MixMatch

❖ MixMatch: A Holistic Approach to Semi-Supervised Learning (NeurIPs, 2019)

- Google Research에서 연구하였으며, 2022년 12월 19일 기준 1824회 인용됨
- 기존 Semi-Supervised Learning 방법 Consistency Regularization, Entropy Minimization, Traditional Regularization (Mix Up)을 결합한 방법론

MixMatch: A Holistic Approach to Semi-Supervised Learning

David Berthelot
Google Research
dberth@google.com

Nicholas Carlini
Google Research
ncarlini@google.com

Ian Goodfellow
Work done at Google
ian-academic@mailfence.com

Avital Oliver
Google Research
avitalo@google.com

Nicolas Papernot
Google Research
papernot@google.com

Colin Raffel
Google Research
craffel@google.com

Abstract

Semi-supervised learning has proven to be a powerful paradigm for leveraging unlabeled data to mitigate the reliance on large labeled datasets. In this work, we unify the current dominant approaches for semi-supervised learning to produce a new algorithm, MixMatch, that guesses low-entropy labels for data-augmented unlabeled examples and mixes labeled and unlabeled data using MixUp. MixMatch obtains state-of-the-art results by a large margin across many datasets and labeled data amounts. For example, on CIFAR-10 with 250 labels, we reduce error rate by a factor of 4 (from 38% to 11%) and by a factor of 2 on STL-10. We also demonstrate how MixMatch can help achieve a dramatically better accuracy-privacy trade-off for differential privacy. Finally, we perform an ablation study to tease apart which components of MixMatch are most important for its success. We release all code used in our experiments.[1]

MixMatch

Background

❖ MixMatch: A Holistic Approach to Semi-Supervised Learning

- MixMatch는 기존 Semi-Supervised Learning 방법론 세 가지를 결합한 형태
  1. Consistency Regularization
  2. Entropy Minimization
  3. Traditional Regularization (Mix Up)

$$Loss = \underset{Supervised}{L_S} + \underset{Unsupervised}{L_U}$$

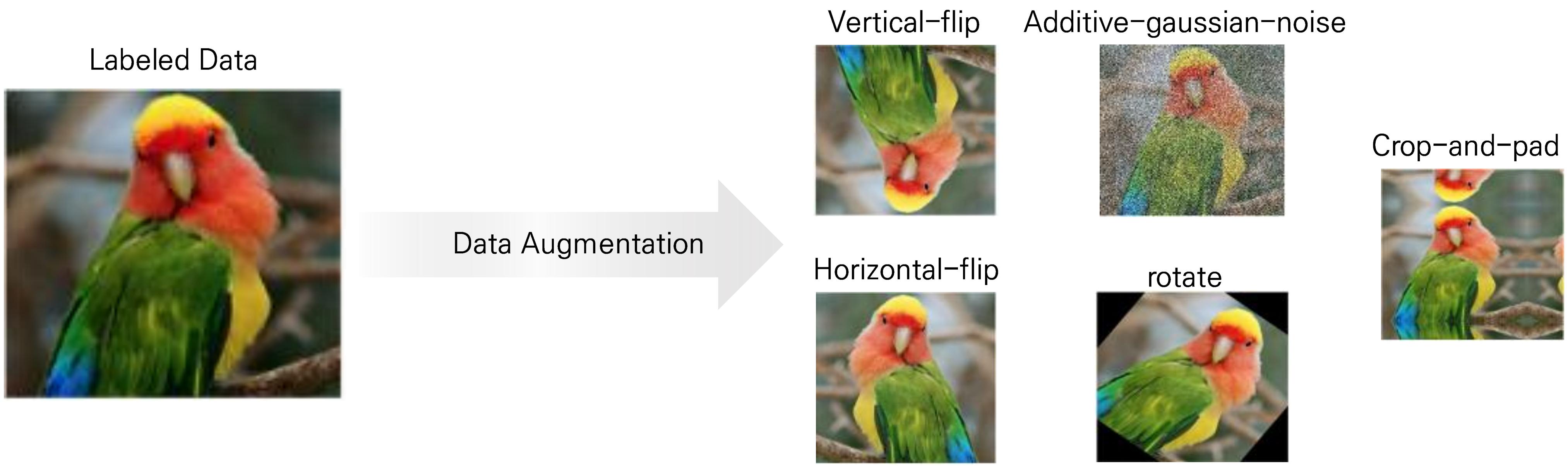
Consistency Regularization + **Entropy Minimization** + **Traditional Regularization (MixUp)**

MixMatch

Background

❖ Consistency Regularization

- Data Augmentation
  - ✓ 지도 학습 : 데이터에 약간의 변형을 가하더라도 클래스 정보는 영향을 받지 않음
  - ✓ 비지도 학습 : 레이블이 없는 데이터에 증강 기법을 적용하면 클래스 예측 분포가 달라짐

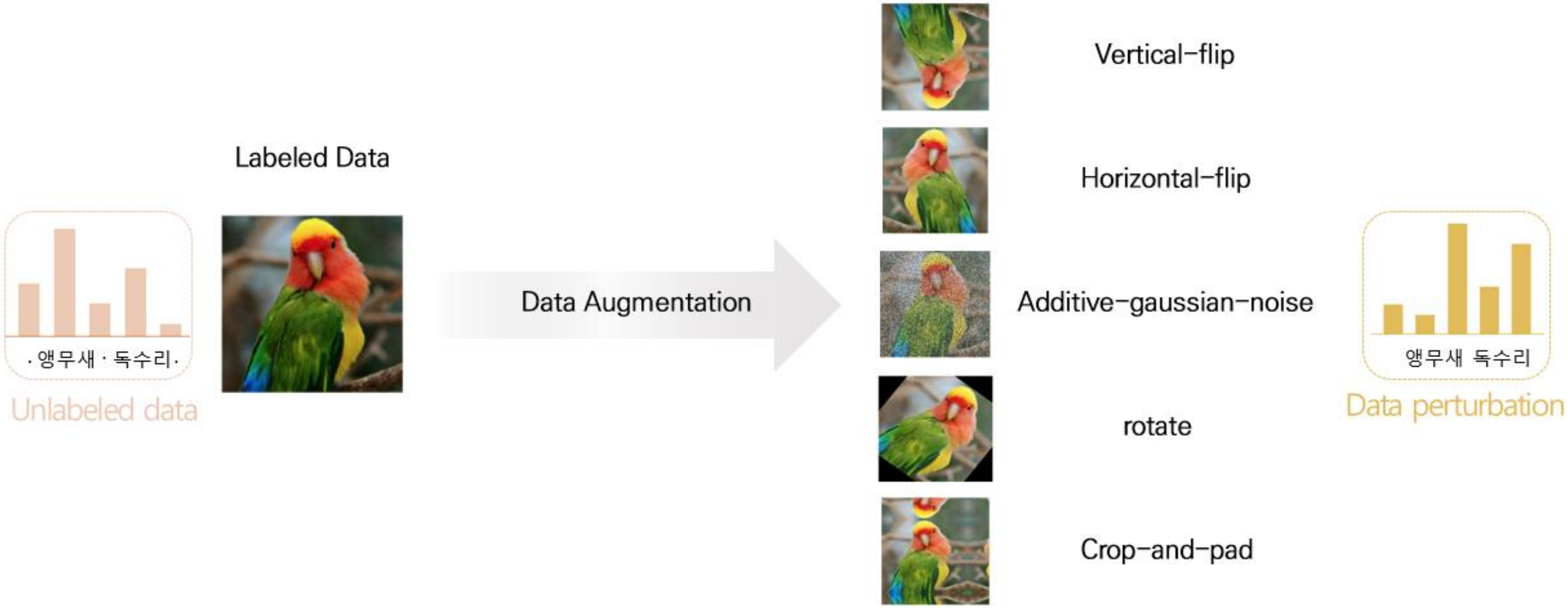


MixMatch

Background

❖ Consistency Regularization

- Data Augmentation
  - ✓ 지도 학습 : 데이터에 약간의 변형을 가하더라도 클래스 정보는 영향을 받지 않음
  - ✓ 비지도 학습 : 레이블이 없는 데이터에 증강 기법을 적용하면 클래스 예측 분포가 달라짐



$$\text{Min} \left\| f_\theta(x_u) - f_\theta(\text{Augment}(x_u)) \right\|_2^2$$

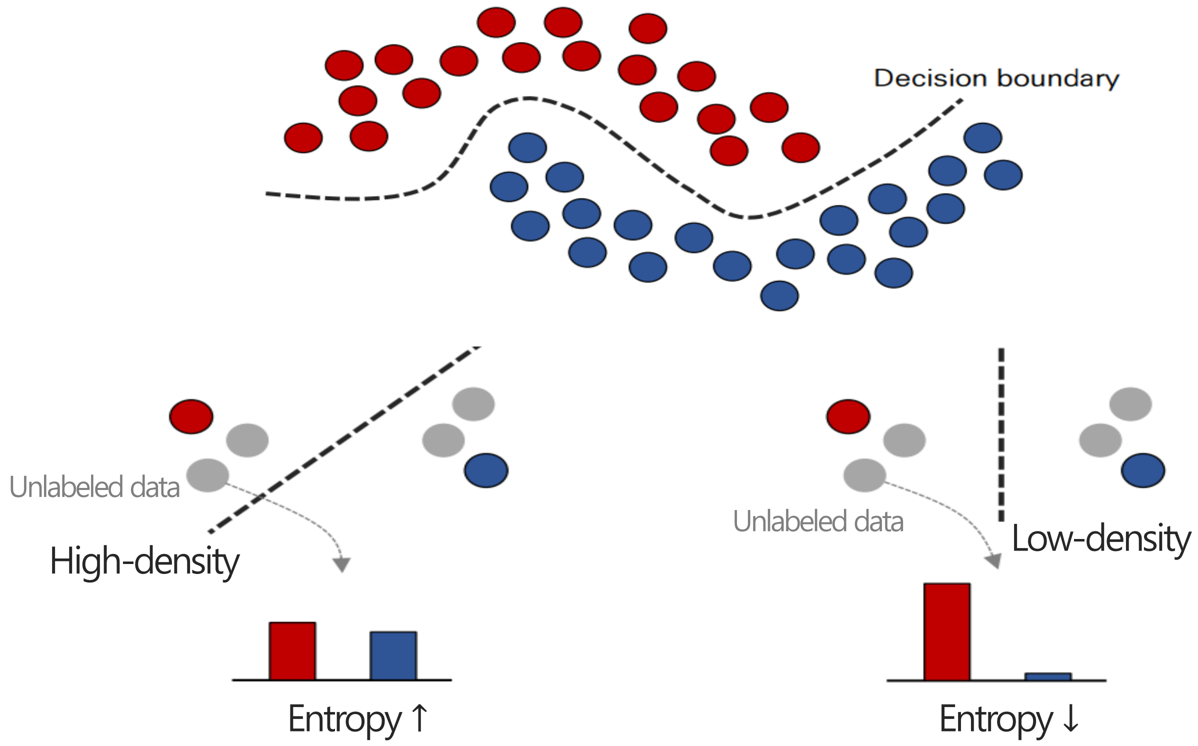
목표: Unlabeled data에 Augmentation을 수행해도 동일한 클래스 분포로 예측하도록 학습

MixMatch

Background

❖ Entropy Minimization

- Unlabeled data에 대한 예측값의 confidency를 높이는 것을 목적으로 함
- 결정 경계는 데이터의 저밀도 지역에서 형성될 것이라는 가정에 기초해서 unlabeled data의 출력의 entropy를 minimization하는 기법
  - ✓ Temperature sharpening을 통해 entropy minimization을 간접적으로 사용

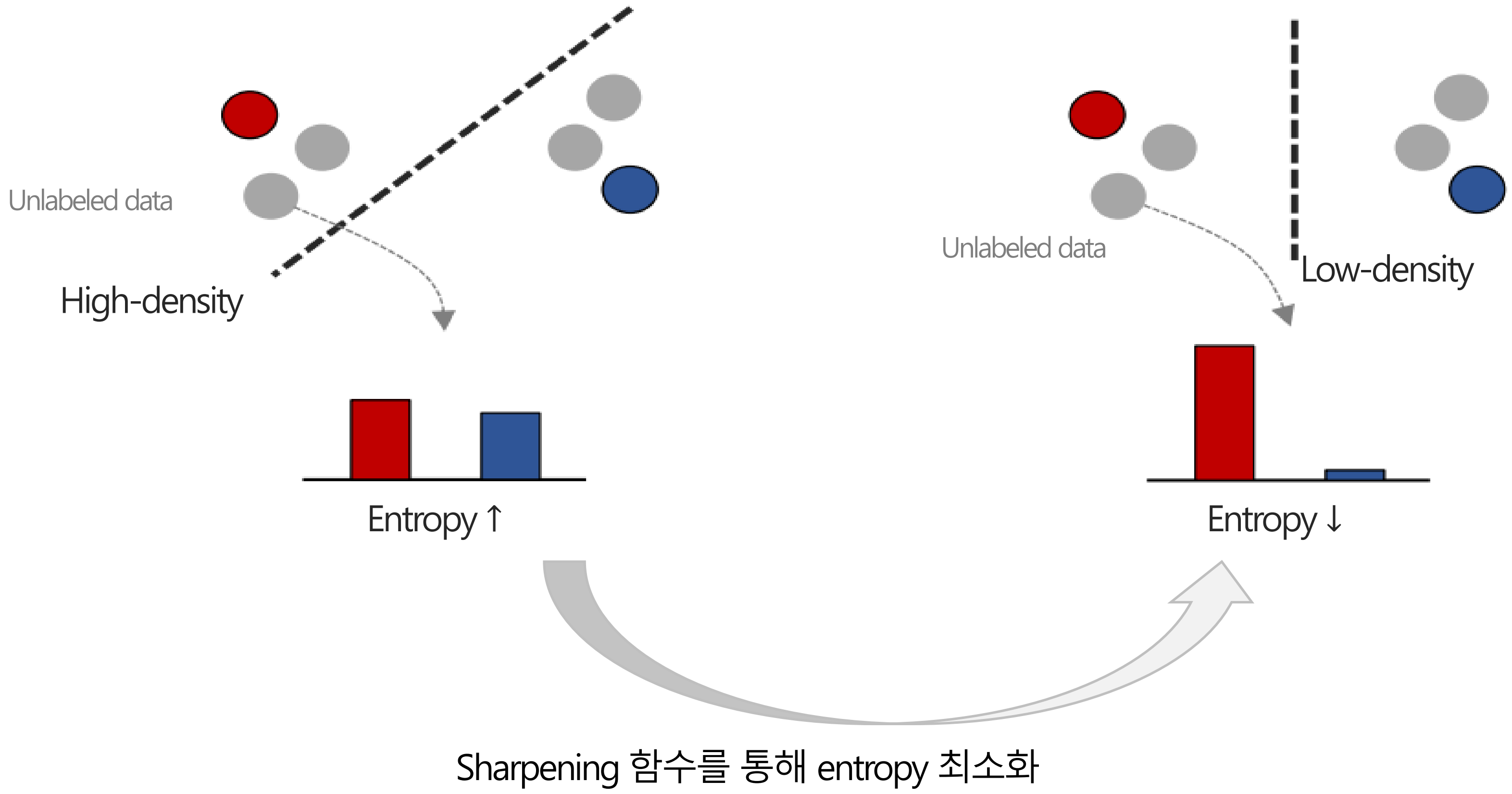


MixMatch

Background

❖ Entropy Minimization

- Unlabeled data에 대한 예측값의 confidency를 높이는 것을 목적으로 함



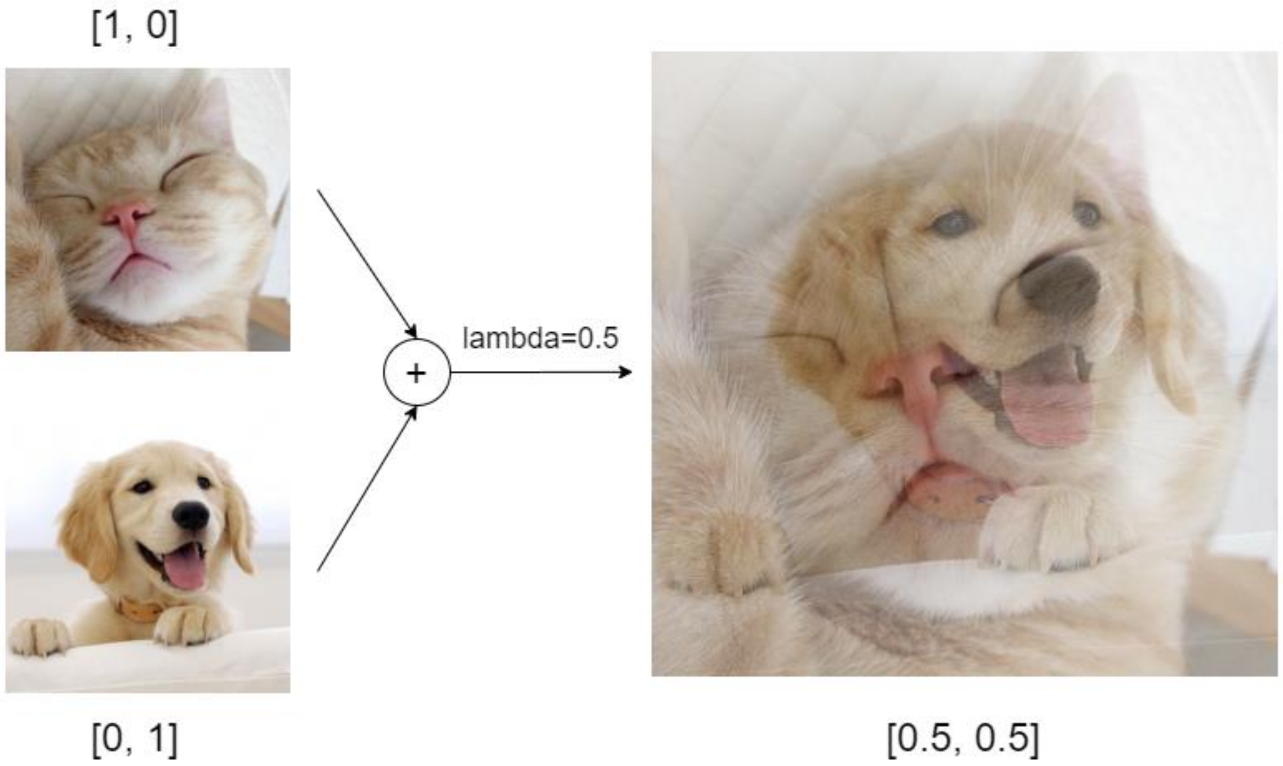
목표: Unlabeled data에 예측 값의 confidence를 높이도록 학습

MixMatch

Background

❖ Traditional Regularization (Mixup)

- Supervised : 데이터와 레이블 각각을 interpolation하여 새로운 데이터 생성
  - ✓ 알려진 지점의 값 사이(중간)에 위치한 값을 알려진 값으로부터 추정하는 것 → interpolation
  - ✓ Overfitting 방지하여 일반화 성능 향상

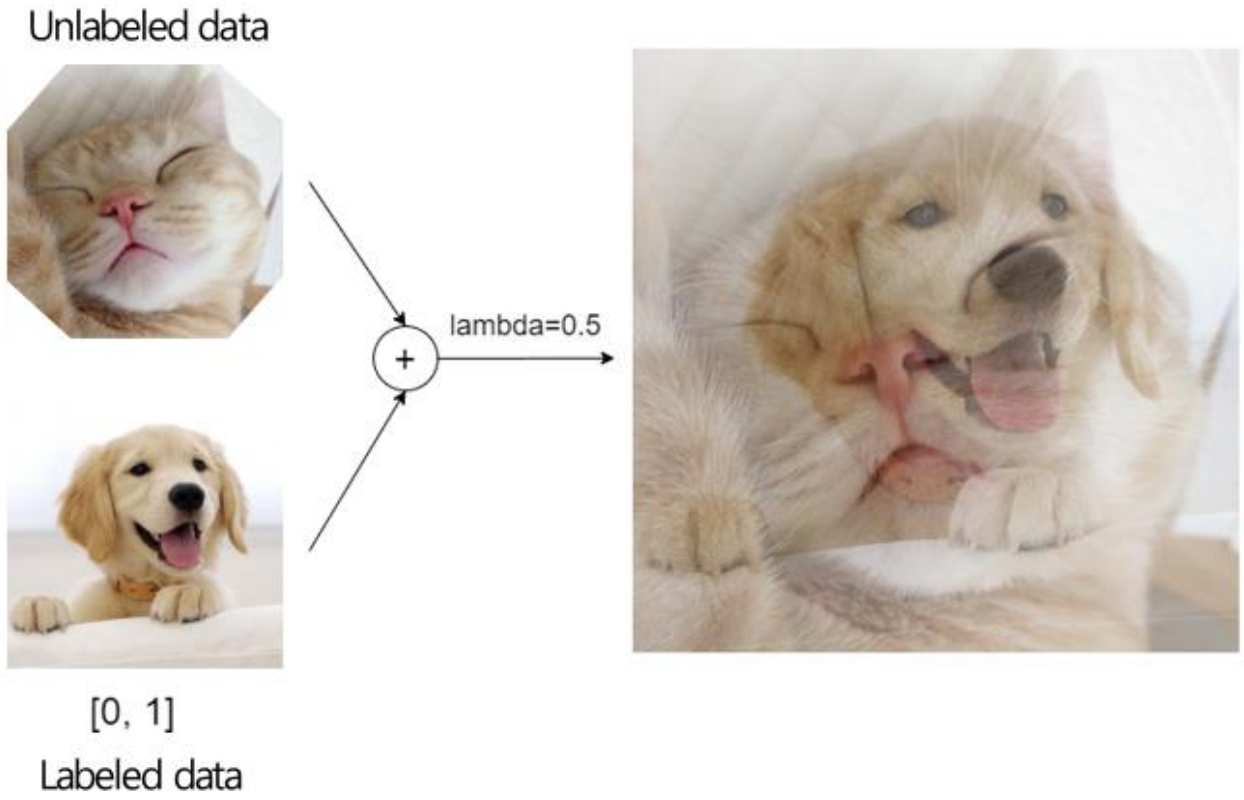


MixMatch

Background

❖ Traditional Regularization (Mixup)

- Unsupervised : 모델이 Unlabeled 데이터에 대한 생성한 가짜 레이블(distribution) 사용
  - ✓ Overfitting 방지하여 일반화 성능 향상

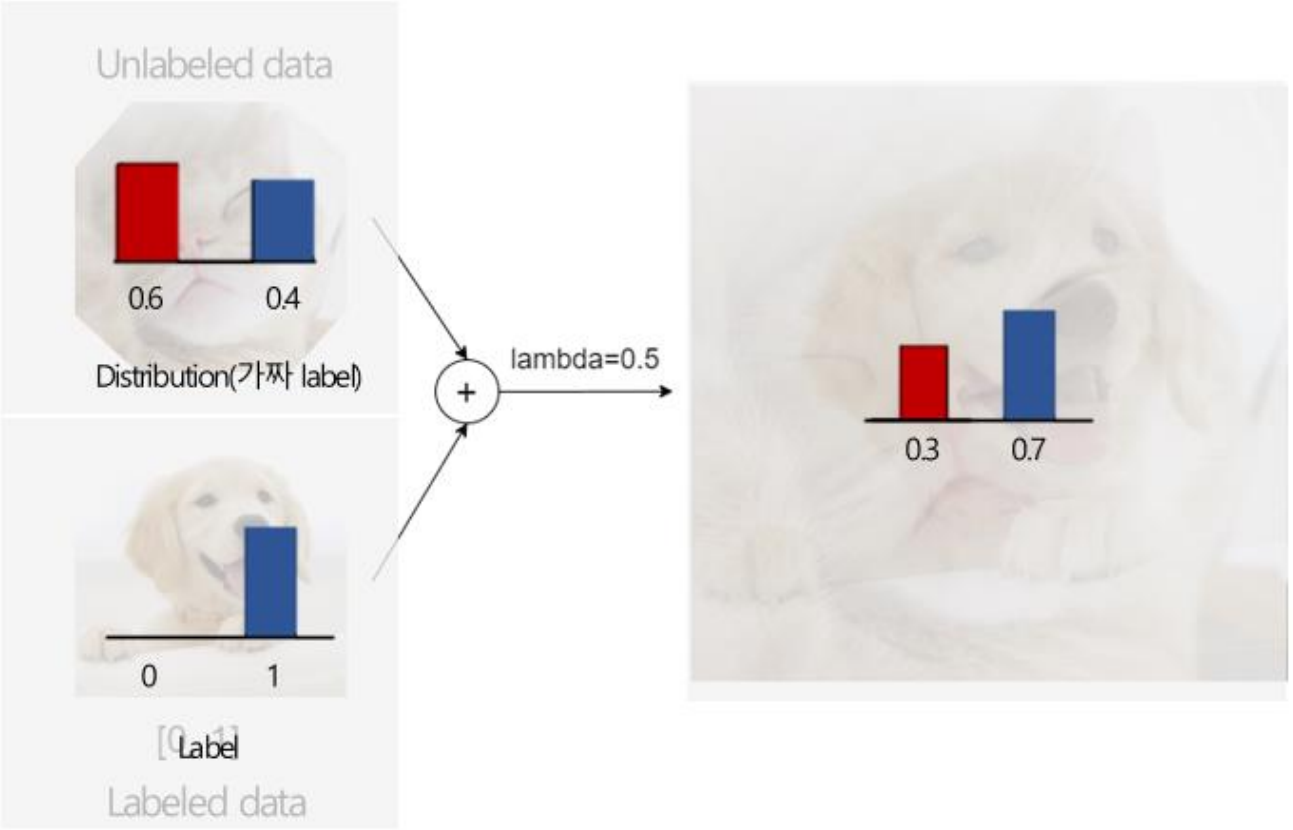


MixMatch

Background

❖ Traditional Regularization (Mixup)

- Unsupervised : 모델이 Unlabeled 데이터에 대한 생성한 가짜 레이블(distribution) 사용
  - ✓ Overfitting 방지하여 일반화 성능 향상



MixMatch

Framework

Algorithm 1 MixMatch takes a batch of labeled data $\mathcal{X}$ and a batch of unlabeled data $\mathcal{U}$ and produces a collection $\mathcal{X}'$ (resp. $\mathcal{U}'$) of processed labeled examples (resp. unlabeled with guessed labels).

Input: Batch of labeled examples and their one-hot labels $\mathcal{X} = \big((x_b, p_b); b \in (1, \dots, B)\big)$, batch of unlabeled examples $\mathcal{U} = \big(u_b; b \in (1, \dots, B)\big)$, sharpening temperature T, number of augmentations K, Beta distribution parameter α for MixUp.
for $b = 1$ **to** B **do**
  $\hat{x}_b = \mathrm{Augment}(x_b)$ *// Apply data augmentation to x_b*
  for $k = 1$ **to** K **do**
    $\hat{u}_{b,k} = \mathrm{Augment}(u_b)$ *// Apply k^{th} round of data augmentation to u_b*
  end for
  $\bar{q}_b = \frac{1}{K} \sum_k \mathrm{p_{model}}(y \mid \hat{u}_{b,k}; \theta)$ *// Compute average predictions across all augmentations of u_b*
  $q_b = \mathrm{Sharpen}(\bar{q}_b, T)$ *// Apply temperature sharpening to the average prediction (see eq. (7))*
end for
$\hat{\mathcal{X}} = \big((\hat{x}_b, p_b); b \in (1, \dots, B)\big)$ *// Augmented labeled examples and their labels*
$\hat{\mathcal{U}} = \big((\hat{u}_{b,k}, q_b); b \in (1, \dots, B), k \in (1, \dots, K)\big)$ *// Augmented unlabeled examples, guessed labels*
$\mathcal{W} = \mathrm{Shuffle}\big(\mathrm{Concat}(\hat{\mathcal{X}}, \hat{\mathcal{U}})\big)$ *// Combine and shuffle labeled and unlabeled data*
$\mathcal{X}' = \big(\mathrm{MixUp}(\hat{\mathcal{X}}_i, \mathcal{W}_i); i \in (1, \dots, |\hat{\mathcal{X}}|)\big)$ *// Apply MixUp to labeled data and entries from $\mathcal{W}$*
$\mathcal{U}' = \big(\mathrm{MixUp}(\hat{\mathcal{U}}_i, \mathcal{W}_{i+|\hat{\mathcal{X}}|}); i \in (1, \dots, |\hat{\mathcal{U}}|)\big)$ *// Apply MixUp to unlabeled data and the rest of $\mathcal{W}$*
return $\mathcal{X}', \mathcal{U}'$

MixMatch

Framework

❖ Framework

- 입력 데이터는 mini batch마다의 labeled data χ, Unlabeled data $\mathcal{U}$ (1)
- Stochastic Data Augmentation: (2-6)
  - ✓ Labeled data에 사전에 정의한 Image Augmentation 기법 중 하나를 임의로 1번 적용
  - ✓ Unlabeled data에 사전에 정의한 Image Augmentation 기법 중 하나를 임의로 K번 적용

1: **Input:** Batch of labeled examples and their one-hot labels $\mathcal{X} = \big((x_b, p_b); b \in (1, \ldots, B)\big)$, batch of unlabeled examples $\mathcal{U} = \big(u_b; b \in (1, \ldots, B)\big)$, sharpening temperature T, number of augmentations K, Beta distribution parameter α for MixUp.

2: **for** $b = 1$ **to** B **do**
3: $\quad \hat{x}_b = \text{Augment}(x_b)$ // *Apply data augmentation to* x_b
4: $\quad$ **for** $k = 1$ **to** K **do**
5: $\quad\quad \hat{u}_{b,k} = \text{Augment}(u_b)$ // *Apply* k^{th} *round of data augmentation to* u_b
6: $\quad$ **end for**
7: $\quad \bar{q}_b = \frac{1}{K} \sum_k \text{p}_{\text{model}}(y \mid \hat{u}_{b,k}; \theta)$ // *Compute average predictions across all augmentations of* u_b
8: $\quad q_b = \text{Sharpen}(\bar{q}_b, T)$ // *Apply temperature sharpening to the average prediction (see eq. (7))*
9: **end for**

MixMatch

Framework

❖ Framework

- Label Guessing: (7)
  - ✓ Augmentation된 unlabeled data k개를 모델을 통해 나온 클래스 분포를 평균

2: **for** $b = 1$ **to** B **do**
3: $\quad \hat{x}_b = \text{Augment}(x_b)$ *// Apply data augmentation to x_b*
4: $\quad$ **for** $k = 1$ **to** K **do**
5: $\quad\quad \hat{u}_{b,k} = \text{Augment}(u_b)$ *// Apply k^{th} round of data augmentation to u_b*
6: $\quad$ **end for**
7: $\quad \boxed{\bar{q}_b = \frac{1}{K} \sum_k \mathrm{p_{model}}(y \mid \hat{u}_{b,k}; \theta)}$ *// Compute average predictions across all augmentations of u_b*
8: $\quad q_b = \text{Sharpen}(\bar{q}_b, T)$ *// Apply temperature sharpening to the average prediction (see eq. (7))*
9: **end for**

Label guessing

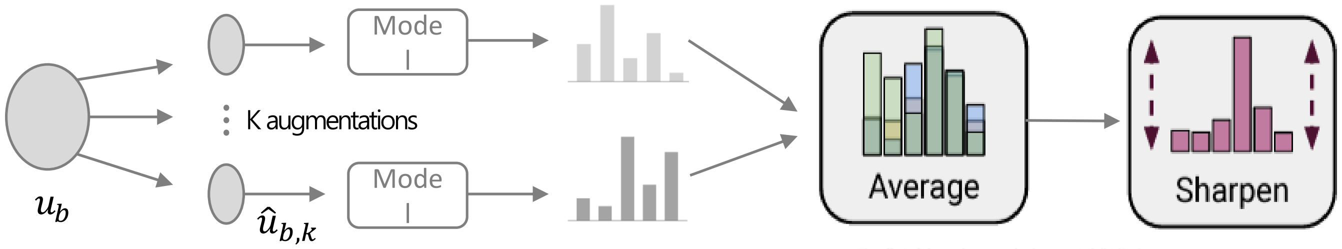
MixMatch

Framework

❖ Framework

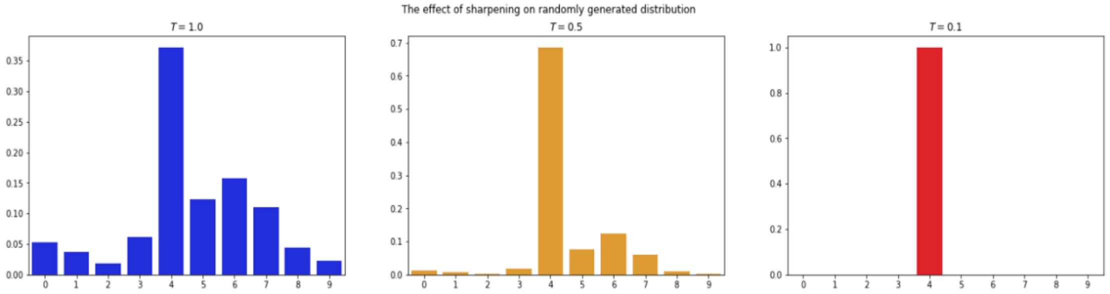
- Sharpening: (8)
  - ✓ Softmax Temperature(T)를 이용한 Entropy Minimization

2: **for** $b = 1$ **to** B **do**
3: $\hat{x}_b = \text{Augment}(x_b)$ *// Apply data augmentation to x_b*
4: **for** $k = 1$ **to** K **do**
5: $\hat{u}_{b,k} = \text{Augment}(u_b)$ *// Apply k^{th} round of data augmentation to u_b*
6: **end for**
7: $\bar{q}_b = \frac{1}{K}\sum_k \text{p}_{\text{model}}(y \mid \hat{u}_{b,k}; \theta)$ *// Compute average predictions across all augmentations of u_b*
8: $q_b = \text{Sharpen}(\bar{q}_b, T)$ *// Apply temperature sharpening to the average prediction (see eq. (7))*
9: **end for**



$$Sharpen(p,T)_i = \frac{p_i^{\frac{1}{T}}}{\sum_{j=1}^{\#\ of\ class} p_j^{\frac{1}{T}}},$$

$$T \rightarrow 0, Sharpen_T \rightarrow \text{one hot}$$



MixMatch

Framework

❖ Framework

- 앞서 labeled data와 unlabeled data에 augmentation을 통해 얻은 데이터와 분포(p, q)를 각각 $\hat{\chi}, \hat{u}$ 정의하고(10, 11), 이를 합친 후, 섞어 W 생성(12)
- MixUp: (13–15)
  - ✓ $\hat{\chi}, \hat{u}$ 를 W와 각각 MixUp 진행

10: $\hat{\mathcal{X}} = \big((\hat{x}_b, p_b); b \in (1, \ldots, B)\big)$ *// Augmented labeled examples and their labels*
11: $\hat{\mathcal{U}} = \big((\hat{u}_{b,k}, q_b); b \in (1, \ldots, B), k \in (1, \ldots, K)\big)$ *// Augmented unlabeled examples, guessed labels*
12: $\mathcal{W} = \text{Shuffle}\big(\text{Concat}(\hat{\mathcal{X}}, \hat{\mathcal{U}})\big)$ *// Combine and shuffle labeled and unlabeled data*
13: $\mathcal{X}' = \big(\text{MixUp}(\hat{\mathcal{X}}_i, \mathcal{W}_i); i \in (1, \ldots, |\hat{\mathcal{X}}|)\big)$ *// Apply MixUp to labeled data and entries from $\mathcal{W}$*
14: $\mathcal{U}' = \big(\text{MixUp}(\hat{\mathcal{U}}_i, \mathcal{W}_{i+|\hat{\mathcal{X}}|}); i \in (1, \ldots, |\hat{\mathcal{U}}|)\big)$ *// Apply MixUp to unlabeled data and the rest of $\mathcal{W}$*
15: **return** $\mathcal{X}', \mathcal{U}'$