MixMatch: A Holistic Approach to Semi-Supervised Learning

School of Industrial and Management Engineering, Korea University

Sangmin Kim

KOREA UNIVERSITY

DMQA

Contents

DMQA

MixMatch

❖ **MixMatch: A Holistic Approach to Semi-Supervised Learning** (NeurIPS 2019)

- Google research에서 연구된 논문이며, 2022년 8월 19일 기준 1,480회 인용됨

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

❖ Brief summary

- MixMatch는 기존 Semi-supervised learning(SSL) 방법 세가지를 결합한 방법론(holistic approach)
  1. **Consistency Regularization**
  2. **Entropy Minimization**
  3. **Traditional Regularization(MixUp)**

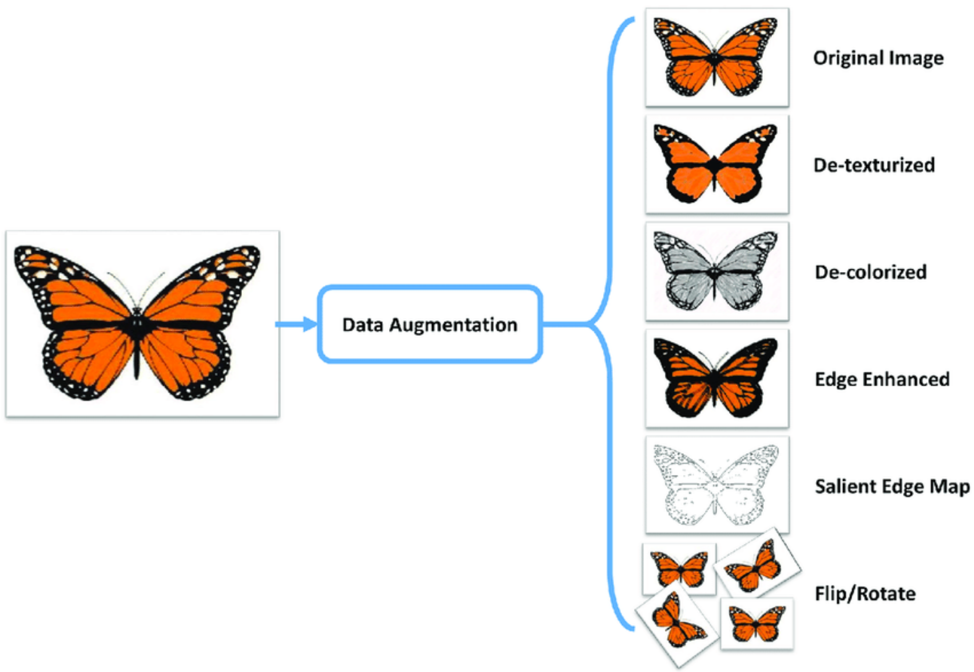
$$Loss = \underset{Supervised}{L_S} + \underset{Unsupervised}{L_U}$$

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

Background

❖ Consistency Regularization

- Data Augmentation
  - ➢ Supervised: 데이터에 약간의 변형을 가하더라도 클래스 정보는 영향을 받지 않을 것
  - ➢ Semi-Supervised: Label이 없는 데이터에 Augmentation을 하면 클래스 예측 분포가 달라짐

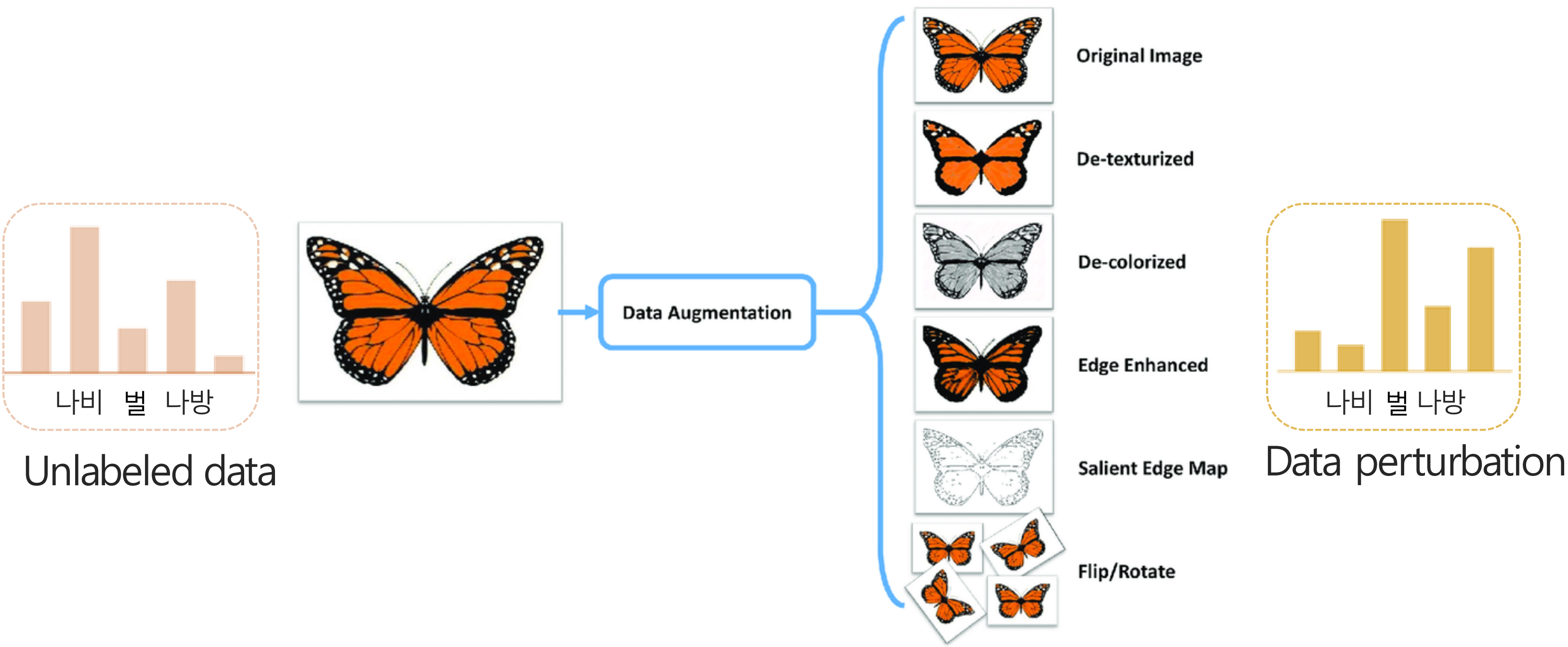


Label: 나비 Label: 나비

Background

❖ Consistency Regularization

- Data Augmentation
  - ➢ Supervised: 데이터에 약간의 변형을 가하더라도 클래스 정보는 영향을 받지 않을 것
  - ➢ Semi-Supervised: Label이 없는 데이터에 Augmentation을 하면 클래스 예측 분포가 달라짐



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

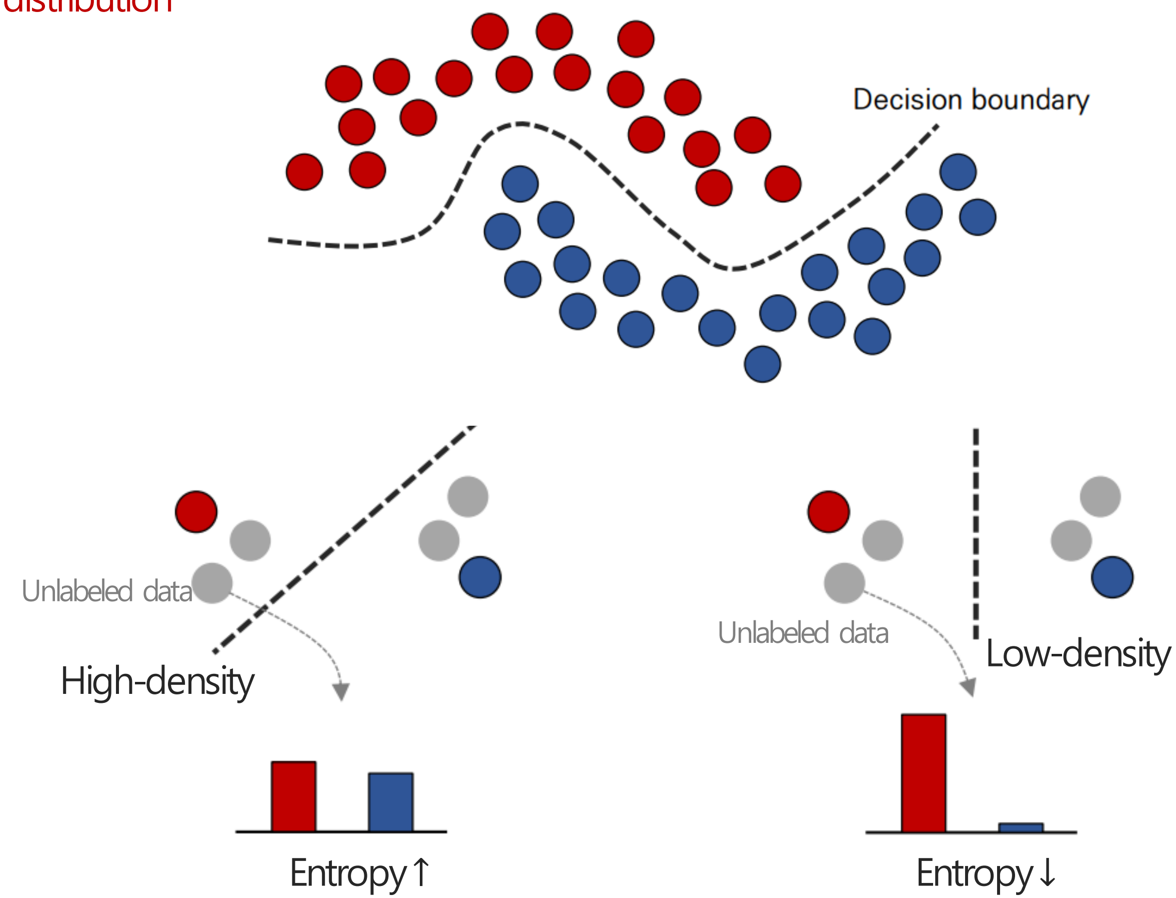
목표: Unlabeled data에 Augmentation을 수행해도 동일한 클래스 분포를 예측하도록 학습

DMQA

Background

❖ Entropy Minimization

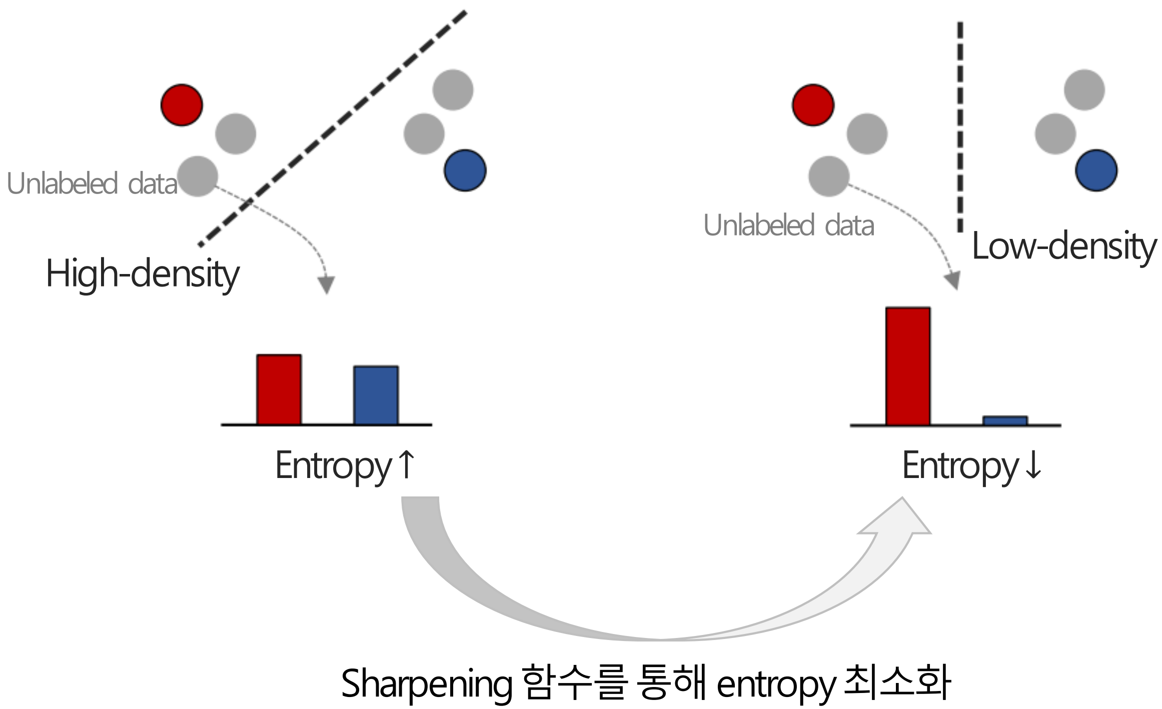
- SSL assumption:
  - ➢ Classifier's decision boundary should not pass through high-density regions of the marginal data distribution



Background

❖ Entropy Minimization

- SSL assumption:
  - ➢ Classifier's decision boundary should not pass through high-density regions of the marginal data distribution

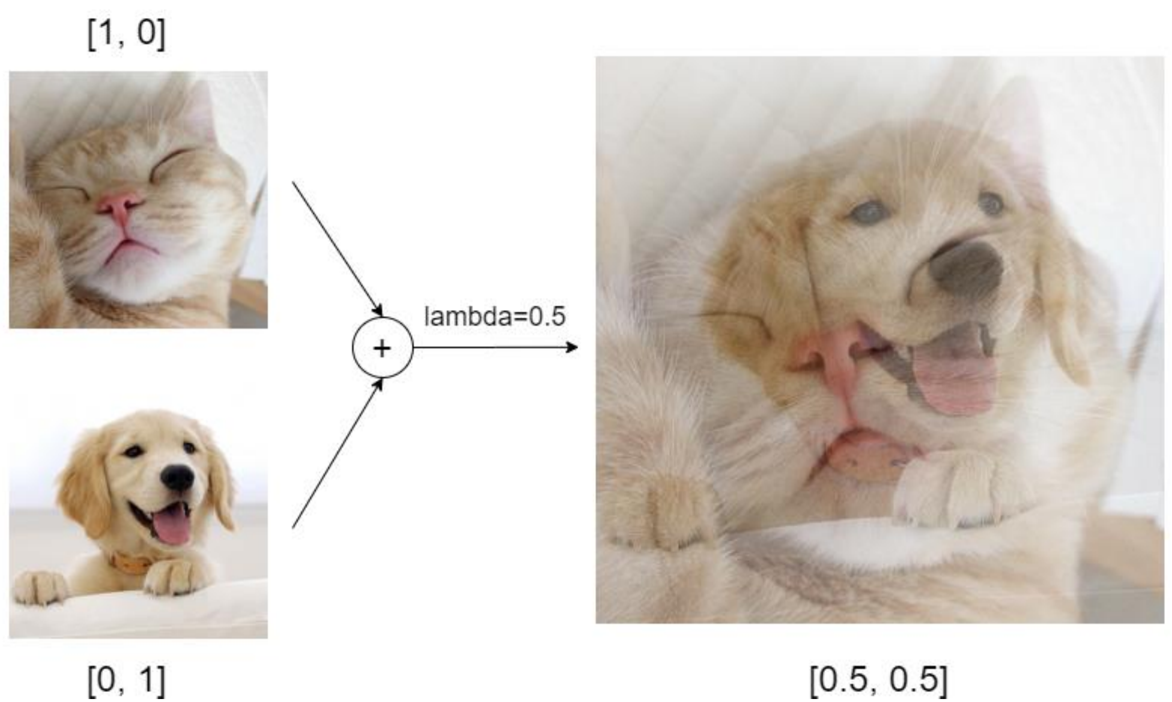


목표: Unlabeled data에 예측 값의 confidence를 높이도록 학습

Background

❖ Traditional Regularization(Mixup)

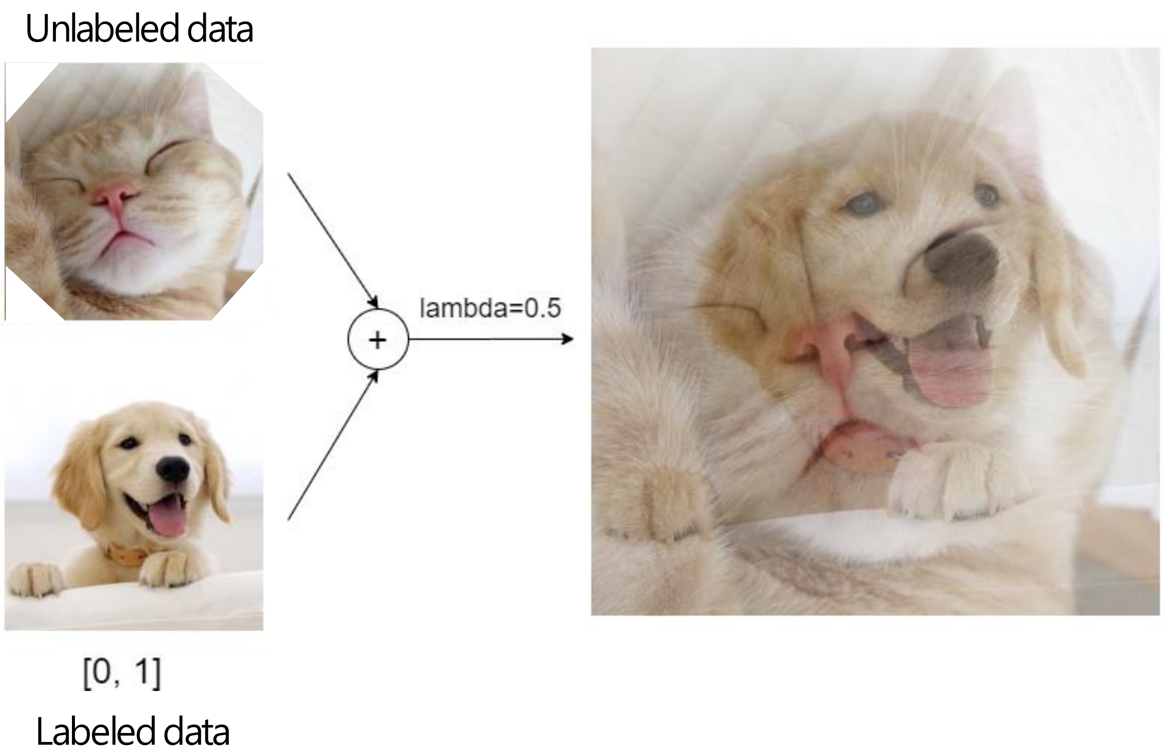
- Supervised: 데이터와 label 각각을 interpolation하여 새로운 데이터 생성
  - ➢ Overfitting 방지하여 일반화 성능 향상



Background

❖ Traditional Regularization(Mixup)

- Unsupervised: 모델이 unlabeled 데이터에 대한 생성한 가짜 label 사용

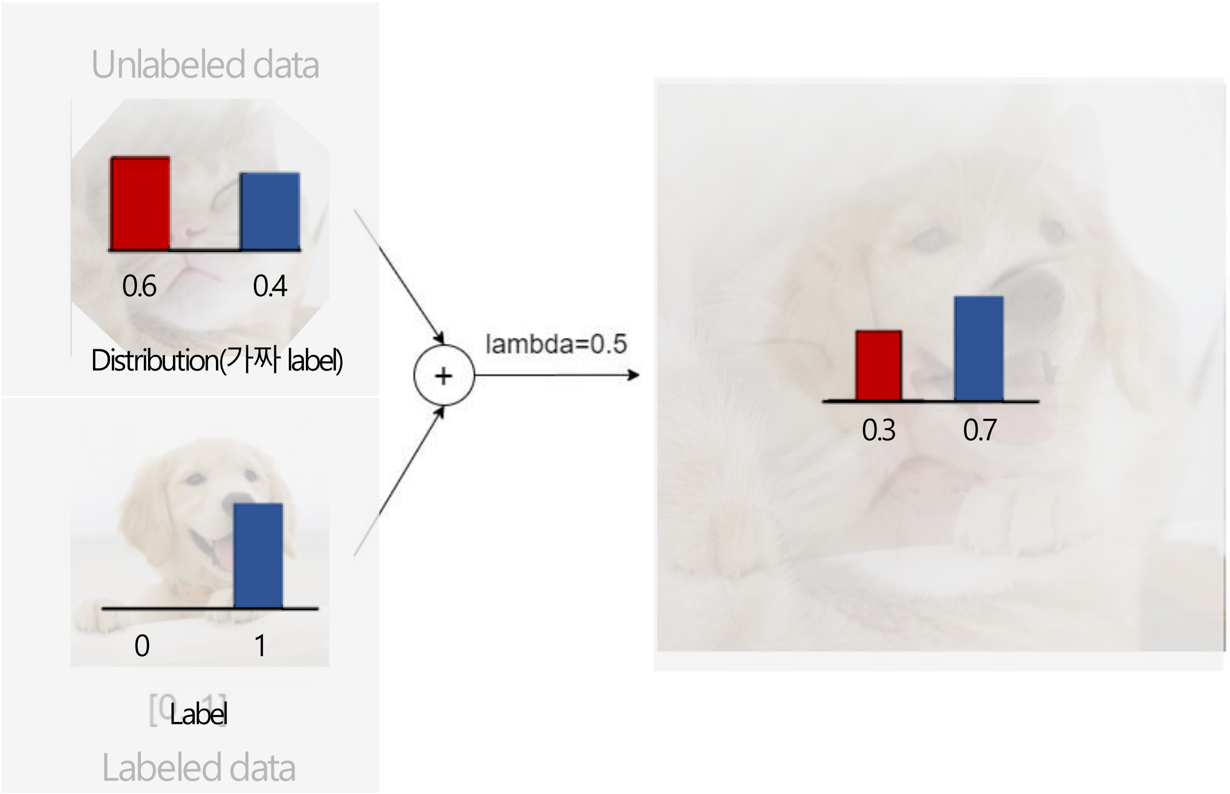


DMQA

Background

❖ Traditional Regularization (Mixup)

- Unsupervised: 모델이 unlabeled 데이터에 대한 생성한 가짜 label 사용



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 X = ((x_b, p_b); b ∈ (1, ..., B)), batch of
    unlabeled examples U = (u_b; b ∈ (1, ..., B)), sharpening temperature T, number of augmentations K,
    Beta distribution parameter α for MixUp.
for b = 1 to B do
    x̂_b = Augment(x_b)    // Apply data augmentation to x_b
    for k = 1 to K do
        û_{b,k} = Augment(u_b)    // Apply k^th round of data augmentation to u_b
    end for
    q̄_b = (1/K) Σ_k p_model(y | û_{b,k}; θ)    // Compute average predictions across all augmentations of u_b
    q_b = Sharpen(q̄_b, T)    // Apply temperature sharpening to the average prediction (see eq. (7))
end for
X̂ = ((x̂_b, p_b); b ∈ (1, ..., B))    // Augmented labeled examples and their labels
Û = ((û_{b,k}, q_b); b ∈ (1, ..., B), k ∈ (1, ..., K))    // Augmented unlabeled examples, guessed labels
W = Shuffle(Concat(X̂, Û))    // Combine and shuffle labeled and unlabeled data
X' = (MixUp(X̂_i, W_i); i ∈ (1, ..., |X̂|))    // Apply MixUp to labeled data and entries from W
U' = (MixUp(Û_i, W_{i+|X̂|}); i ∈ (1, ..., |Û|))    // Apply MixUp to unlabeled data and the rest of W
return X', U'
```

MixMatch

❖ Framework

- 입력 데이터는 mini batch마다의 labeled data χ, Unlabeled data $\mathcal{U}$ (1)
- Stochastic Data Augmentation: (3-6)
  - ➢ 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: $\quad \bar{q}_b = \frac{1}{K} \sum_k \mathrm{p}_{\text{model}}(y \mid \hat{u}_{b,k}; \theta)$ *// 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

- 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: $\quad \bar{q}_b = \frac{1}{K} \sum_k \text{p}_{\text{model}}(y \mid \hat{u}_{b,k}; \theta)$ *// 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**

Label guessing

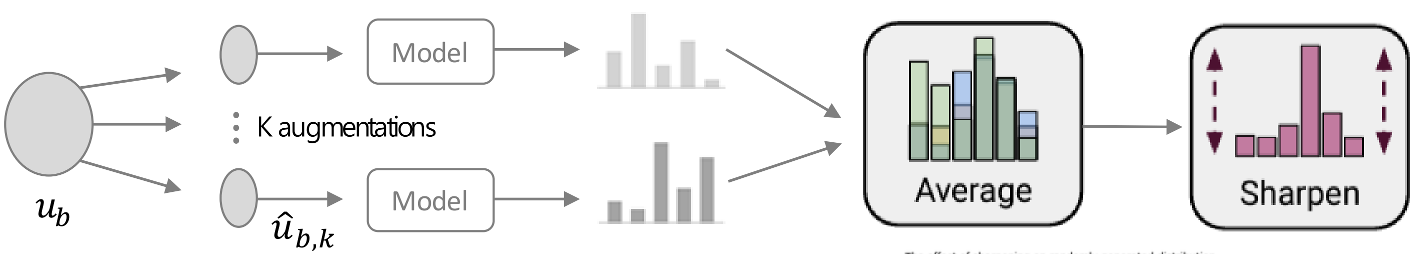
DMQA

MixMatch

❖ Framework

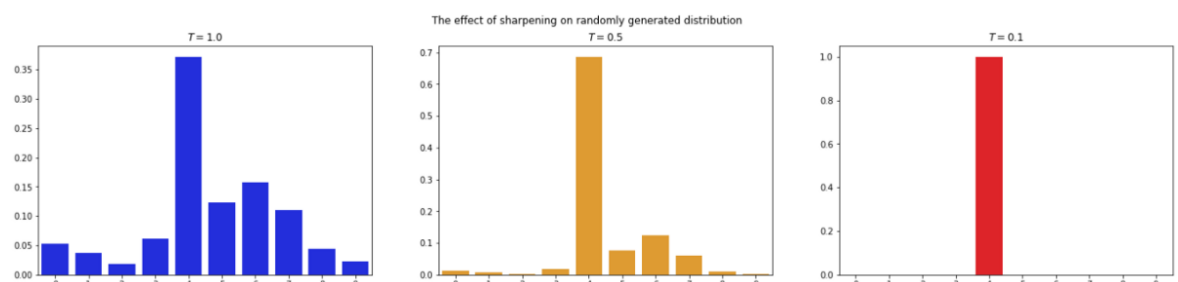
- Sharpening: (8)
  - ➢ Softmax Temperature를 이용한 Entropy Minimization

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: $\quad \bar{q}_b = \frac{1}{K} \sum_k \mathrm{p}_{\text{model}}(y \mid \hat{u}_{b,k}; \theta)$ *// 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**



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

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



MixMatch

❖ Framework

- 앞서 labeled data와 unlabeled data에 augmentation을 통해 얻은 데이터와 분포(p,q)를 각각 $\hat{\chi},\hat{\mathcal{U}}$ 정의하고(10, 11), 이를 합친 후, 섞어 W 생성(12)
- MixUp: (13-15)
  - ➢ $\hat{\chi},\hat{\mathcal{U}}$ 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** $\mathcal{X}',\mathcal{U}'$

MixMatch

❖ Framework

- **Loss function:**

$$\mathcal{X}', \mathcal{U}' = \text{MixMatch}(\mathcal{X}, \mathcal{U}, T, K, \alpha)$$

Minibatch마다 labeled data χ, Unlabeled data $\mathcal{U}$ 에 MixMatch 를 적용하여 χ',$\mathcal{U}'$생성

Supervised Loss $$\mathcal{L}_{\mathcal{X}} = \frac{1}{|\mathcal{X}'|} \sum_{x,p \in \mathcal{X}'} \underbrace{\mathrm{H}(p, \mathrm{p}_{\text{model}}(y \mid x; \theta))}_{\text{CrossEntropy}}$$

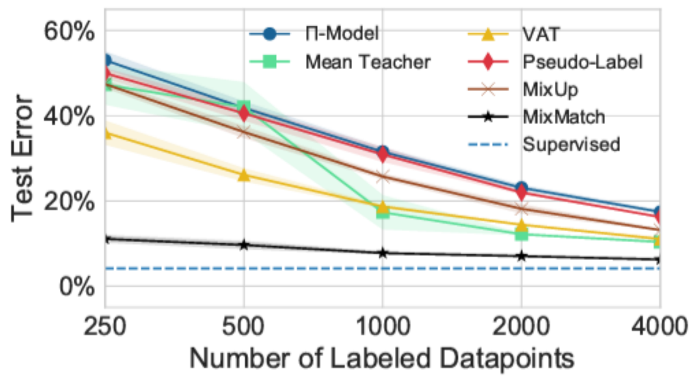
Consistency Loss $$\mathcal{L}_{\mathcal{U}} = \frac{1}{L|\mathcal{U}'|} \sum_{u,q \in \mathcal{U}'} \underbrace{\|q - \mathrm{p}_{\text{model}}(y \mid u; \theta)\|_2^2}_{\text{L2 loss(Mean Squared Error)}}$$

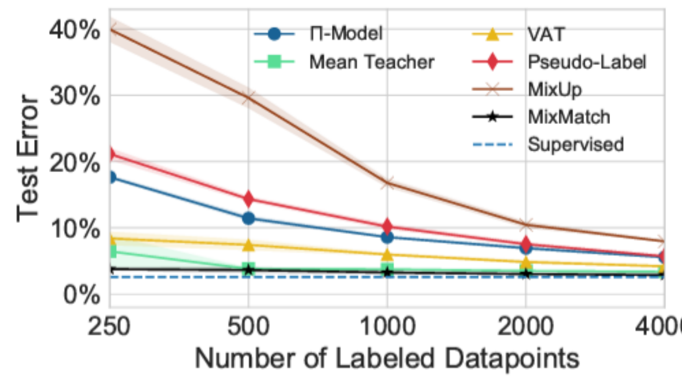
Total Loss $$\mathcal{L} = \mathcal{L}_{\mathcal{X}} + \lambda_{\mathcal{U}} \mathcal{L}_{\mathcal{U}}$$

Experiments

❖ Results

- Baseline 모델과 MixMatch 모두 Wide ResNet-28 을 사용
- Labeled data의 수를 점점 늘려가며 모델 성능 평가 및 비교
- SSL 비교 방법론 대비 월등하며, 지도학습 성능과 유사할 만큼 우수함
  - ➢ Supervised learning training: CIFAR-10에서 50,000개, SVHM에서 73,257개를 학습
  - ➢ Supervised learning testing error: 4.13%(CIFAR-10), 2.59%(SVHN)





Methods/Labels	250	500	1000	2000	4000
PiModel	53.02 ± 2.05	41.82 ± 1.52	31.53 ± 0.98	23.07 ± 0.66	17.41 ± 0.37
PseudoLabel	49.98 ± 1.17	40.55 ± 1.70	30.91 ± 1.73	21.96 ± 0.42	16.21 ± 0.11
Mixup	47.43 ± 0.92	36.17 ± 1.36	25.72 ± 0.66	18.14 ± 1.06	13.15 ± 0.20
VAT	36.03 ± 2.82	26.11 ± 1.52	18.68 ± 0.40	14.40 ± 0.15	11.05 ± 0.31
MeanTeacher	47.32 ± 4.71	42.01 ± 5.86	17.32 ± 4.00	12.17 ± 0.22	10.36 ± 0.25
MixMatch	11.08 ± 0.87	9.65 ± 0.94	7.75 ± 0.32	7.03 ± 0.15	6.24 ± 0.06

Table 5: Error rate (%) for CIFAR10.

250	500	1000	2000	4000
17.65 ± 0.27	11.44 ± 0.39	8.60 ± 0.18	6.94 ± 0.27	5.57 ± 0.14
21.16 ± 0.88	14.35 ± 0.37	10.19 ± 0.41	7.54 ± 0.27	5.71 ± 0.07
39.97 ± 1.89	29.62 ± 1.54	16.79 ± 0.63	10.47 ± 0.48	7.96 ± 0.14
8.41 ± 1.01	7.44 ± 0.79	5.98 ± 0.21	4.85 ± 0.23	4.20 ± 0.15
6.45 ± 2.43	3.82 ± 0.17	3.75 ± 0.10	3.51 ± 0.09	3.39 ± 0.11
3.78 ± 0.26	3.64 ± 0.46	3.27 ± 0.31	3.04 ± 0.13	2.89 ± 0.06

Table 6: Error rate (%) for SVHN.

Experiments

❖ Ablation study

- 모델 내 조건을 변경하며, 실험 진행
- 각 구성 요소들이 모두 성능을 향상시키는데, 필요함을 입증

Ablation	250 labels	4000 labels
MixMatch	11.80	6.00
MixMatch without distribution averaging ($K = 1$)	17.09	8.06
MixMatch with $K = 3$	11.55	6.23
MixMatch with $K = 4$	12.45	5.88
MixMatch without temperature sharpening ($T = 1$)	27.83	10.59
MixMatch with parameter EMA	11.86	6.47
MixMatch without MixUp	39.11	10.97
MixMatch with MixUp on labeled only	32.16	9.22
MixMatch with MixUp on unlabeled only	12.35	6.83
MixMatch with MixUp on separate labeled and unlabeled	12.26	6.50
Interpolation Consistency Training [45]	38.60	6.81

Table 4: Ablation study results. All values are error rates on CIFAR-10 with 250 or 4000 labels.

DMQA

Conclusion

❖ Conclusion

- "Holistic" approach which incorporates ideas and components from the dominant paradigms for SSL (기존 SSL 방법론 총망라)
- 뛰어난 성능에도 불구하고, 비교적 많은 hyperparmeter들을 조작해야하는 단점 존재

❖ Reference

- Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., & Raffel, C. A. (2019). Mixmatch: A holistic approach to semi-supervised learning. Advances in neural information processing systems, 32.
- http://dmqm.korea.ac.kr/activity/seminar/303
- http://dsba.korea.ac.kr/seminar/?mod=document&uid=68

Thank You