Cartography Active Learning

Mike Zhang

mikz@itu.dk

Barbara Plank

bapl@itu.dk

IT UNIVERSITY OF COPENHAGEN

Department of Computer Science
Copenhagen, Denmark

Reading group GroNLP, October 2021

Table of Contents

Table of Contents

Active Learning (AL)

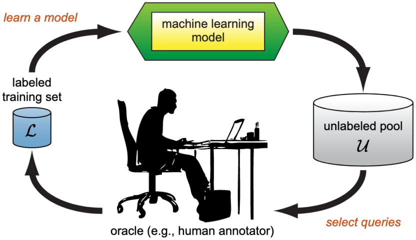

Figure 1: Pool-based Active Learning [Settles, 2009]

- Iterative process of selecting the data points the model "learns the most from" and adding it to the training set;

Active Learning (AL)

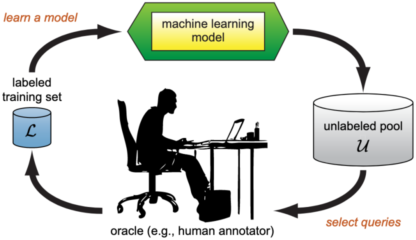

Figure 1: Pool-based Active Learning [Settles, 2009]

- Iterative process of selecting the data points the model "learns the most from" and adding it to the training set;
- Benefits: Achieve greater performance with fewer labeled training instances.

Table of Contents

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

Figure 2: Full data map of AGNews (120,000 training instances) w.r.t. an MLP.

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

Figure 2: Full data map of AGNews (120,000 training instances) w.r.t. an MLP.

Distinguishable regions: *easy-to-learn*, *ambiguous*, *hard-to-learn*. *Ambiguous* samples are best to train on (i.e., highest performance) [Swayamdipta et al., 2020].

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E} \sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E} \sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E} \sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E}\sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E}\left(p_{\boldsymbol{\theta}^{(e)}}(y_i^*|\boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E}\sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E}\left(p_{\boldsymbol{\theta}^{(e)}}(y_i^* | \boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E} \sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E} \left(p_{\boldsymbol{\theta}^{(e)}}(y_i^* | \boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$: Standard deviation of the confidence over E.

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E} \sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E} \left(p_{\boldsymbol{\theta}^{(e)}}(y_i^* | \boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$: Standard deviation of the confidence over E.

(3) Correctness

$\hat{\phi}_i = \frac{1}{E} \sum_{e=1}^{E} 1(\hat{y}_i = y_i^* \mid \boldsymbol{x}_i)$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E}\sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E}\left(p_{\boldsymbol{\theta}^{(e)}}(y_i^* | \boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$: Standard deviation of the confidence over E.

(3) Correctness

$\hat{\phi}_i = \frac{1}{E}\sum_{e=1}^{E} 1(\hat{y}_i = y_i^* \mid \boldsymbol{x}_i)$

Data maps

Data maps [Swayamdipta et al., 2020] help identify characteristics of instances within the broader trends of a dataset by leveraging their training dynamics (i.e., the behavior of a model during training).

(1) Confidence

$\hat{\mu}_i = \frac{1}{E}\sum_{e=1}^{E} p_{\boldsymbol{\theta}^{(e)}}(y_i^* \mid \boldsymbol{x}_i)$: Mean of output "probability" of the model over # of epochs (E).

(2) Variability

$\hat{\sigma}_i = \sqrt{\frac{\sum_{e=1}^{E}\left(p_{\boldsymbol{\theta}^{(e)}}(y_i^* | \boldsymbol{x}_i) - \hat{\mu}_i\right)^2}{E}}$: Standard deviation of the confidence over E.

(3) Correctness

$\hat{\phi}_i = \frac{1}{E}\sum_{e=1}^{E} 1(\hat{y}_i = y_i^* \mid \boldsymbol{x}_i)$: # of times gold label is correctly classified by the model over E.

Small data map

Do we still see this trend in a small labeled set?

Small data map

Do we still see this trend in a small labeled set? Yes.

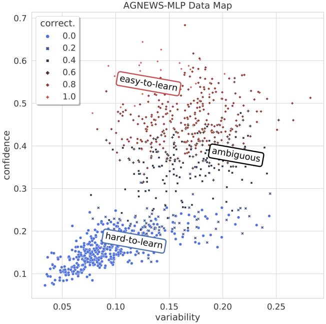

Figure 3: data map of AGNews (1,000 training instances) w.r.t. an MLP.

Table of Contents

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Process:

1. Use main classifier (for AL iteration) to create data map statistics for the small initial set.

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Process:

1. Use main classifier (for AL iteration) to create data map statistics for the small initial set.
2. Re-label small initial set with threshold t_{cor} (correctness) to train binary classifier on and apply to the unlabeled pool of data points.

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Process:

1. Use main classifier (for AL iteration) to create data map statistics for the small initial set.
2. Re-label small initial set with threshold t_{cor} (correctness) to train binary classifier on and apply to the unlabeled pool of data points.
3. Select top-k that are closest to the decision boundary: $\underset{\boldsymbol{x} \in \Psi(\mathcal{U})}{\operatorname{argmin}} |0.5 - P_{\theta'}(\hat{y} = 1 \mid \boldsymbol{x})|$

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Process:

1. Use main classifier (for AL iteration) to create data map statistics for the small initial set.
2. Re-label small initial set with threshold t_{cor} (correctness) to train binary classifier on and apply to the unlabeled pool of data points.
3. Select top-k that are closest to the decision boundary: $\underset{\boldsymbol{x} \in \Psi(\mathcal{U})}{\operatorname{argmin}} |0.5 - P_{\theta'}(\hat{y} = 1 \mid \boldsymbol{x})|$
4. Add the selected top-k to the initial labeled set.

Cartography Active Learning

Idea

Binary classifier to distinguish between *hard-to-learn* and *ambiguous/easy-to-learn* instances.

Process:

1. Use main classifier (for AL iteration) to create data map statistics for the small initial set.
2. Re-label small initial set with threshold t_{cor} (correctness) to train binary classifier on and apply to the unlabeled pool of data points.
3. Select top-k that are closest to the decision boundary: $\underset{\boldsymbol{x} \in \Psi(\mathcal{U})}{\operatorname{argmin}} \left|0.5 - P_{\theta'}(\hat{y} = 1 \mid \boldsymbol{x})\right|$
4. Add the selected top-k to the initial labeled set.
5. Repeat (1), (2), (3), and (4) until budget is exhausted.

Setup

Setup:

- Two text classification tasks, AGNews [Zhang et al., 2015] & TREC [Li and Roth, 2002];
- Multi-layer Perceptron (main classifier), one-layer network (binary classifier), FastText representations [Bojanowski et al., 2017], 30 AL iterations in batches of 50;
- Compared against: **Random** baseline, **Entropy** [Dagan and Engelson, 1995], **Least Confidence** [Culotta and McCallum, 2005], **BALD** [Gal and Ghahramani, 2016], **DAL** [Gissin and Shalev-Shwartz, 2019].

Does it work?

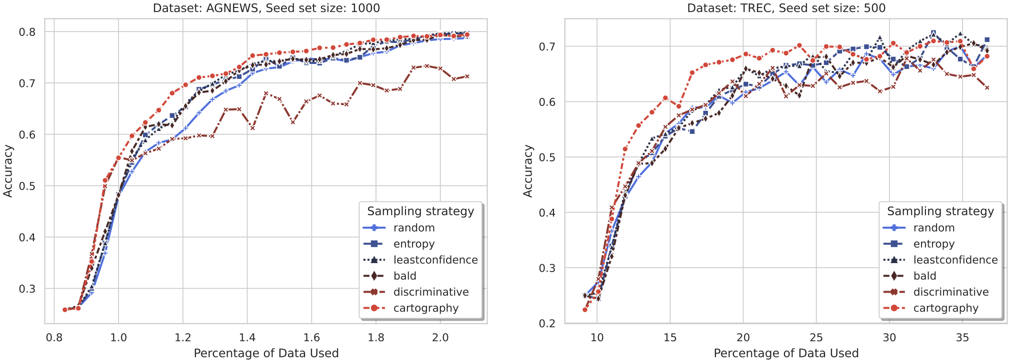

Figure 4: Performance AL strategies.

Why does it work?

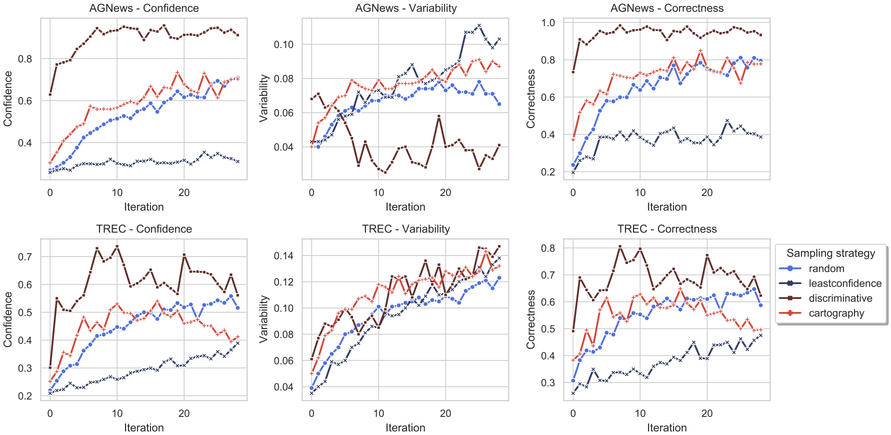

Figure 5: Statistics of several AL strategies over iterations.

Table of Contents

Conclusion

CAL is competitive or significantly outperforms various popular AL methods (more details in paper).

Thanks!

Code: `https://github.com/jjzha/cal`

Contact: `mikz@itu.dk`

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017).
Enriching word vectors with subword information.
Transactions of the Association for Computational Linguistics, 5:135–146.

Culotta, A. and McCallum, A. (2005).
Reducing labeling effort for structured prediction tasks.
In *AAAI*, volume 5, pages 746–751.

Dagan, I. and Engelson, S. P. (1995).
Committee-based sampling for training probabilistic classifiers.
In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier.

Gal, Y. and Ghahramani, Z. (2016).
Dropout as a bayesian approximation: Representing model uncertainty in deep learning.
In Balcan, M. and Weinberger, K. Q., editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org.

Gissin, D. and Shalev-Shwartz, S. (2019).
Discriminative active learning.
arXiv preprint arXiv:1907.06347.

Li, X. and Roth, D. (2002).
Learning question classifiers.
In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Settles, B. (2009).
Active learning literature survey.

Swayamdipta, S., Schwartz, R., Lourie, N., Wang, Y., Hajishirzi, H., Smith, N. A., and Choi, Y. (2020).
Dataset cartography: Mapping and diagnosing datasets with training dynamics.
In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293.

Zhang, X., Zhao, J. J., and LeCun, Y. (2015).
Character-level convolutional networks for text classification.

In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657.