectures can vary between even similar tasks. On new datasets, NAS is often individually run for each task which can be quite costly. With Meta Architecture Search, we aim to learn task-agnostic representations that can be used to speed up the process of architecture search on a large number of tasks.

Main Contribution:

- We propose a Bayesian inference view of architecture learning.
- We derive a variational inference method to learn the architectures of an entire set of tasks simultaneously using the optimization embedding technique to design the parameterization of the posterior.
- Demonstrates a concrete algorithm for Meta Architecture Search that can use a prior trained over multiple tasks to find competitive models for unseen datasets with just quick adaptation.

Bayesian View of Architecture Search



We consider NAS as an operation selection problem.

$$x_k = \sum_{i=1}^{k-1} \left(z_{i,k}^\top \mathcal{A}_i(\theta)\right) \circ x_i := \sum_{i=1}^{k-1}\sum_{l=1}^{L} z_{i,k}^l \phi_i^l(x_i;\theta),$$

Assume the probabilistic model as

$$\theta \sim \mathcal{N}(\mu, \sigma^2),$$
$$z_{i,k} \sim \mathcal{C}ategorial(\alpha_{i,k}),\ \ k = 1, \ldots, K,$$
$$y \sim p(y|x;\theta,z) \propto \exp\left(-\ell\left(f(x;\theta,z), y\right)\right),$$

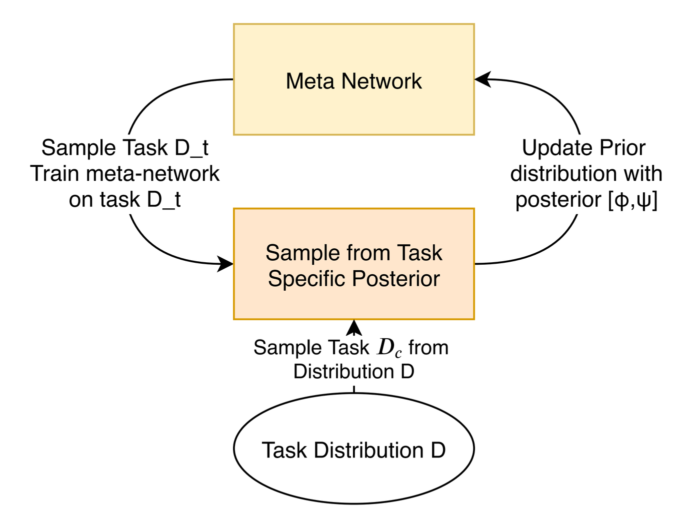
$W(\mu,\sigma,\alpha)$ can be estimated via Maximum Likelihood Estimation (MLE).

$$\max_W \widehat{\mathbb{E}}_{x,y}\left[\log \int p(y|x;\theta,z)\, p(z;\alpha)\, p(\theta;\mu,\sigma)\, dz d\theta\right].$$

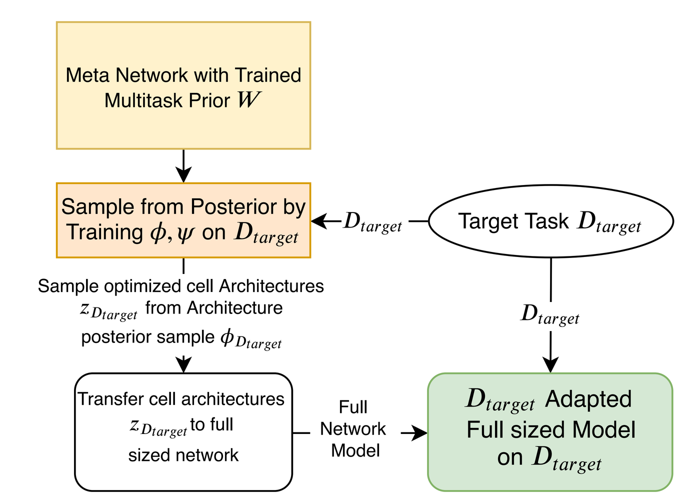
We extend this to the meta-learning setting with many tasks $\mathcal{D}_t$. Weight and architecture priors (μ, σ, α) are shared between all tasks.

$$\max_W \widehat{\mathbb{E}}_{\mathcal{D}_t}\widehat{\mathbb{E}}_{(x,y)\sim\mathcal{D}_t}\left[\log \int p(y|x;\theta,z)\, p(z;\alpha)\, p(\theta;\mu,\sigma)\, dz d\theta\right]$$

Meta Architecture Search and Adaptation



(a) Meta Architecture Search



(b) Quick Architecture Adaptation

Variational Inference by Optimization Embedding

Variational Bayesian Inference: Since the Maximum Likelihood Estimation (MLE) is intractable due to the integral over latent variable z, we consider optimizing the Evidence Lower Bound (ELBO).

$$\widehat{\mathbb{E}}_{\mathcal{D}}\Bigg[\max_{\phi_{\mathcal{D}},\psi_{\mathcal{D}}} \underbrace{\widehat{\mathbb{E}}_{x,y}\mathbb{E}_{\xi,\epsilon}\left[-\ell\left(f\left(x;\theta_{\mathcal{D}}(\epsilon,\psi), z_{\mathcal{D}}(\xi,\phi)\right), y\right)\right] - \log\frac{q_\phi(z|\mathcal{D})}{p(z;\alpha)} - \log\frac{q_\psi(\theta|\mathcal{D})}{p(\theta;\mu,\sigma)}}_{L(\phi_{\mathcal{D}},\psi_{\mathcal{D}};W)}\Bigg].$$

Optimization Embedding: With this objective, we follow the parameterized Coupled Variational Bayes derivation for embedding the optimization procedure for (ϕ, ψ).

$$\left[\phi_{\mathcal{D}}^t, \psi_{\mathcal{D}}^t\right] = \eta_t \hat{g}_{\phi_{\mathcal{D}},\psi_{\mathcal{D}}}(\mathcal{D}, W) + \left[\phi_{\mathcal{D}}^{t-1}, \psi_{\mathcal{D}}^{t-1}\right] \quad \text{where} \quad \hat{g}_{\phi_{\mathcal{D}},\psi_{\mathcal{D}}}(\mathcal{D}, W) = \frac{\partial \widehat{L}}{\partial(\phi_{\mathcal{D}},\psi_{\mathcal{D}})}$$

$\widehat{L}$ is the stochastic approximation for $L(\phi_{\mathcal{D}}, \psi_{\mathcal{D}}; W)$

We initialize $(\phi^0, \psi^0) = W$ where W is shared across all the tasks. After T optimization steps, we obtain $(\phi_{\mathcal{D}}^T, \psi_{\mathcal{D}}^T)$, which leads to $\left(\theta_{\mathcal{D}}^T(\xi, \psi_{\mathcal{D}}^T), z_{\mathcal{D}}(\xi, \phi_{\mathcal{D}}^T)\right)$.
In other words, we derive the parameterization of $q(\theta|\mathcal{D})$ and $q(z|\mathcal{D})$ by unfolding the optimization.

$$\max_W \widehat{\mathbb{E}}_{\mathcal{D}}\widehat{\mathbb{E}}_{x,y}\mathbb{E}_{\xi,\epsilon}\Bigg[\underbrace{-\ell\left(f\left(x;\theta_{\mathcal{D}}^T(\epsilon,\psi), z_{\mathcal{D}}^T(\xi,\phi)\right), y\right) - \log\frac{q_{\phi_{\mathcal{D}}^T}(z|\mathcal{D})}{p(z;\alpha)} - \log\frac{q_{\psi_{\mathcal{D}}^T}(\theta|\mathcal{D})}{p(\theta;\mu,\sigma)}}_{\widehat{L}(x,y,\epsilon,\xi;W)}\Bigg].$$

Results Tables

Classification Accuracy on Cifar10

Architecture	Top-1 Test Error	Params (M)	Search Time (Gpu Days)
DenseNet-BC	3.46	25.6	-
NASNet-A + cutout	**2.65**	3.3	1800
AmoebaNet-A + cutout	3.34 ± 0.06	3.2	3150
AmoebaNet-B + cutout	2.55 ± 0.05	**2.8**	3150
Hierarchical Evo	3.75 ± 0.12	15.7	**300**
DARTS (1st order)	3.00 ± 0.14	3.3	1.5
DARTS (2nd order)	$\mathbf{2.76 \pm 0.09}$	3.3	4
SNAS (single-level)	2.85 ± 0.02	**2.8**	1.5
ENAS + cutout	2.89	4.6	**0.5**
PNAS	3.41 ± 0.09	3.2	225
SMASH	4.03	16	1.5
BASE(Multi-task Prior)	3.18	**3.22**	8
BASE(`Imagenet32`)	3.00	3.29	**0.04 Adap / 8 Meta**
BASE(`CIFAR10`)	**2.83**	3.07	0.05 Adap / 8 Meta

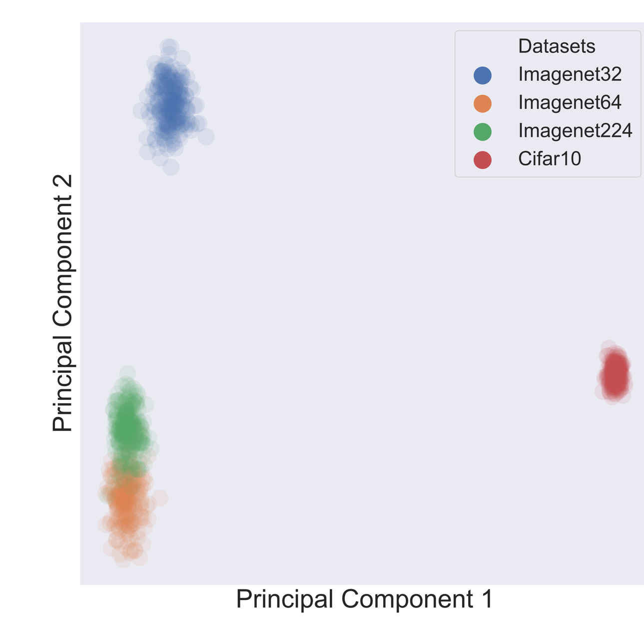
Classification Accuracy on Imagenet

Architecture	Top-1 Err	Top-5 Err	MACs (M)	Search Time (GPU Days)
NASNet-A	26.0	8.4	564	1800
NASNet-B	27.2	8.7	**488**	1800
NASNet-C	27.5	9.0	558	1800
AmoebaNet-A	25.5	8.0	555	3150
AmoebaNet-B	26.0	8.5	555	3150
AmoebaNet-C	**24.3**	**7.6**	570	3150
PNAS	25.8	8.1	588	**225**
DARTS	**26.9**	**9.0**	595	4
SNAS	27.3	9.2	**522**	**1.5**
BASE (Multi-task Prior)	26.12	8.52	**544**	**0 Adap / 8 Meta**
BASE (`Imagenet`)	**25.71**	**8.08**	559	0.04 Adap / 8 Meta

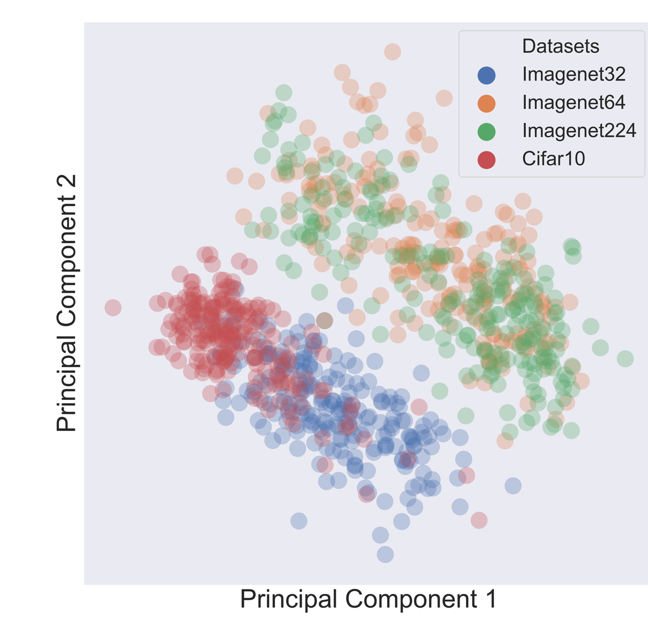
Bayesian meta-Architecture SEarch (BASE) Algorithm

Initialize meta-network parameters W_0.
for $e = 1, \ldots, E$ **do**
  Sample C tasks $\{\mathcal{D}_c\}_{c=1}^C \sim \mathcal{D}$.
  for $\mathcal{D}_c$ in $\mathcal{D}$ **do**
    Sample $\{x_t, y_t\}_{t=1}^T \sim \mathcal{D}_c$.
    Let $\phi_c^0, \psi_c^0 = W_{e-1}$.
    for $t = 1, \ldots, T$ **do**
      Sample $\xi \sim \mathcal{G}(0, 1)$.
      Update $[\phi_c^t, \psi_c^t] = [\phi_c^{t-1}, \psi_c^{t-1}] - \eta\nabla_{\phi_c^{t-1},\psi_c^{t-1}}\widehat{L}(f(x_t;\phi_c^{t-1},\psi_c^{t-1},\xi), y_t)$.
  Update $W_e = W_{e-1} + \lambda\frac{1}{C}\sum_{c=1}^C([\phi_c^T, \psi_c^T] - W_{e-1})$.

Visualization of Weights and Architecture Posterior Distributions

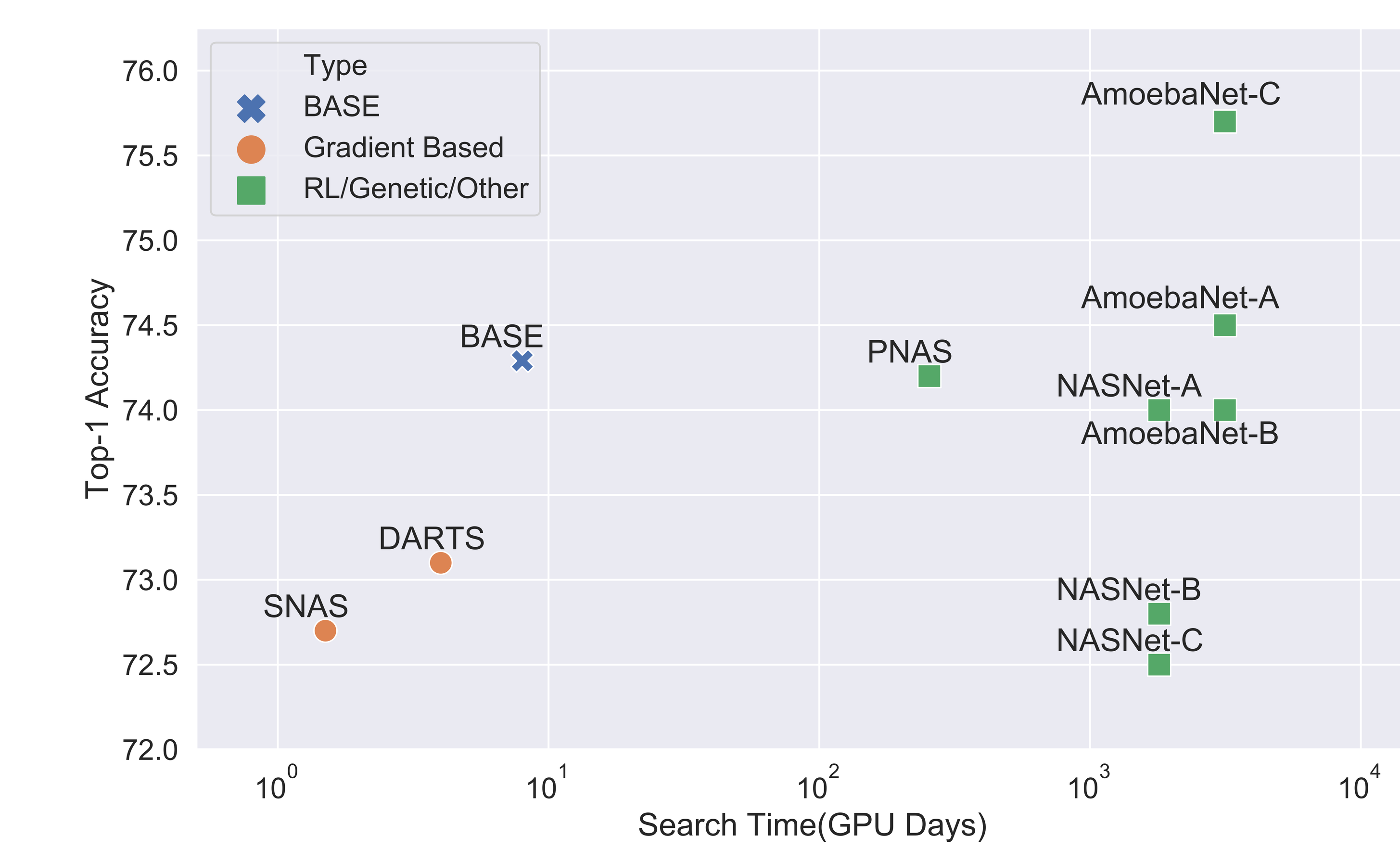


PCA of weights



PCA of architecture

Imagenet Accuracy vs Search Time



[1]Georgia Institute of Technology, [2]Google Research, [3]Ant Financial

Mailto: `ashaw596@gatech.edu`