**Ensemble pseudo labeling with Coreset**

GSDS 석사재학 20224314

강현구

1. **Proposal**

최근Deep learning(이하 DL)모델은 많은 양의 Labeled Data를 기반으로 다양한 분야에서 높은 성능을 이끌어 냈다. 하지만 모델이 필요로 하는 데이터가 증가함에 따라, Labeling에 필요한 Human resource 또한 비례하여 Labeling Cost가 많이 든다는 문제점이 있다. 이와 관련하여 Labeling 했을 때 모델 학습에 효과적인 Unlabeled Data를 Sampling을 하는 Deep Active learning(이하 DAL)은 Labeling cost를 줄이는데 효과적이다. 동일한 맥락에서 주어진 Label Data를 최대한 활용하는 Semi-supervised learning(이하 SSL) 또한 중요성이 강조된다.

DAL의 acquisition Strategy와 SSL의 Data expansion은 상호적으로 긍정적인 영향을 준다. DAL의 acquisition Strategy는 학습 모델의 성능에 의존한다. 따라서 Data expansion을 통한 모델 성능 향상은 DAL의 성능 또한 향상시킨다. 한편으로 SSL 학습 과정에서 DAL의 acquisition Strategy은 대표성 또는 정보 가치가 높은 데이터의 Labeling을 가능하게 한다. 이는 Data expansion 간 정확도 높은 추측을 가능하게 만든다. 하지만 각각의 방법에 대한 독립적인 연구는 많이 되었으나, 각 방법 간의 연관성에 대한 연구는 부족하다. 위 접근법을 취한 연구로는 Simeoni의 연구(Siméoni, Budnik, Avrithis, & Gravier, 2021)에서 pseudo labeling 및 pre-training 방법과 다양한 Batch Sampling 방식을 적용하며 관련성을 확인했다. 하지만 Simeoni의 연구는 각 방법을 Orthogonal 하게 사용하여 DAL과 SSL의 연관성을 활용하지 못했다는 한계점이 있다.

이 연구에서는 DAL의 한 방법인 coreset Selection을 활용한 새로운 Pseudo labelling 방법을 제안한다. Coreset Selection은 전체 데이터를 덮을 수 있는 주어진 Batch 개수 점을 중심으로 subgraph들을 구성할 때, 최소한의 반지름(threshold)를 가지게 만드는 점들을 Sampling 하는 방법을 말한다. 동일한 클래스 간의 거리는 가까울 것이라는 가정 하에, Coreset selection 간 형성한 subgraph 속에 있는 데이터들은 중점과 유사한 클래스일 확률이 높을 것이다. 이를 활용하여 신뢰성을 보장할 수 있는 unlabeled data에 한해 Pseudo labeling을 부여할 것이다. 본 방법론은 Coreset selection의 결과물을 pseudo labeling에서 활용한다는 점에서, 별개로 각 방법을 활용하는 것보다 성능 향상 및 계산량이 적을 것이다. 또한 클래스의 수가 적고 데이터 분포가 밀접해있는 데이터셋에 대해선 높은 효율 및 정확도의 Pseudo labeling을 할 수 있을 것이라 기대한다.

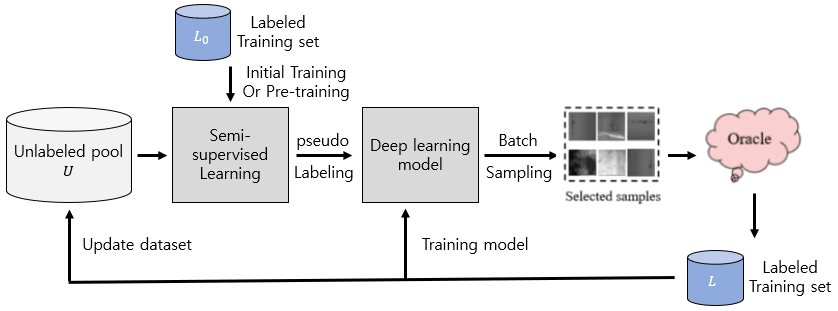
**1. Task Description**

**1-1) Data set : MNIST, SVHN, CIFAR-10, CIFAR-10**

**1-2) Notation**

* n examples where := [n] := {1, …, n}
* Given a set X :=
* labels for training
* U : Unlabeled pool Dataset
* L : Labeled pool Dataset
* of labels , a set if indices with
* Batch

**1-3) Naiive Model**



Phase 1 : Random sampling Labeled Training set

Phase 2 : Pre-training

Phase 3 : Batch sampling with acquisition strategy of Deep Active learning

Phase 4 : Get new labeled data from Oracle and update dataset

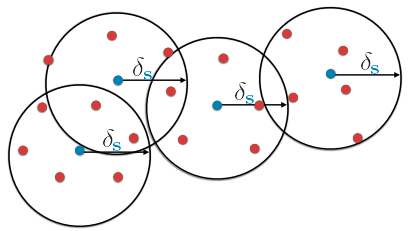
Phase 5 : Generate Pseudo labeling by SSL

Phase 6 : finetune Deep learning model

Repeat phase 3 ~ 6 until 목표 성능까지 도달하기.

**2. Idea Sketch**

2-1). Coreset Selection



Coreset Selection은 전체 데이터를 덮을 수 있는 주어진 Batch 개수 점을 중심으로 subgraph들을 구성할 때, 최소한의 반지름(threshold)를 가지게 만드는 점들을 Sampling 하는 방법을 말한다{Sener, 2017 #4}. 는 Sampling 된 데이터로 각 i번째 subgraph의 중점을 의미한다. 는 j번째 data가 i번째 subgraph에 속했는가를 의미한다. 는 각 subgraph들의 반지름을 의미한다. Coreset은 모든 데이터들을 Subgraph으로 덮을 때 을 최소화하는 방법이다. 즉, 최소한의 Sampling 개수로 전체 데이터 셋에서의 성능 차이를 최소화하는 데이터를 Sampling 하는 방식이다. 또한 각각의 subgraph의 중점은 전체 데이터 셋을 전체 데이터를 최소 이상 간격을 두고 데이터를 뽑은 것과 동일하다. 즉, 각 중점들은 비교적 데이터의 밀집 정도에 무관하게 넓게 퍼져 있음을 알 수 있다. 이로 인해 Sampling point 는 특정 클래스의 값 만을 포함하는 것이 아닌 전반적인 클래스의 경우를 고려하게 된다. 한편으로 의 Labelling 분포가 각 클래스별 데이터의 밀도가 아닌 넓이에 기반한다고 할 수 있다.

**2-2) Pseudo labeling**

동일한 클래스를 가진 데이터들은 거리가 가까울 것이란 가정 하에 Pseudo labelling 부여 방식은 2가지를 고려한다.

첫번째는 의 Label이 가 속한 subgraph와 접하는 subgraph들의 중점 Label과 모두 동일할 때, subgraph에 속하는 점들에 대해서 와 동일한 pseudo label을 부여한다. 거리 가정으로 인해 동일한 클래스를 가진 데이터들은 서로 밀집해 있을 것이다. 따라서 Subgraph 안에 있는 데이터들을 중점과 동일한 Label을 가질 확률이 높다. 하지만 주변 subgraph의 중점 Label과 다를 경우, 해당 subgraph는 서로 다른 클래스의 경계에 속해 있을 확률이 크다. 반대로 접하고 있는 subgraph의 중점 Label이 동일할 경우, 두 subgraph의 사이에 있는 데이터는 높은 확률로 동일한 Label을 가질 것이다. 이에 접하고 있는 모든 subgraph 중점의 클래스와 동일한 클래스를 가진 subgraph의 데이터에 한해 pseudo labeling을 부여한다. 이 방식의 장점은 라벨 클래스의 수가 적거나 Unlabel data가 많아 특정 클래스의 분포가 명확히 구분할 때 부각된다. Subgraph의 넓이 개념을 기반으로 각 클래스의 주변부와 중심부를 구분하며, 중심부로 판단되는 subgraph에 대해서 pseudo labeling을 다 적용해준다. 즉, 적절한 , 데이터의 클래스별 밀집되어 있음을 보장할 수 있다면 전체 데이터에서 각 클래스별 중심부에 해당하는 모든 데이터를 한번에 Pseudo labeling을 부여할 수 있다. 단, Confirmation bias 문제를 해소하지 않는다면 본 방식은 효율적이나 부정확한 결과를 낼 수 있다. 따라서 이후 기존의 SSL 연구에서 Confirmation bias를 해소할 방법들을 같이 적용할 것이다. 지금은 해당 방법과 관련된 Confirmation bias 해소 방안만을 언급하겠다. Pseudo labeling의 Confirmation bias를 피하기 위해 기존에 부여한 pseudo labelled data가 이후 coreset selection 간 다른 클래스를 가진 subgraph에 한 번이라도 속할 경우 label을 제거한다.

두번째로 각 Coreset selection을 진행할 때 마다 각 데이터가 속한 subgraph의 중점의 클래스 개수와 각 Subgraph의 반지름 을 기반으로 각 클래스에 속할 확률을 계산한다. Subgraph의 중점 과 첫번째 방식으로 pseudo label을 부여하지 않은 모든 데이터에 대해 적용한다. 각 데이터 별로 지금껏 속했던 subgraph들의 중점의 클래스의 경우의 수를 고려하여 확률 값을 부여하다. 예로 들어 한 데이터가 4번의 Coreset selection 간 3번을 중점의 클래스가 a인 subgraph에 속하고, 1번을 b 클래스가 중점인 subgraph에 속했다고 하자. 그럼 해당 데이터는 a에 속할 확률이 가장 크며, 그 다음은 b에 속할 확률이 클 것이다. 또한 한번의 Coreset selection 간 다수의 Subgraph의 속할 경우 각각의 경우도 같이 고려해준다. 또한 각 subgraph의 반지름인 에 반비례하여 subgraph안의 데이터들은 중점 와 동일한 클래스일 확률이 높을 것이다. 따라서 각 데이터들을 subgraph에 속하는 횟수와 반지름을 기반으로 softmax를 통해 수치화할 것이다.

또한 가장 높은 확률 값으로 고정할 경우 Confirmation bias가 발생할 수 있다. 따라서 Confirmation bias를 방지하기 위해 SSL에서 자주 사용하는 Mix up data agumentation을 적용하여 Stochastic하게 pseudo labeling을 부여한다. 해당 방법은 충분한 횟수의 subgraph의 중첩이 생기기까지 잘못된 분류를 할 확률이 높다. 따라서 허용 오차율 을 적용하여 특정 class일 확률이 1-보다 클 경우만 pseudo label을 부여한다.

3. 예상 문제점

3-1). 적절한 선정

Coreset subgraph의 기반 Pseudo labeling에서 가장 중요한 것은 적절한 subgraph의 반지름 을 선정하는 것이다. Coreset 과정에 따라 은 Sampling size b(= ||)에 달려 있다. 즉, b의 크기가 작을 땐 Labelling Cost는 줄어들지만 이 커져 유의미한 pseudo labeling을 적용하기 어렵다. 반대로 b의 크기가 크다면 이 줄고, subgraph의 밀도가 높아져 pseudo labeling을 높은 정확도로 부여할 수 있다. 따라서 Sampling 개수와 pseudo labeling 간의 Trade off 관계가 형성된다. 기존의 연구 방안에선 b를 임의로 부여하는 값으로, 어떠한 가이드라인도 주고 있지 않다. 따라서 적정한 을 선정한 가이드라인을 제안해야 한다.

본 연구에서는 subgraph의 데이터 밀도를 통해 가이드라인을 제안하고자 한다. Pseudo labeling 간 가장 피해야할 문제는 confirmation bias이다. 따라서 Labelling cost를 많이 들여 생기는 문제보다, 가 커서 pseudo label을 잘못 부여하는 것을 피하는 데 중점을 잡겠다. 따라서 데이터셋에서 데이터 밀도가 가장 높은 곳을 기준으로, 1개의 subgraph에 속하는 일정 이상의 데이터의 수(ex- M)를 제한하는 값을 찾는다. Subgraph의 밀도는 를 통해서 구할 수 있다. 가 j번째 data가 i번째 subgraph에 속했는가를 의미하기 때문에, i번째 row에 대한 값을 다 더하면 i번째 subgraph에 속하는 데이터의 개수를 구할 수 있다. 즉, 임의로 Coreset selection을 진행하여 밀도가 가장 높은 subgraph를 찾는다. 이후 해당 subgraph에 속한 데이터에 한하여 subgraph당 데이터의 수가 M개 아래로 가게 하는 를 찾는다.

단, 추가적으로 해결해야할 문제가 있다. 값을 찾았다고 해서 최적의 Sampling 개수 b를 구할 수 있는 것이 아니다. 즉, 주어진 b에 대해 을 최적화하는 것은 Coreset에서 제안한 K-center-greedy를 통해 구할 수 있으나, 반대 방향의 최적화할 방안이 없다. 이에 대한 추가 방안을 제안해야 좀 더 유의미한 Solution이 될 수 있다.

문제점 1 : 주변의 subgraph의 접했는가와 동일한 Label인가를 서로 겹치는 점을 통해서 판단했다. 그러면 그런 겹치는 점이 하나도 없다면? 그 subgraph에 대해서도 동일하게 다 Labeling을 할 것인가?

문제점 2 : 다른 클래스의 데이터가 차지하는 넓이가 delta 보다 작아서 한 subgraph안에 속한다면? 이건 전부 잘못 pseudo labeling 되는 건데? 이 방법은 Minor Class, Class imbalance에 취약한 방법인듯.

이 경우에 대한 해답이 delta를 줄인다 말고 있나?

3-2). Confirmation bias

SSL 방식의 Pseudo labeling의 주요 문제점 중 하나는 Confirmation bias이다. 이전의 Pseudo labeling 연구에서 Mixup data agumentation, soft labeling, reinitiate model parameter을 통해서 일부 해결함을 보였다. 본 연구에서도 나와있는 방법들을 활용 및 Ablation study를 통해서 confirmation bias를 해결하는 지 확인하겠다.

3-3). Distance - based approach

본 연구는 Coreset selection 간 Distance에 기반을 두고 있다. Distance 계산이 많아지는 고차원의 큰 데이터셋 일 경우 계산양이 크게 늘어난다. 이에 Dimension reduction technique를 적용한다. 차원 축소 방법으로 PCA, LDA, LLE, t-SNE과 deep learning 기반의 autoencoder 및 Variational autoencoder가 있다. Autoencoder과 Variational autoencoder는 현장에서 많은 성과를 내고 있다. 또는 CNN의 도입을 통해서 representation을 추출하는 방법이 있다. 본 연구에선 CNN, conditional Autoencoder과 t-SNE 도입에 초점을 맞출 것이다. CNN은 특히 이미지 데이터에 대해서 공간 정보를 유지한 채 차원을 축소하는데 유용하다. Conditional autoencoder은 주어진 label data를 활용할 수 있으며, 효과적으로 data의 representation을 학습할 수 있다. 한가지 점검해야 하는 점으로 차원 축소를 했을 때 클래스 별 거리 관계가 유지되는지 확인해야 한다. Hidden variable 들은 decode, encode에 최적화되어 있어 클래스별 거리 관계를 고려하지 않았을 수 있다. 다른 한편으론 클래스별 특징을 잘 반영했기 때문에 거리 관계 또한 재현되어 있을 수 있다. 또는 t-SNE를 적용하는 것도 방법이다. t-SNE의 경우 차원 축소 뿐만 아니라 각 클래스별로 데이터를 더욱 밀집시키는 특징이 있다. 하지만 계산량이 O()으로 큰 데이터셋에 적용하기 어렵다는 제한이 있다. 실제 구현에는 Barnes hut t-SNE 라는 방법을 통해 더 효율적으로 계산할 수 있어 현장에 충분히 적용할 만인 것인지 확인하겠다.

1. 클래스별 데이터가 안 모여 있을 때. How?

또한 동일한 클래스의 데이터 간 거리가 가깝지 않을 경우 subgraph을 통한 pseudo labeling을 적용할 수 없다. 이에 데이터를 저차원으로 줄이기 위해 CNN, AE 등의 representation learning을 통해서 저차원의 데이터로 축소할 것이다. 이 과정에서 representation 들은 동일 클래스 사이에선 유사하게, 서로 다른 클래스 사이 간에는 다르게 설정될 것이다. 여기에 더불어 t-SNE 와 같이 동일한 클래스끼리 데이터 분포를 좀 더 밀집시키는 방안을 탐색할 것이다.

4. 예상 기대효과

본 방식은 Subgraph을 통해 각 클래스의 주변부와 중심부를 구분하며, 중심부에 대해서 한번에 pseudo labeling을 부여할 수 있다. 즉, unlabeled 데이터가 많으면서 클래스의 수가 적은 데이터셋과 같이 클래스 별로 밀도가 높은 데이터에 대해 높은 효율의 pseudo labeling이 가능하다. 또한 다음 coreset selection 간에는 pseudo labeling이 적용되지 않은 데이터에 한정하여 적은 Sampling data로 작은 의 subgraph을 부여할 수 있다. 이를 통해 적은 횟수로 정확하며 빠르pseudo labeling이 가능하다.

**<참고문헌>**

Liu, P., Zhang, H., & Eom, K. B. (2016). Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10*(2), 712-724.

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B. B., . . . Wang, X. (2021). A survey of deep active learning. *ACM computing surveys (CSUR), 54*(9), 1-40.

Sener, O., & Savarese, S. (2017). Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*.

Siméoni, O., Budnik, M., Avrithis, Y., & Gravier, G. (2021). *Rethinking deep active learning: Using unlabeled data at model training.* Paper presented at the 2020 25th International Conference on Pattern Recognition (ICPR).

Wang, K., Zhang, D., Li, Y., Zhang, R., & Lin, L. (2016). Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology, 27*(12), 2591-2600.

Zhu, J.-J., & Bento, J. (2017). Generative adversarial active learning. *arXiv preprint arXiv:1702.07956*.