Active Learning in Unseen Domain Text Classification

Xiang Pan, Joon Kim

New York University

Executive Summary

- Pretrained Language Models does not satisfy self clustering on unseen domain and thus get low accuracy in low resource setting.
- We used same task, same domain adaptive pretraining and active learning to boost the performance of low-resource learning in unseen domains
- We experimented with Covid QCLS dataset
- We hope our work can reduce manual annotation done by human labor

Motivation

- Labeling is expensive and highly human-relied.
- BERT embedding is self-clustered for general domain text classification tasks, (e.g., sentiment analysis or topic classification)
    - It does not need many annotations to get high accuracy.
- But for a new domain (e.g., COVID-19), the Pretrained Language Models' embedding may not satisfy the good self-clustered property
- **Goal**: Make classification models work well in unseen domains with less annotation as possible.

Related Work

- Gururangan et al showed that in domain pretraining leads to performance gain under both high and low resource setting
- Liat et al have shown that Active Learning (AL) can boost the BERT performance in binary classification tasks. However, the tasks are general topics such as sentiment classification and topic classification.
- Yue et al have shown that BERT embedding and attention own extent self-clustering property

Technical Challenge

- The general Pretrained Language Models (PLMs) is not pretrained on the unseen domain corpus
- For unseen domains, the BERT self clustering property can not hold, making the low-resource setting difficult.
- How to select representative and valuable samples to label in unseen domain is the key problem.

Technical Challenge

- We tested our experiment with CovidQCLS dataset
  - Question category classification dataset consisting of 15 different categories
  - ex) Transmission, Prevention
  - 1120 train data and 125 test data
  - Example Questions:
    - For whom does covid pose the greatest threat? (Transmission)
    - Does CDC recommend the use of facemask or face coverings to prevent covid? (Prevention)
  - Need to ensure that model understands these new domain vocabularies.

Approach - PLMs

We explore various PLMS trained with various settings and fine tune on top of them

- Baseline: Roberta Base
- Same Task: Roberta Base SQuAD2
- Same Domain: Bertweet Covid 19 Base Cased
- Same Task, Same Domain: Roberta Base SQuAD2 Covid

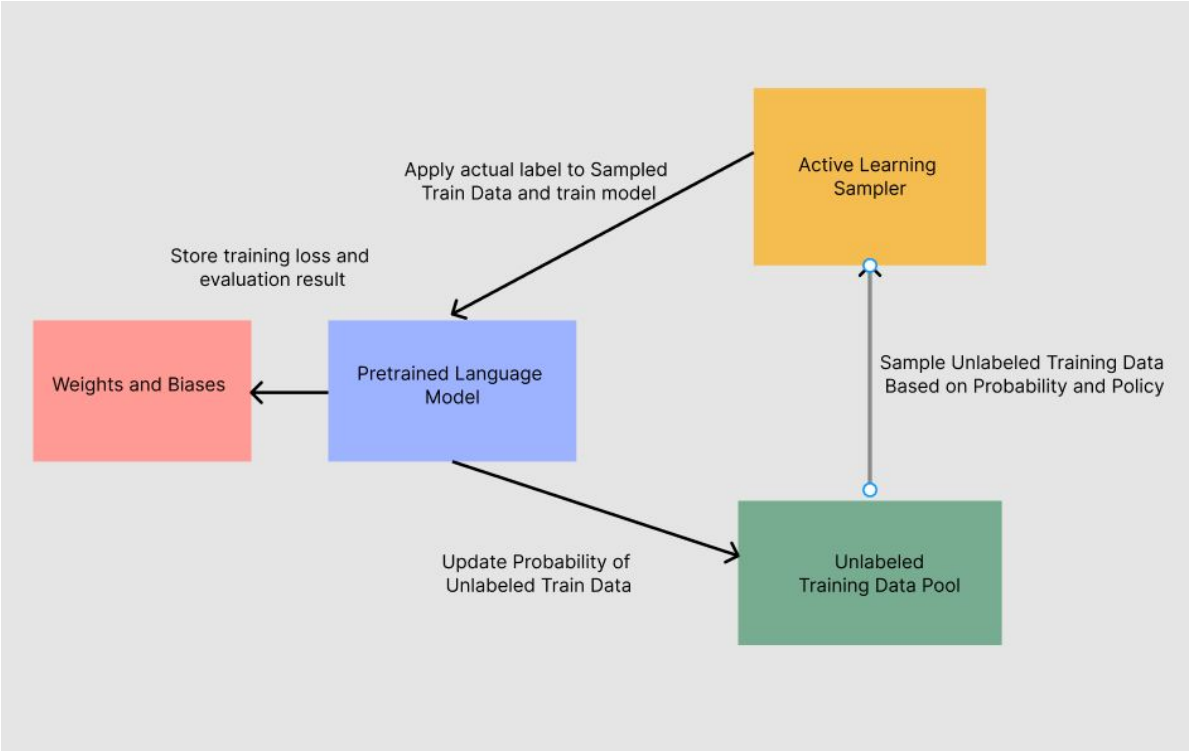
Approach - Active Learning

We use active learning with various sampling methods on the PLMs to boost the performance of low-resource learning

Sampling Methods

- Random Sampling - Baseline
- Entropy Based Method
  - Decreasing order of $\sum_{\hat{y} \in Y} \hat{y} \cdot \log(\hat{y})$
- Least Confidence
  - Increasing order of $\max_{\hat{y} \in Y}(\hat{y})$
- Margin Based Method
  - Increasing order of $(\hat{y_1} - \hat{y_2})$ where y_1 is highest probability class and y_2 is second highest probability of a class

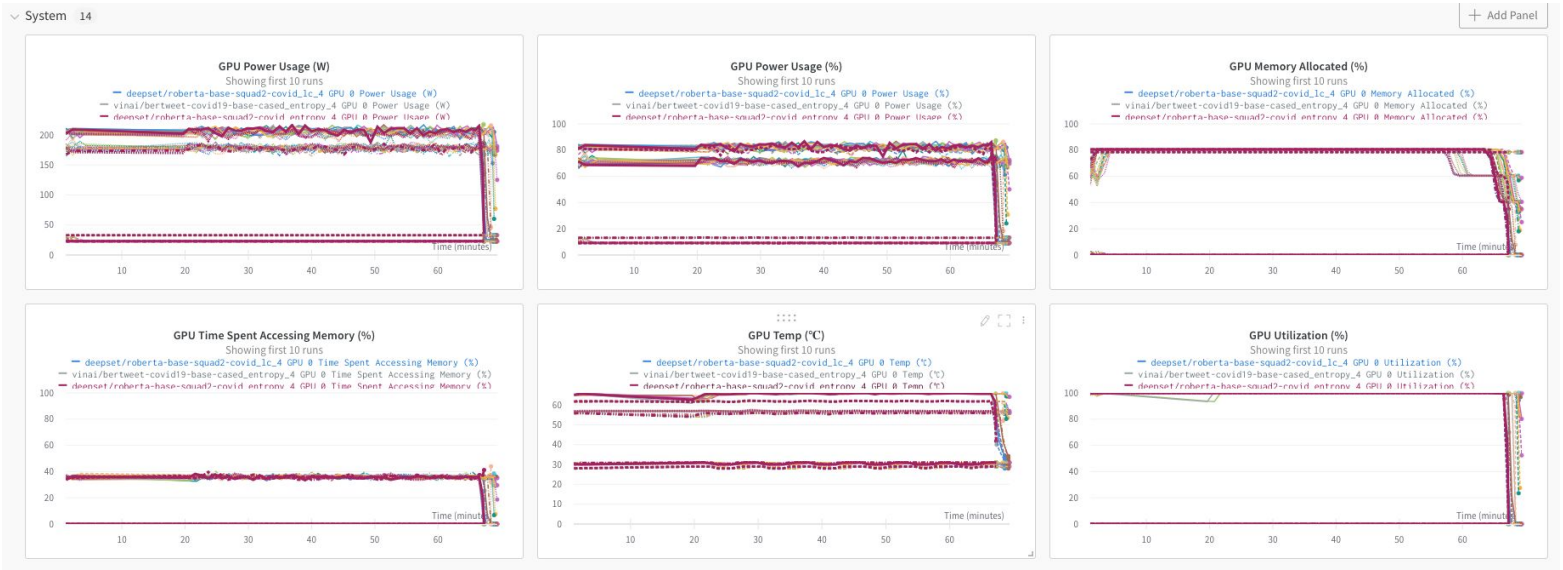
Solution Diagram



ML System Lifecycle

1. Model Implementation
    a. We use huggingface datasets hub to manage the dataset version and dataset processing.
    b. PytorchLightning for model implementation
2. Model Monitoring
    a. Wandb for logging and system monitoring
3. Model Prediction
    a. We save our prediction results and use unique runid to manage the Experiments Results.

ML System Lifecycle (System Monitoring)



ML System Lifecycle (Performance Tracking)

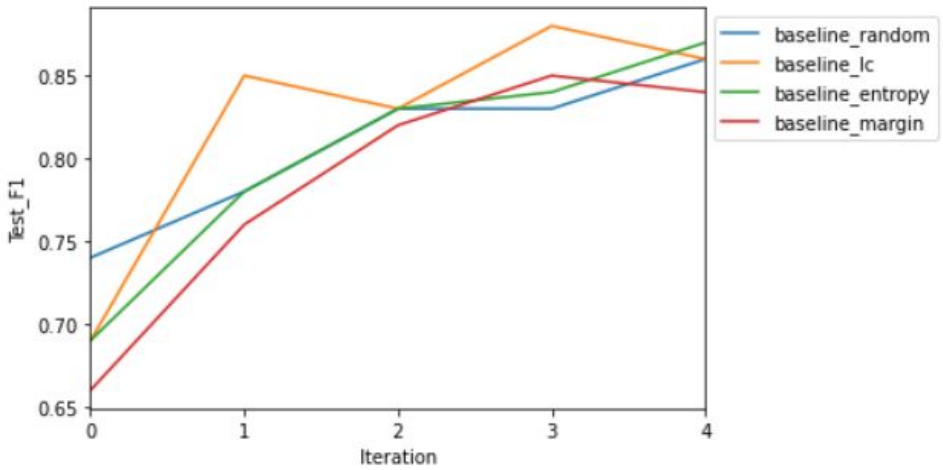
Implementation Detail

- We got the Pretrained Models from HuggingFace.
- Pytorch Lightning is used for model training and evaluation
- Weights and Biases is used for logging model results

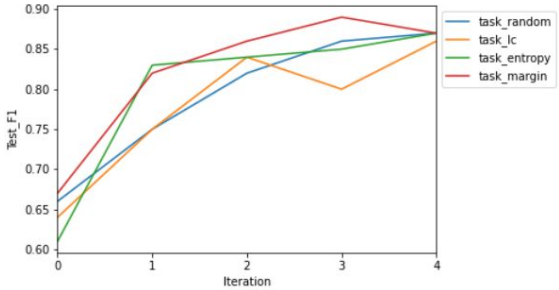
Implementation Detail

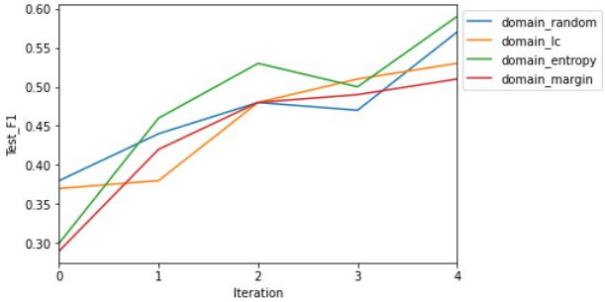
1. We first evaluated the entire training dataset and get the prediction distribution for each data points
2. Then we sampled based subset of training data to acquire actual labels and train the model.
3. From the updated model, we evaluate against the test dataset
4. We evaluate the remaining training dataset and repeat sampling and retraining until entire dataset is used.

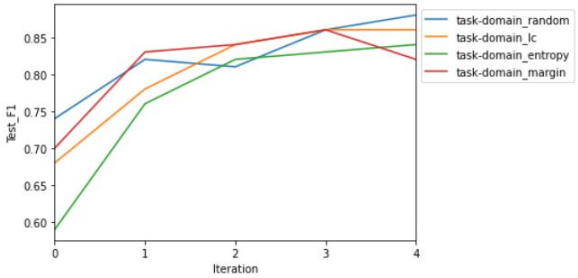
Demo (Active Learning Method Comparison)



Demo (Transfer Methods)

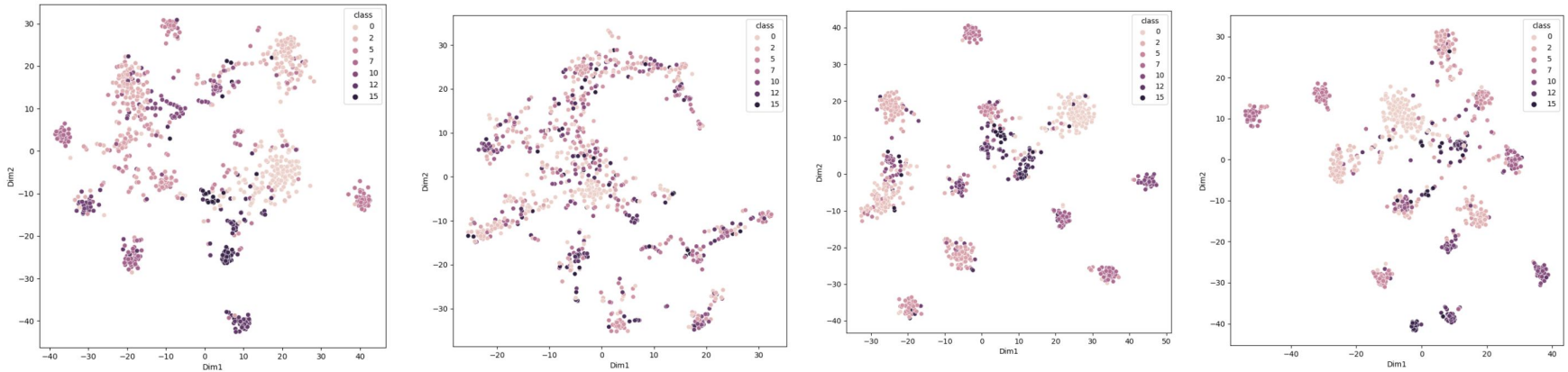






- Same Domain is Coivd PostTrained Model
- Same Task is SQuad2 Trained Model
- Domain-Task is SQuad2 Trained and PostTrained in Covid Domain
- Iteration: Active Learning Iteration
    - For each iteration, we simulate the annotation based on the sampling method

Demo (Representation Visualization)



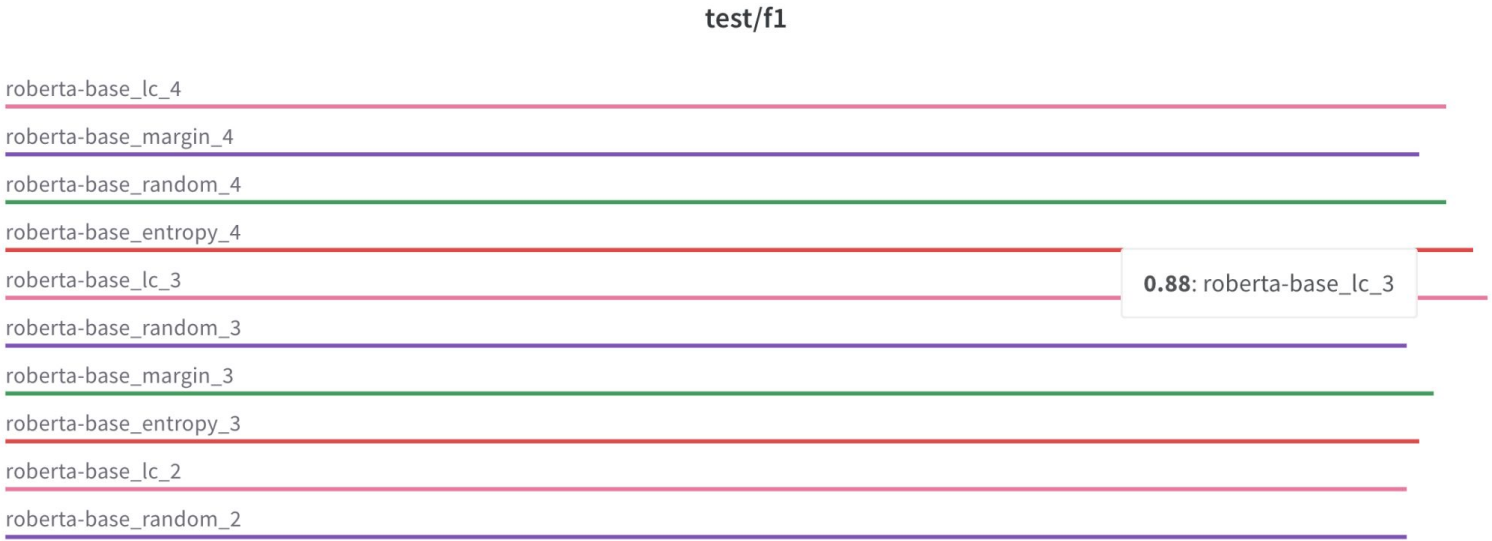
- From left to right: Baseline, Domain, Task, Domain&Task
- We can see the Transferred Models' representation is more clear separated
- The Domain PostTrained Model may affect the initial self-clustering property.

Demo (Distance Metrics)

Log_name	Max_l2_dist	Min_l2_dist	Mean_l2_dist	Max_cos_dis	Min_cos_dist	Mean_cos_di
baseline_0	14.14	5.09	9.38	1	-0.37	0.13
baseline_1	14.8	7.85	11.33	1	-0.3	0.03
baseline_2	14.57	7.55	11.07	1	-0.25	0.07
baseline_3	14.89	10.78	12.23	1	-0.2	0.03
baseline_4	15.05	11.3	12.43	1	-0.18	0.02
domain_0	4.07	1.66	2.52	1	0.12	0.56
domain_1	5.94	2.27	3.69	1	-0.1	0.27
domain_2	5.76	2.96	4.11	1	-0.32	0.19
domain_3	6.87	3.7	4.72	1	-0.25	0.14
domain_4	6.51	4.03	5	1	-0.27	0.15
task_0	12.9	5.41	9.23	1	-0.3	0.09
task_1	14.45	5.46	10.39	1	-0.28	0.1
task_2	14.83	9.5	11.96	1	-0.22	0.03
task_3	14.84	11.39	12.41	1	-0.17	0.02
task_4	14.65	10.56	12.21	1	-0.19	0.04
task-domain_0	13.36	5.37	9.48	1	-0.33	0.06
task-domain_1	13.76	6.57	10.47	1	-0.36	0.06
task-domain_2	14.69	9.19	11.74	1	-0.25	0.02
task-domain_3	14.84	9.89	11.83	1	-0.2	0.02
task-domain_4	14.45	8.74	11.77	1	-0.3	0.04

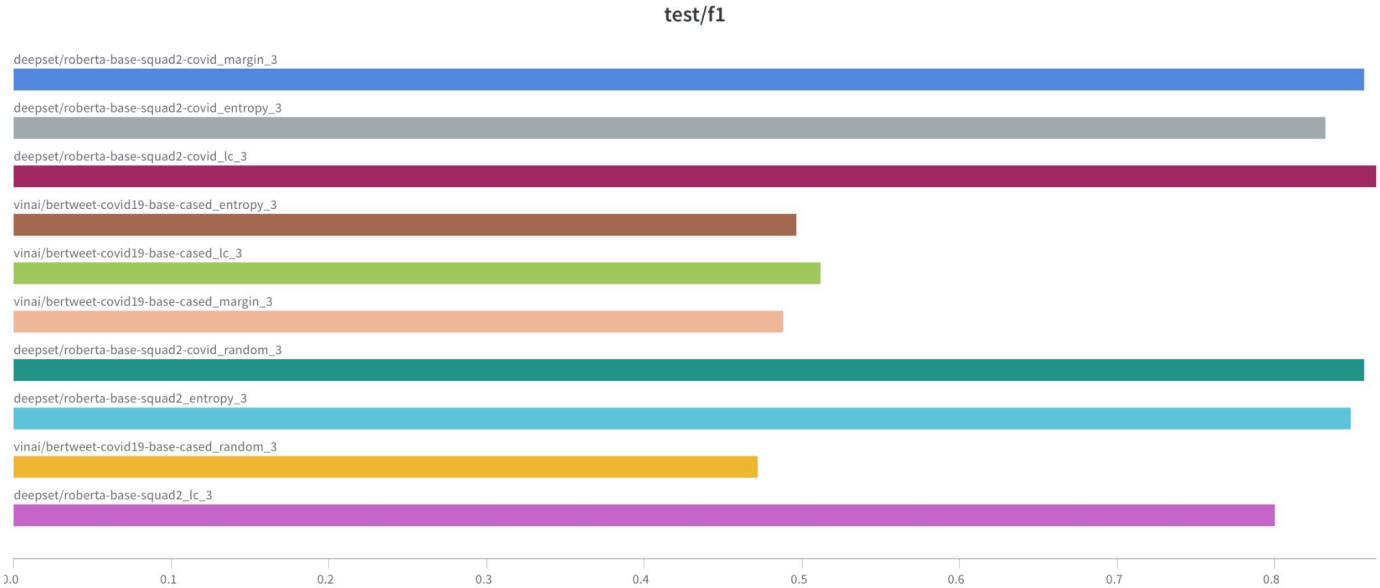
- The Distance is getting larger between clustering centric with the Active Learning Iterations
- L2 Distance maybe a better metric for distance measurement in our case.
- Domain mean distance is less than baseline, which match our representation visualization results.

Experimental Evaluation



- For Active Selection Methods, the least confidence works best for baseline model.

Experimental Evaluation



- For Transfer Learning Methods, Task&Domain transferred model works best, and the transferred model is compatible with the active learning section methods.

Conclusion

- In this work, we explored how will different transferred methods affect the results. Specifically, we tested the same domain and same task transferred model, and the combined method. The results show that same task transferring helps more than same domain and the combine method works the best.
- To alleviate the data hungry problem in unseen domain, we use active learning data selection methods including random selection, Entropy Based Method, Least Confidence and Margin Based Method. The **least confidence method** works best in **a reasonable annotation range**. But for very small annotation size, random selection performs better.
    - We can use random selection method to **burn-in** the active learning process in unseen domain.
- We analyze the representation space of different methods, which give us some intuition about how to design sample selection methods.

Project Repo & READ.ME

https://github.com/Xiang-Pan/NYU_DL_Sys_Project