Transferable Candidate Proposal with Bounded Uncertainty

Kyeongryeol Go KRGO@SUPERB-AI.COM
Kye-Hyeon Kim KHKIM@SUPERB-AI.COM
Superb AI
Seoul, South Korea

Abstract

From an empirical perspective, the subset chosen through active learning cannot guarantee an advantage over random sampling when transferred to another model. While it underscores the significance of verifying transferability, experimental design from previous works often neglected that the informativeness of a data subset can change over model configurations. To tackle this issue, we introduce a new experimental design, coined as Candidate Proposal, to find transferable data candidates from which active learning algorithms choose the informative subset. Correspondingly, a data selection algorithm is proposed, namely Transferable candidate proposal with Bounded Uncertainty (TBU), which constrains the pool of transferable data candidates by filtering out the presumably redundant data points based on uncertainty estimation. We verified the validity of TBU in image classification benchmarks, including CIFAR-10/100 and SVHN. When transferred to different model configurations, TBU consistency improves performance in existing active learning algorithms. Our code is available at `https://github.com/gokyeongryeol/TBU`.

Keywords: Active Learning, Transferability, Uncertainty Estimation

1. Introduction

Recent empirical observations show that the subset chosen through active learning, referred to as the active set, does not maintain its utility across different model architectures and experimental settings (Lowell et al., 2018; Munjal et al., 2022; Ji et al., 2023). Our pilot study, as illustrated in Figure 1, also reveals that when there is a mismatch in architectures or learning algorithms, a subset chosen by a proxy model offers no advantage over random sampling when fed to target models.

Ji et al. (2023) underlined the necessity to verify the transferability of the active set and encouraged the evaluation of the target model performance on selected subsets by the proxy model. However, such experimental design expects that there are universally informative data subsets regardless of model configurations, which we argue to be unrealistic. To illustrate the point, consider that many machine learning algorithms operate under the assumption that the effectiveness of individual data points to model training is not solely dependent on the data set itself. Instead, it encompasses numerous factors such as network architecture, learning algorithm, and attained performance. For example, curriculum learning (Soviany et al., 2022) and the adaptive data selection framework (Mirzasoleiman et al., 2020) propose that orchestrating the data loading process can yield tangible benefits. Moreover, in semi-supervised learning, Sohn et al. (2020); Wang et al. (2022b) selectively choose a

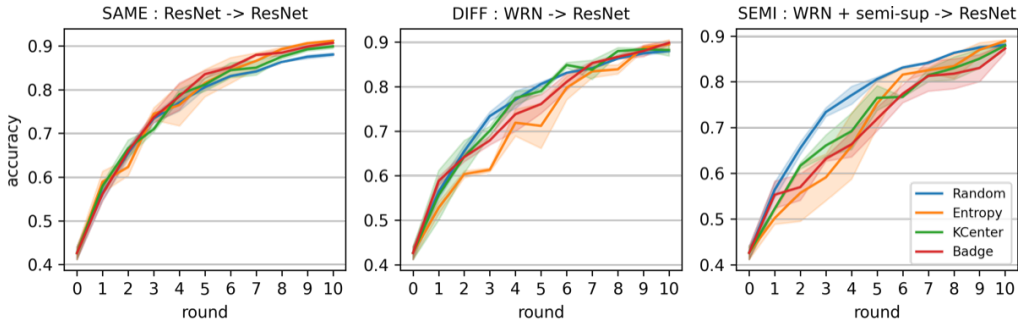


Figure 1: Per-round accuracy of distinct active learning algorithms in CIFAR-10. Regarding the dominance relationship and the resulting performance improvement, active sets achieve data efficiency over random sampling only when the proxy-target pair coincides (left). Transferability is not guaranteed when differing in architecture (middle) or learning algorithm (right). For a detailed explanation of SAME, DIFF, and SEMI, see section 4.

subset of unlabeled data to facilitate regularization. These suggest that the informativeness of data subsets evolves dynamically as the learning proceeds.

In this respect, we introduce a new experimental design called Candidate Proposal, which can verify the transferability depending on model configurations. Compared to the previous experimental design, the limited awareness of the proxy model regarding target models is compensated by refining the role of the proxy model while expanding the role of target models. Specifically, the proxy model only contributes to constraining a pool of potentially transferable data candidates within unlabeled data. Then, among those candidates, target models choose the informative subset of a fixed budget using active learning algorithms. Resorting to the training signal from the proxy model, the size of transferable data candidates can exceed the budget, and whether each element of candidates is informative depends on the target model configuration.

Besides the active learning algorithms for subset selection, Candidate Proposal additionally requires a candidate selection algorithm. We propose **Transferable candidate proposal with Bounded Uncertainty (TBU)** that filters out non-transferable data candidates using uncertainty estimation. Instances of low epistemic uncertainty (in brief, LE instances) offer minimal information gain (Zhou et al., 2022), whereas instances of high aleatoric uncertainty (in brief, HA instances) are susceptible to adversarial perturbations (Smith and Gal, 2018). TBU chooses data candidates from sources other than the LE and HA instances, preventing the inclusion of redundant data that provides marginal enhancements to model performance. The validity of TBU is demonstrated in image classification benchmarks by complementary effects on existing active learning algorithms.

2. Related Works

Conventional approaches, represented by Entropy (Gal et al., 2017) and CoreSet (Sener and Savarese, 2017), selected subsets based on either uncertainty or diversity. Despite their longstanding relevance, these methods remain competitive baselines in contemporary research. To achieve better-calibrated uncertainty estimations, newer techniques have

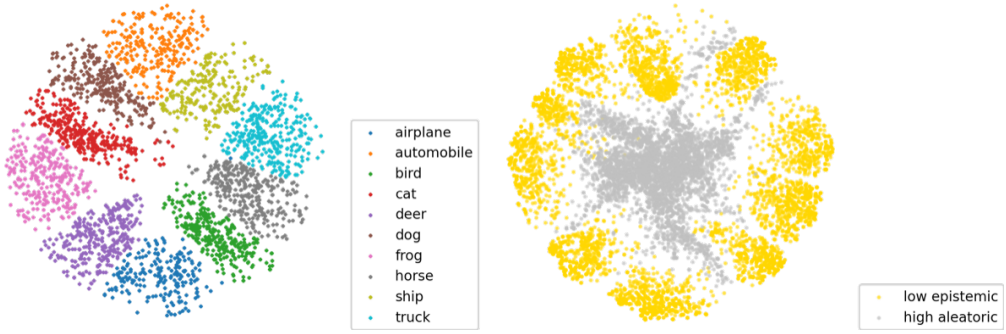


Figure 2: T-SNE embedding of the labeled (left) and the instances of the unlabeled filtered by TBU (right). Different colors indicate different class labels for the labeled data embedding, while the LE and HA instances in the unlabeled data embedding are denoted with gold and silver, respectively.

emerged. Monte-Carlo Dropout (Gal and Ghahramani, 2016) and Deep Ensemble (Lakshminarayanan et al., 2017), for instance, incorporate architectural modifications to enhance uncertainty estimation (Beluch et al., 2018). Additionally, approaches like Houlsby et al. (2011); Beluch et al. (2018); Kirsch et al. (2019) have quantified information gain through predictive entropy. Also, gradient information has been leveraged, combining uncertainty and diversity in a hybrid manner. Notably, Badge (Ash et al., 2019) used the K-means++ clustering in the gradient embedding space, and Liu et al. (2021); Wang et al. (2022a) quantified the expected change in model performance using the influence function. Commonly, most of the previous studies have been dedicated to better exploiting the captured training signal. Please refer to Appendix A for a more detailed background on the acquisition functions.

The most similar work is the data valuation framework (Ghorbani and Zou, 2019; Kwon and Zou, 2021; Wang and Jia, 2023), which compares the relative contribution of each training data point to clarify both the most effective and defective data points regardless of model configurations. Yet, due to expensive computational costs, their practical usage is limited to low-dimensional data. To the best of our knowledge, this is the first work on verifying the transferability of the Candidate Proposal in active learning.

3. Methodology

This section explains TBU, the candidate selection algorithm for Candidate Proposal. TBU filters out LE and HA instances using deterministic uncertainty methods and semi-supervised learning. As illustrated in Figure 2, LE instances tend to have far margins from the decision boundaries while HA instances have a small margin from the decision boundary. Please refer to Appendix B for the overall pipeline.

3.1 Quantifying Epistemic Uncertainty

In estimating epistemic uncertainty, we resort to the deterministic uncertainty methods (Liu et al., 2020; van Amersfoort et al., 2021; Mukhoti et al., 2023) which has been regarded as

a valid substitute for Bayesian Neural Networks (BNN). Among various choices, we applied the last-layer Laplace approximation (Kristiadi et al., 2020) for its simplicity and scalability.

The last-layer Laplace approximation (Kristiadi et al., 2020; Daxberger et al., 2021) falls into the category that replaces the final soft-max layer with a distance-sensitive function. Equipping the parameter $w = [w_1, ..., w_C]$ of the linear classifier with prior $p(w_c) = \mathcal{N}(w_c; 0, I)$ and posterior $p(w_c|D) = \mathcal{N}(w_c; \mu_c, \Sigma_c)$, the proxy network is now composed of a fixed feature extractor ϕ and a bayesian linear classifier. The parameter of posterior distribution can be estimated via Laplace approximation. Specifically, $\mu = [\mu_1, ..., \mu_C]$ is the Maximum A Posteriori (MAP) estimate, and $\Sigma = [\Sigma_1, ..., \Sigma_C]$ is the inverse of the precision matrix that is computed as the Hessian of negative log posterior at μ.

$$\mu = \arg\max_w \log p(D|w)p(w)$$
$$\Sigma = (-\nabla_w^2 \log p(D|w)p(w)|_{w=\mu})^{-1}$$

Note that D denotes the train data and $p(D|w) = \prod_{(x,y)\in D} p(y|\phi(x), w)$ implies the likelihood. Then, the model prediction turns to marginalization over the posterior such that

$$p(y|x, D) = \int p\left(y|\phi(x), w\right) p(w|D)dw = \int \sigma\left(\phi(x)^T w\right) p(w|D)dw$$

where σ implies the soft-max activation function. Please refer to Appendix C for details on the approximation for the inference process.

We regard the unlabeled data instances as having low epistemic uncertainty if their predictive entropy is small enough. Specifically, for each predicted class label, the unlabeled data instances whose predictive entropy is lower than q-th percentile of that of the labeled data set are categorized as LE instances.

3.2 Identifying Instances of High Aleatoric Uncertainty

Instances of high aleatoric uncertainty can be inspected more straightforwardly. Unlike previous approaches that explicitly incorporate complex noise distribution using parameter-intensive modules (Fortuin et al., 2021; Collier et al., 2021, 2023), we rely on semi-supervised learning, harnessing the unlabeled data to identify HA instances effectively. In short, we employ the class-wise confidence threshold proposed in FreeMatch (Wang et al., 2022b) to set an upper limit on the predictive confidence.

Free-match (Wang et al., 2022b) suggested iteratively updating the class-wise confidence threshold to lower bound the beneficial subset of the unlabeled data set. For every iteration t, global threshold $g^{(t)} \in \mathbb{R}$ and the local threshold $l^{(t)} = [l_1^{(t)}, ..., l_C^{(t)}] \in \mathbb{R}^C$ are updated using the model predictive confidence $[p_1, ..., p_C] \in \mathbb{R}^C$ as follows:

$$g^{(t)} = \eta \cdot g^{(t-1)} + (1-\eta) \cdot \frac{1}{|S|} \sum_{s=1}^{|S|} max(p_1, ..., p_C)$$
$$l_c^{(t)} = \eta \cdot l_c^{(t-1)} + (1-\eta) \cdot \frac{1}{|S|} \sum_{s=1}^{|S|} p_c \quad \forall c \in [C]$$

		1st	2nd	3rd	4th
Entropy	SAME	0.589 ± 0.024	0.623 ± 0.019	0.745 ± 0.014	0.793 ± 0.026
	DIFF	0.527 ± 0.013	0.603 ± 0.007	0.613 ± 0.004	0.710 ± 0.027
	SEMI	0.502 ± 0.013	0.557 ± 0.062	0.591 ± 0.051	0.648 ± 0.034
	TBU(q=10)	0.573 ± 0.021	0.682 ± 0.039	0.766 ± 0.018	0.807 ± 0.015
	TBU(q=20)	**0.617 ± 0.006**	0.653 ± 0.065	**0.768 ± 0.022**	**0.814 ± 0.010**
	TBU(q=50)	0.546 ± 0.032	**0.701 ± 0.018**	0.759 ± 0.014	0.804 ± 0.010
CoreSet	SAME	0.579 ± 0.008	0.666 ± 0.019	0.710 ± 0.006	0.786 ± 0.018
	DIFF	0.556 ± 0.056	0.643 ± 0.036	0.701 ± 0.011	0.760 ± 0.031
	SEMI	0.522 ± 0.003	0.617 ± 0.007	0.661 ± 0.023	0.707 ± 0.028
	TBU(q=10)	0.568 ± 0.030	**0.675 ± 0.004**	**0.743 ± 0.028**	**0.796 ± 0.013**
	TBU(q=20)	**0.608 ± 0.005**	0.674 ± 0.005	0.733 ± 0.024	0.784 ± 0.010
	TBU(q=50)	0.586 ± 0.024	0.654 ± 0.024	0.740 ± 0.008	0.768 ± 0.011
Badge	SAME	0.561 ± 0.016	0.659 ± 0.009	0.736 ± 0.012	0.784 ± 0.022
	DIFF	0.588 ± 0.009	0.642 ± 0.010	0.679 ± 0.010	0.766 ± 0.025
	SEMI	0.553 ± 0.030	0.570 ± 0.029	0.632 ± 0.005	0.708 ± 0.020
	TBU(q=10)	**0.601 ± 0.030**	0.680 ± 0.028	0.744 ± 0.009	**0.819 ± 0.013**
	TBU(q=20)	0.584 ± 0.036	**0.696 ± 0.010**	**0.754 ± 0.024**	0.797 ± 0.013
	TBU(q=50)	0.566 ± 0.018	0.672 ± 0.014	0.743 ± 0.021	0.780 ± 0.037

Table 1: Per-round accuracy of the target model with ablation of the percentile q in TBU. The first and the second highest values are marked in bold and underlined, respectively.

Note that $g^{(0)}$ and every element of $l^{(0)}$ are initialized to $\frac{1}{C}$, and η denotes the coefficient of the exponential moving average. The local threshold then modulates the global threshold to compute the class-wise confidence threshold.

$$\tau_c^{(t)} = g^{(t)} \cdot l_c^{(t)} / max(l_1^{(t)}, ..., l_C^{(t)})$$

We used this threshold to upper bound the predictive confidence of HA instances in the unlabeled data set since those are uncertain enough to be ignored throughout model training. Practically, considering that the model prediction can drastically change across epochs (Toneva et al., 2018), only the unlabeled instances that consistently show low confidence for five times of periodic evaluation at every 12.5% of total epochs are regarded to have high aleatoric uncertainty. Please refer to Appendix D detailing the rationale in class-wise confidence threshold.

4. Experiment

We conducted experiments on several image classification benchmarks in active learning, including CIFAR-10/100 and SVHN. In every round, proxy and target models are trained from scratch with a cumulative collection of the selected subsets from the previous rounds. In between rounds, the proxy model conducts the candidate proposal by filtering out the redundant instances using TBU, and the target model selects the subset of a fixed budget within those candidates. To verify the efficacy of TBU, we considered three baselines, namely SAME, DIFF, and SEMI. These do not use the candidate proposal but select a fixed-sized subset either from the proxy or the target.[1]

1. Unless otherwise specified, all the baseline models are trained using supervised learning.

- SAME: The proxy and the target coincide. In other words, the target selects the subset from the unconstrained pool of unlabeled data.
- DIFF: The proxy and the target have different architectures, and the proxy selects a fixed-sized subset on behalf of the target.
- SEMI: Keeping all other conditions identical to DIFF, the proxy is trained by semi-supervised learning.

For generality, active learning algorithms of distinct categories (uncertain, diverse, hybrid), represented by Entropy, CoreSet, and Badge, are considered. For reproducibility, we made our codes available. Please refer to Appendix E for experimental settings.

In Table 1, we reported the per-round accuracy of the target model in CIFAR-10, revealing three intriguing outcomes. First, SEMI appears to fall short when compared to DIFF. This outcome might strike one as counter-intuitive, given that more advanced learning techniques, such as semi-supervised learning, typically lead to a deeper understanding of the underlying data distribution. We posit this result as empirical evidence illustrating that the informativeness of a data point is not solely dictated by the data set itself; instead, it can vary depending on the specific model configurations employed. Second, TBU has the potential to surpass SAME in performance, irrespective of the active learning algorithms. This suggests that, without altering the acquisition function, the effectiveness of existing active learning algorithms can be enhanced by constraining the search space of target models through the proxy model using TBU. Third, TBU is robust to the choice of the percentile q unless it is set too high. Throughout the experiments, we fix q to 10.

Beyond the findings from per-round accuracy in CIFAR-10, we have investigated TBU through experiments in various aspects. To begin with, for generality, we demonstrated that TBU could achieve higher per-round accuracy over baselines even when varying data sets and target model architectures. Furthermore, the trained model with TBU is shown to be robust to input perturbation when evaluated in CIFAR-10/100-C. Next, for qualitative analysis, instances of low epistemic uncertainty exhibit semantic similarity in their imagery, whereas instances of high aleatoric uncertainty incorporate fragmented object components or exhibit a distinct foreground/background color palette. Lastly, TBU could automatically schedule the difficulty of subsets along selection rounds. Due to space constraints, we report these experimental results in Appendix F.

5. Conclusion

Although active learning has a rich historical background and enjoys consensus among researchers regarding its importance, the application of active learning remains a challenge due to empirical criticisms. We aim to expand upon the previous experimental design by the Candidate Proposal and propose a candidate selection algorithm, namely Transferable candidate proposal with Bounded Uncertainty (TBU). Throughout the experiments, it is demonstrated that TBU can seamlessly integrate with existing active learning algorithms, offering consistent performance improvement.

References

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9368–9377, 2018.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*, 2019.

Mark Collier, Basil Mustafa, Efi Kokiopoulou, Rodolphe Jenatton, and Jesse Berent. Correlated input-dependent label noise in large-scale image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1551–1560, 2021.

Mark Collier, Rodolphe Jenatton, Basil Mustafa, Neil Houlsby, Jesse Berent, and Effrosyni Kokiopoulou. Massively scaling heteroscedastic classifiers. *arXiv preprint arXiv:2301.12860*, 2023.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux-effortless bayesian deep learning. *Advances in Neural Information Processing Systems*, 34:20089–20103, 2021.

Michael Dusenberry, Ghassen Jerfel, Yeming Wen, Yian Ma, Jasper Snoek, Katherine Heller, Balaji Lakshminarayanan, and Dustin Tran. Efficient and scalable bayesian neural nets with rank-1 factors. In *International conference on machine learning*, pages 2782–2792. PMLR, 2020.

Vincent Fortuin, Mark Collier, Florian Wenzel, James Allingham, Jeremiah Liu, Dustin Tran, Balaji Lakshminarayanan, Jesse Berent, Rodolphe Jenatton, and Effrosyni Kokiopoulou. Deep classifiers with label noise modeling and distance awareness. *arXiv preprint arXiv:2110.02609*, 2021.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR, 2017.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International conference on machine learning*, pages 2242–2251. PMLR, 2019.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Yilin Ji, Daniel Kaestner, Oliver Wirth, and Christian Wressnegger. Randomness is the root of all evil: More reliable evaluation of deep active learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3943–3952, 2023.

Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32, 2019.

Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *International conference on machine learning*, pages 5436–5446. PMLR, 2020.

Yongchan Kwon and James Zou. Beta shapley: a unified and noise-reduced data valuation framework for machine learning. *arXiv preprint arXiv:2110.14049*, 2021.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems*, 33: 7498–7512, 2020.

Jeremiah Zhe Liu, Shreyas Padhy, Jie Ren, Zi Lin, Yeming Wen, Ghassen Jerfel, Zachary Nado, Jasper Snoek, Dustin Tran, and Balaji Lakshminarayanan. A simple approach to improve single-model deep uncertainty via distance-awareness. *J. Mach. Learn. Res.*, 24: 42–1, 2023.

Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234: 11–26, 2017.

Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence selection for active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9274–9283, 2021.

David Lowell, Zachary C Lipton, and Byron C Wallace. How transferable are the datasets collected by active learners. *arXiv preprint arXiv:1807.04801*, 3, 2018.

Zhiyun Lu, Eugene Ie, and Fei Sha. Mean-field approximation to gaussian-softmax integral with application to uncertainty estimation. *arXiv preprint arXiv:2006.07584*, 2020.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54 (6):1–35, 2021.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, pages 6950–6960. PMLR, 2020.

Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023.

Prateek Munjal, Nasir Hayat, Munawar Hayat, Jamshid Sourati, and Shadab Khan. Towards robust and reproducible active learning using neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 223–232, 2022.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14: 265–294, 1978.

Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Lewis Smith and Yarin Gal. Understanding measures of uncertainty for adversarial example detection. *arXiv preprint arXiv:1803.08533*, 2018.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409*, 2021.

Haonan Wang, Wei Huang, Ziwei Wu, Hanghang Tong, Andrew J Margenot, and Jingrui He. Deep active learning by leveraging training dynamics. *Advances in Neural Information Processing Systems*, 35:25171–25184, 2022a.

Jiachen T Wang and Ruoxi Jia. Data banzhaf: A robust data valuation framework for machine learning. In *International Conference on Artificial Intelligence and Statistics*, pages 6388–6421. PMLR, 2023.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022b.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020.

Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 93–102, 2019.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021.

Xinlei Zhou, Han Liu, Farhad Pourpanah, Tieyong Zeng, and Xizhao Wang. A survey on epistemic (model) uncertainty in supervised learning: Recent advances and applications. *Neurocomputing*, 489:449–465, 2022.

Appendix A. Background on Acquisition Functions

Along with transfer learning, semi-supervised learning, and data augmentation, active learning aims to train machine learning models with limited labeled data effectively. Concretely, let $L = \{(x_l, y_l)\}$ and $U_x = \{x_u\}$ denote the labeled and unlabeled data set. The model is trained from scratch with L using supervised learning or L and U_x using semi-supervised learning at each iteration. Then, a model-dependent acquisition function selects a subset of size K from U_x, labeled by an oracle to be added to L.

Over the past few years, deep neural networks have been highly regarded in active learning for their flexibility as function approximators and broad applicability in various data analysis tasks, such as dimensionality reduction and uncertainty estimation (Liu et al., 2017). Therefore, active learning research has focused on developing improved acquisition functions using deep neural networks, namely deep active learning (Ren et al., 2021).

The acquisition function, also known as the utility, is crucial in quantifying the informativeness for selecting uncertain and diverse subsets. In essence, it is a set function, and its purpose is to identify the subset that is most likely to yield maximal performance improvements within a fixed budget. However, traversing and comparing all possible combinations of an unlabeled data set is not computationally scalable. Consequently, many approaches concentrate on evaluating each data instance independently by considering subsets of size one and selecting the top-K instances based on their scores (Gal et al., 2017; Yoo and Kweon, 2019; Liu et al., 2021; Wang et al., 2022a). Alternatively, another approach uses an acquisition function that exhibits the properties of monotonicity and sub-modularity, which transforms the problem into sub-modular optimization (Sener and Savarese, 2017; Kirsch et al., 2019; Mirzasoleiman et al., 2020). By employing a greedy selection strategy, these methods achieve a solution with only a small approximation error compared to the optimal solution (Nemhauser et al., 1978).

Following are the active learning algorithms we have considered throughout the experiments, which are representative of distinct categories (uncertainty, diversity, hybrid).

- Entropy (Gal et al., 2017): top-K samples of predictive entropy
- CoreSet (Sener and Savarese, 2017): K greedy selection based on the maximum distances between the feature embeddings of the labeled and unlabeled data sets
- Badge (Ash et al., 2019): K-means++ clustering results in gradient embedding space with respect to the weight parameter of the last linear layer

Appendix B. Pipeline of Candidate Proposal with TBU

In Figure 3, we compare the previous experimental design and our new experimental design, Candidate Proposal. Given an unlabeled data set, previous experimental designs apply active learning on training signals from the proxy model to choose an informative subset for target models. In Candidate Proposal, transferable data candidates are first determined by TBU that filters out LE and HA instances, then an informative subset is chosen from the candidates by target models. Please refer to algorithm 1 for the step-by-step description.

Now, we describe how TBU differs from the three baselines that appeared in section 4, SAME, DIFF, and SEMI, which correspond to the previous experimental design. While

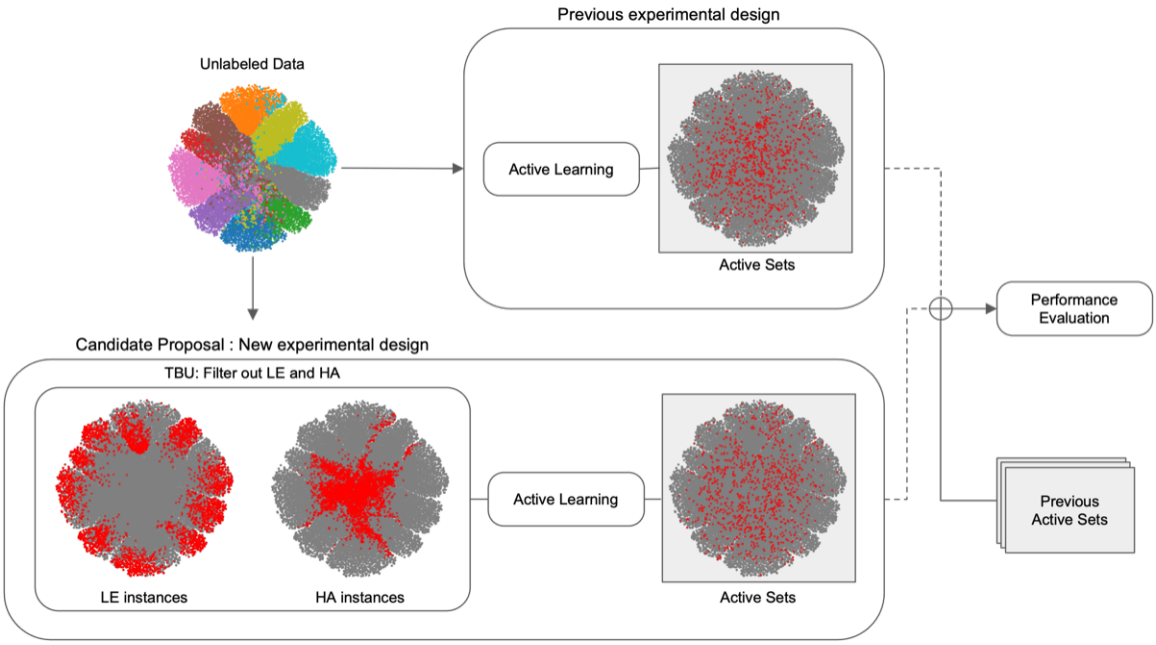


Figure 3: An overall pipeline of Candidate Proposal with TBU in comparison to the previous experimental design. TSNE embeddings of unlabeled data are attached along the pipeline. The unlabeled data in the top-left corner is colored by the ground-truth class labels. For the rest, the data points corresponding to the text description written below are colored red while the remaining are colored gray.

DIFF and SEMI assume different model configuration for the proxy and target models, SAME use the same configuration. Therefore, SAME is the only baseline where the target model selects the subset by itself. Likewise, TBU also allows the target model to directly search the informative subset. However, unlike SAME, the search space is shrunk from the whole unlabeled data set by filtering out presumably redundant data points using the proxy model of different configurations.

Appendix C. Last-layer Laplace Approximation

The last-layer Laplace approximation (Kristiadi et al., 2020; Daxberger et al., 2021) falls into the category that replaces the final soft-max layer with a distance-sensitive function. Equipping the parameter $w = [w_1, ..., w_C] \in \mathbb{R}^{F \times C}$ of the linear classifier with prior $p(w_c) = \mathcal{N}(w_c; 0, I)$ and posterior $p(w_c|D) = \mathcal{N}(w_c; \mu_c, \Sigma_c)$, the proxy network is now composed of a fixed feature extractor ϕ and a bayesian linear classifier. Then, the model prediction turns to marginalization over the posterior such that

$$p(y|x, D) = \int p\left(y|\phi(x), w\right) p(w|D) dw = \int \sigma\left(\phi(x)^T w\right) p(w|D) dw \tag{1}$$

where D is the train data and σ implies the soft-max activation function. The parameter of posterior distribution can be estimated via Laplace approximation. Specifically, $\mu =$

Algorithm 1: Transferable Candidate Proposal with Bounded Uncertainty (TBU)

Dataset : labeled, unlabeled, validation data set: L, U_x, V
Model : proxy, target models: f_p, f_t
Input: acquisition function: $\mathcal{A}$, scaling factor: λ, percentile: q, budget size: K
Output: acquisition batch: S
Randomly initialize the parameters of proxy and target models
while *not converged* **do**
  Update parameters of the proxy model f_p with L and U_x
  Following Equation 4, update global and local threshold
end
Following Equation 2, compute the logit variance $\hat{\Sigma}$ with L
Following Equation 3, use L-BFGS optimizer for λ to minimize NLL in V
Inspect LE instances S_{LE} of lower predictive entropy than the q-th percentile in L
Inspect HA instances S_{HA} of lower predictive confidence than Equation 5
while *not converged* **do**
  Update parameters of the target model f_t with L
end
$S \leftarrow \emptyset$, $C \leftarrow U_x - (S_{LE} \cup S_{HA})$
for $k = 1, ..., K$ **do**
  $s = \arg\max_c \mathcal{A}(f_t, x_c)$ for $x_c \in C$
  $S \leftarrow S \cup \{x_s\}, C \leftarrow C - \{x_s\}$
end

$[\mu_1, ..., \mu_C]$ is the Maximum A Posteriori (MAP) estimate, and $\Sigma = [\Sigma_1, ..., \Sigma_C]$ is the inverse of the precision matrix that is computed as the Hessian of negative log posterior at μ.

$$\mu = \arg\max_w \log p(D|w)p(w)$$
$$\Sigma = (-\nabla_w^2 \log p(D|w)p(w)|_{w=\mu})^{-1}$$

Note that $p(D|w) = \prod_{(x,y)\in D} p(y|\phi(x), w)$ denotes the likelihood. μ can be obtained by standard neural network training with L2 regularization, and Σ is computed after model training as post-processing, which has a closed-form expression that does not require a backward pass for computing the Hessian. Considering the scalability in a dataset of too many classes, the shared covariance $\hat{\Sigma}$ is used throughout the experiments.

$$\hat{\Sigma} = \left(\sum_{(x,y)\in D} p^*(1-p^*)\phi(x)\phi(x)^T \right)^{-1} \tag{2}$$

This is proven to be an upper bound for all class-specific covariances Σ_c, where $p^* = \max(p_1, ..., p_C)$ denotes the predictive confidence (Liu et al., 2023). The covariance is estimated only once before the inference phase, which is computationally affordable compared to other efficient ensemble methods (Wen et al., 2020; Dusenberry et al., 2020).

Mean-field approximation for Gaussian-Softmax integral (Lu et al., 2020) can play a role as a decent replacement of Monte-Carlo averaging for Equation 1

$$p(y|x, D) = \int \sigma(l)\, \mathcal{N}\left(l; \phi(x)^T \mu, v(x)\right) dl \approx \sigma\left(\frac{\phi(x)^T \mu}{\sqrt{1 + \lambda \cdot v(x)}}\right) \tag{3}$$

where λ is a scaling hyper-parameter. Note that this enables the inference by a single forward pass. Regarding the symbols, $v(x) = \phi(x)^T \hat{\Sigma} \phi(x)$ is a quadratic form of the feature embedding $\phi(x)$ with respect to the computed covariance $\hat{\Sigma}$, namely logit variance. Note that compared to a standard way of prediction during inference, the logit variance is additionally introduced to soften the prediction as temperature scaling.

After the standard model training, we compute the covariance $\hat{\Sigma}$ using the labeled dataset and estimate the optimal λ by L-BFGS optimizer minimizing negative log-likelihood in the validation data set.

Appendix D. Rationale in Class-wise Confidence Threshold

By leveraging unlabeled data, semi-supervised learning has been highlighted to mitigate the requirement of substantial labeling costs and achieve comparable performance to a fully supervised setting with only a small amount of data (Yang et al., 2022). Based on consistency regularization, the model acquires robustness to input perturbations, and the pseudo-labeling technique makes the decision boundary located in the low-density regions to separate the different class clusters.

FixMatch (Sohn et al., 2020) is the most widely adopted algorithm in image classification and object detection. It uncovers the beneficial subset of the unlabeled data set for training using a fixed confidence threshold τ and two data augmentations of different strengths. Specifically, for every mini-batch of the unlabeled data set, it first computes the pseudo-label of weakly augmented input and only leaves the subset S whose confidence is greater than or equal to τ

$$S = \{x | \max(p_1, ...p_C) \geq \tau \quad \text{s.t.} \quad \forall c \in [C], \quad p_c = p\left(y = c | a(x), D\right)\}$$

where $a(\cdot)$ denotes the weak augmentation. Then, the strongly augmented input of the subset is used to train the model along with the mini-batch of the labeled data set.

As τ is generally set to a high value in FixMatch (for example, 0.95 in CIFAR-10/100), the convergence speed of FixMatch can be slow whenever the model struggles during training, especially in the early stage. Therefore, variants of FixMatch have been proposed to adaptively tune the confidence threshold τ online (Zhang et al., 2021; Wang et al., 2022b). Notably, FreeMatch (Wang et al., 2022b) suggested to iteratively updating the global threshold $g^{(t)} \in \mathbb{R}$ and the local threshold $l^{(t)} = [l_1^{(t)}, ..., l_C^{(t)}] \in \mathbb{R}^C$ for every iteration t. Specifically, by initializing $g^{(0)}$ and every element of $l^{(0)}$ to $\frac{1}{C}$, the thresholds are updated

using the model predictive confidence

$$g^{(t)} = \eta \cdot g^{(t-1)} + (1-\eta) \cdot \frac{1}{|S|} \sum_{s=1}^{|S|} max(p_1, ..., p_C) \tag{4a}$$

$$l_c^{(t)} = \eta \cdot l_c^{(t-1)} + (1-\eta) \cdot \frac{1}{|S|} \sum_{s=1}^{|S|} p_c \quad \forall c \in [C] \tag{4b}$$

where η is the coefficient of the exponential moving average. Then, the global threshold is modulated by the local threshold to compute the class-wise adaptive threshold.

$$\tau_c^{(t)} = g^{(t)} \cdot l_c^{(t)} / max(l_1^{(t)}, ..., l_C^{(t)}) \tag{5}$$

We used the threshold to identify the instances of high aleatoric uncertainty by upper bounding the predictive confidence in the unlabeled data set.

Appendix E. Experimental Settings

We performed experiments in CIFAR-10/100 and SVHN, the commonly used image classification benchmarks in deep active learning. As a default data augmentation, we used random horizontal flip, random cropping followed by padding, and normalization.

The number of initially labeled data is set to 1000 in CIFAR-10, 5000 in CIFAR-100, and 1000 in SVHN, while the budget is set to 1000, 2500, and 1000, respectively. These values are inspected based on the insight that the optimal experimental settings to verify the effectiveness of active learning may differ across the data sets. We expect the initial accuracy to be over 40 and below 80 when trained by supervised learning. This is to obtain moderately reliable training signals and reflect the preference for considering both uncertainty and diversity. The budget size is adjusted to be either 50 or 100% of the number of the initially labeled data set, and the number of rounds is limited to 4.

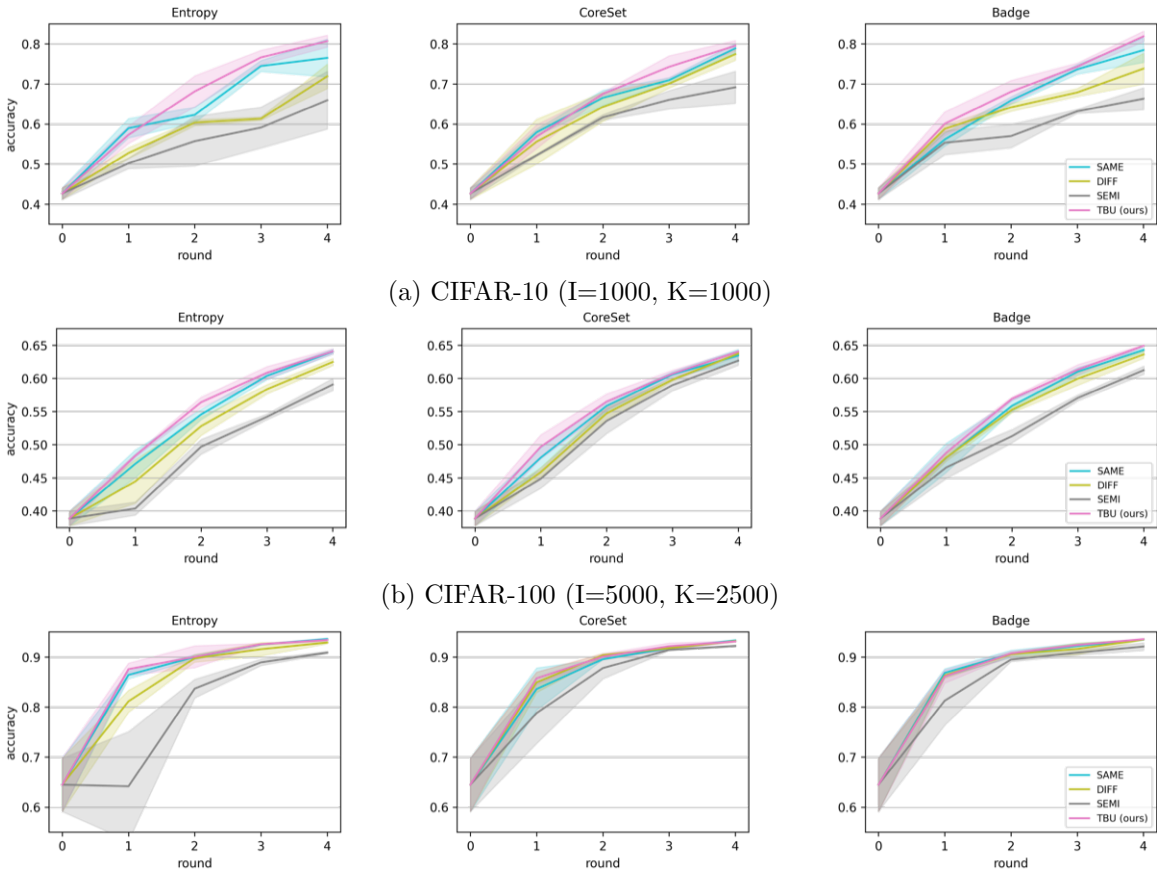
Unless otherwise specified, Wide-ResNet-28-2 is used as a network architecture for the proxy model. When using supervised learning (i.e. DIFF), an SGD optimizer with weight decay 5e-4 is used for training 300 epochs with the learning rate scheduler that multiplies the initial learning rate of 0.1 by 0.5 at the 160-th, 240-th, and 280-th epoch. A batch size of 128 is employed. When using semi-supervised learning (i.e. SEMI, TBU), we mostly adopt the hyper-parameters from Wang et al. (2022b). To note, the total number of epochs is set to 200, where the number of iterations is somewhat reduced than the typical semi-supervised learning. This refers to Coleman et al. (2019) for fast experiments with no significant difference in evaluating active learning algorithms. For fast convergence, a warm-up stage is added in CIFAR-100 and SVHN for 50 epochs of supervised learning with the labeled data where the learning rate is fixed to 0.1.

Following the recommended settings proposed in Munjal et al. (2022); Ji et al. (2023), the target model is built upon ResNet-18 trained by an SGD optimizer with weight decay 5e-4 for 200 epochs. In CIFAR-10/100, the initial learning rate is set to 0.1 and multiplied by 0.1 at the 160-th epoch, while in SVHN, the learning rate is fixed to 0.01. A batch size of 128 is used. For an ablation study in Appendix F, VGG-16 is additionally considered as the target model. Keeping all other conditions identical to ResNet-18, the learning rate is fixed at 0.01 without scheduling.

Appendix F. Additional Experimental Results

In this section, we report (i) per-round accuracy in variations of data sets and target model architectures, (ii) robustness of TBU with respect to input perturbation, (iii) qualitative analysis on the LE and HA instances, and (iv) difficulty scheduling property of TBU which are not included in the main text due to space constraints.

F.1 Comparison of Per-round Accuracy with Variations



(a) CIFAR-10 (I=1000, K=1000)

(b) CIFAR-100 (I=5000, K=2500)

(c) SVHN (I=1000, K=1000)

Figure 4: Per-round accuracy of the target model set to ResNet-18.

We compare the per-round accuracy with variations in data sets and target model architectures. In Figure 4, per-round accuracy in CIFAR-10/100 and SVHN is visualized. We also report the specific values in Table 2 and Table 3 for CIFAR-100 and SVHN. Note that I and K indicate the number of initially labeled data and the budget for each selection round, respectively. Also, in Figure 5, we add an experimental result in CIFAR-10 and SVHN where the target model architecture is set to VGG-16. Note that solid lines indicate the average over three random seeds and the shaded region represents one standard deviation.

		1st	2nd	3rd	4th
Entropy	SAME	0.471 ± 0.021	0.545 ± 0.008	0.604 ± 0.006	0.640 ± 0.004
	DIFF	0.445 ± 0.033	0.528 ± 0.014	0.584 ± 0.007	0.625 ± 0.005
	SEMI	0.404 ± 0.010	0.497 ± 0.011	0.542 ± 0.004	0.591 ± 0.009
	TBU	**0.483 ± 0.002**	**0.564 ± 0.008**	**0.609 ± 0.009**	**0.641 ± 0.004**
CoreSet	SAME	0.481 ± 0.015	0.558 ± 0.007	0.606 ± 0.004	0.635 ± 0.009
	DIFF	0.459 ± 0.007	0.547 ± 0.011	0.598 ± 0.002	0.637 ± 0.001
	SEMI	0.449 ± 0.014	0.536 ± 0.019	0.590 ± 0.009	0.627 ± 0.007
	TBU	**0.497 ± 0.018**	**0.565 ± 0.012**	**0.607 ± 0.005**	**0.640 ± 0.002**
Badge	SAME	0.480 ± 0.022	0.559 ± 0.008	0.610 ± 0.006	0.643 ± 0.003
	DIFF	0.481 ± 0.006	0.553 ± 0.005	0.600 ± 0.010	0.636 ± 0.006
	SEMI	0.466 ± 0.017	0.513 ± 0.010	0.570 ± 0.005	0.612 ± 0.006
	TBU	**0.488 ± 0.009**	**0.569 ± 0.003**	**0.613 ± 0.008**	**0.649 ± 0.001**

Table 2: Per-round accuracy of the target model set to ResNet-18 in CIFAR-100.

		1st	2nd	3rd	4th
Entropy	SAME	0.864 ± 0.008	0.900 ± 0.005	0.925 ± 0.001	**0.935 ± 0.002**
	DIFF	0.811 ± 0.022	0.897 ± 0.008	0.915 ± 0.013	0.929 ± 0.002
	SEMI	0.641 ± 0.109	0.837 ± 0.019	0.889 ± 0.006	0.909 ± 0.002
	TBU	**0.875 ± 0.012**	**0.900 ± 0.022**	**0.925 ± 0.002**	0.934 ± 0.003
CoreSet	SAME	0.836 ± 0.042	0.895 ± 0.001	0.918 ± 0.003	**0.933 ± 0.002**
	DIFF	0.849 ± 0.020	**0.904 ± 0.003**	0.918 ± 0.002	0.932 ± 0.002
	SEMI	0.788 ± 0.061	0.878 ± 0.021	0.914 ± 0.001	0.922 ± 0.002
	TBU	**0.857 ± 0.013**	0.901 ± 0.006	**0.921 ± 0.006**	0.931 ± 0.001
Badge	SAME	**0.868 ± 0.008**	**0.907 ± 0.007**	0.922 ± 0.005	0.934 ± 0.001
	DIFF	0.863 ± 0.006	0.906 ± 0.003	0.916 ± 0.012	0.935 ± 0.001
	SEMI	0.813 ± 0.049	0.895 ± 0.004	0.909 ± 0.005	0.921 ± 0.007
	TBU	0.861 ± 0.013	0.907 ± 0.004	**0.924 ± 0.002**	**0.935 ± 0.002**

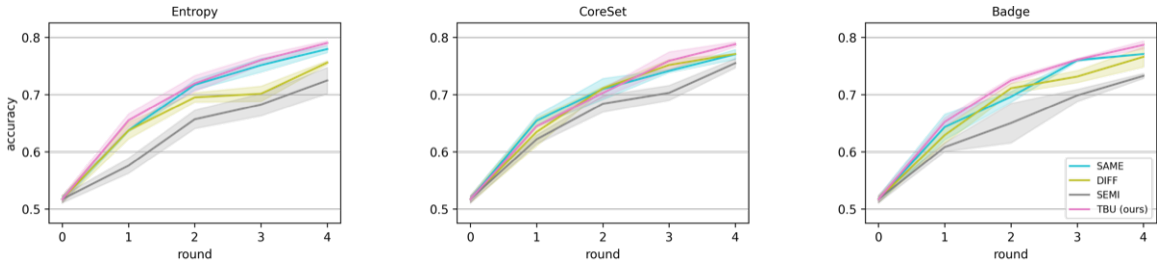
Table 3: Per-round accuracy of the target model set to ResNet-18 in SVHN. Given the same mean values, one with higher variance is marked as bold in preference to the maximal achievable performance.

TBU could outperform most of the baselines and verify its complementary effect on existing active learning algorithms.

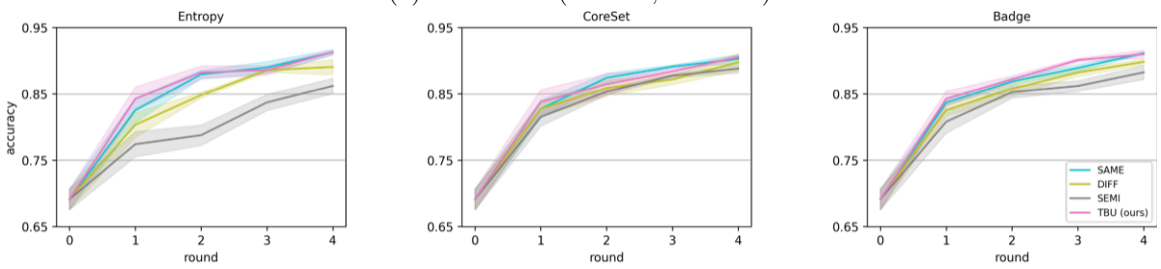
F.2 Robustness of TBU in Corrupted Data Sets

Due to the over-parameterized nature, deep neural networks are vulnerable to over-fitting, leading to wrong predictions while being over-confident. In order to tackle the over-fitting, many regularization techniques have emerged including dropout, batch normalization, and weight decay. However, it has been shown to be not enough in noisy labels (Song et al., 2022) and data set bias (Mehrabi et al., 2021) that are common in real-world data management.

To equip robustness, finding an edge case becomes crucial to cover the under-explored regions of the underlying data distribution. Therefore, a model trained along with a desirable active learning algorithm is expected to be robust to input perturbations that implicitly lead to expanded data distribution.



(a) CIFAR-10 (I=1000, K=1000)



(b) SVHN (I=1000, K=1000)

Figure 5: Per-round accuracy of the target model set to VGG-16.

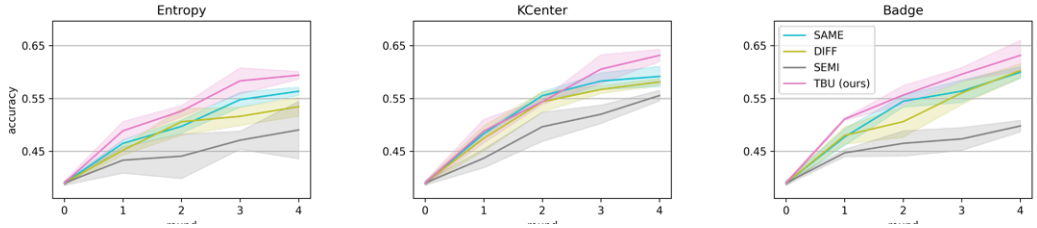
CIFAR-10/100-C (Hendrycks and Dietterich, 2019) is the frequently used data set to verify the robustness of model prediction to input perturbation. Among various options, we chose five different corruption types (gaussian blur, brightness, contrast, fog, and saturate) for generality. In Figure 6 and Figure 7, per-round accuracy evaluated in CIFAR-10/100-C is reported, showing the superiority of TBU.

F.3. Qualitative Analysis on LE and HA instances

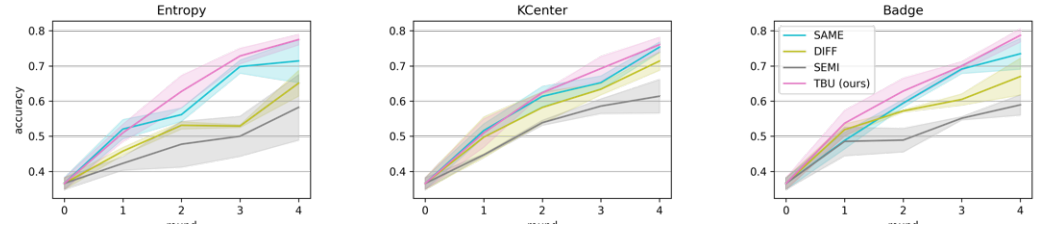
We attach five randomly chosen images from the LE and HA instances in Figure 8 where each row comprises images of the same ground-truth class label. LE instances show stronger semantic similarity in their imagery, while HA instances incorporate unusual patterns like fragmented object components. In short, LE instances can be easily substituted by others, and HA instances can lead to incorrect inductive bias in the model training. Particularly, in the early rounds of the active learning cycle, the labeled data is scarce and more instances in the unlabeled data tend to be assigned to HA instances. In contrast, in the later rounds, LE instances would become dominant in numbers as the model prediction gets more accurate and calibrated.

F.4 Difficulty Scheduling Property of TBU

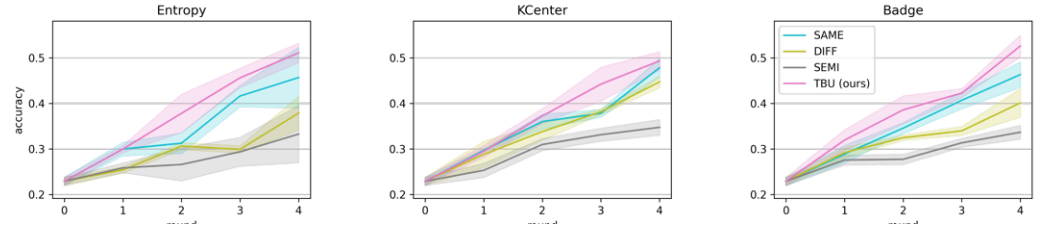
Trivially evident data instances guide the inductive bias about the underlying data distribution. In contrast, edge case data can contribute to model calibration by alleviating overconfidence and robustifying decision boundaries. According to theoretical analysis of training dynamics (Oymak and Soltanolkotabi, 2020; Wang et al., 2022a), easy instances are particularly effective in the early phase of model training, which gradually hands over the attention to hard instances.



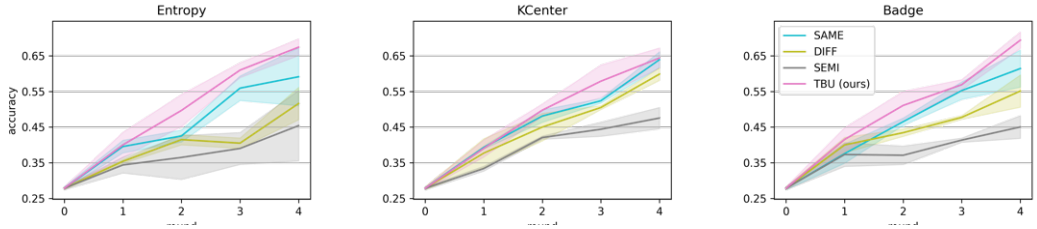
(a) Gaussian blur



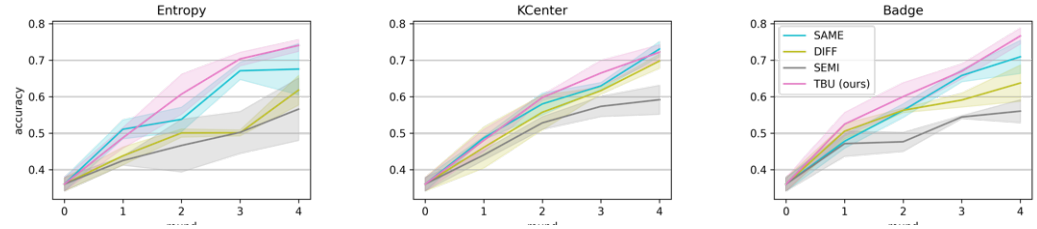
(b) Brightness



(c) Contrast



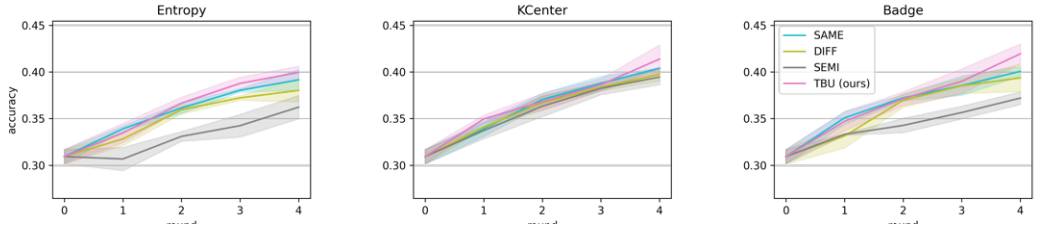
(d) Fog



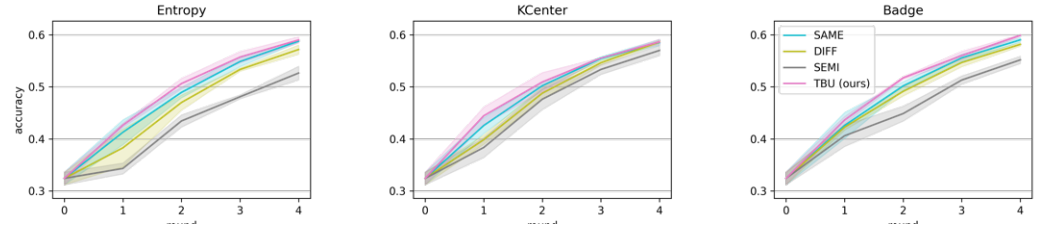
(e) Saturate

Figure 6: Per-round accuracy of the target model set to ResNet-18 in CIFAR-10-C.

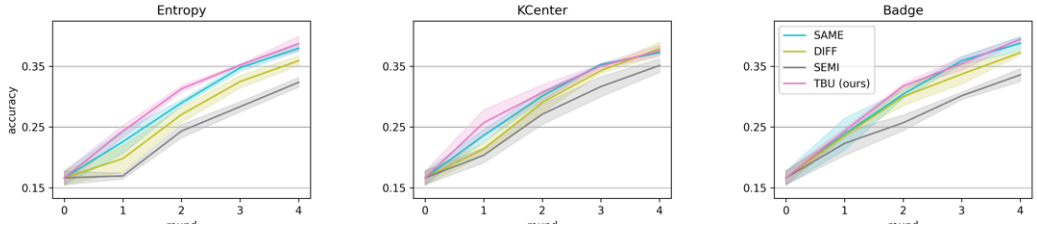
Regarding the LE instances as the easy and the HA instances as the hard, TBU was able to schedule the difficulty of subsets along selection rounds adaptively. In Table 4, the percentage of LE and HA instances in the unlabeled data are reported in CIFAR-10/100



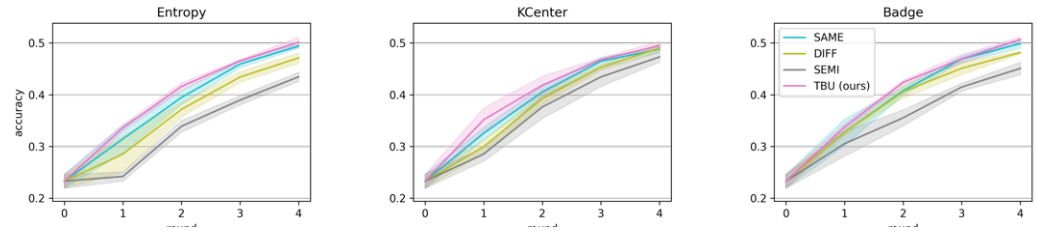
(a) Gaussian blur



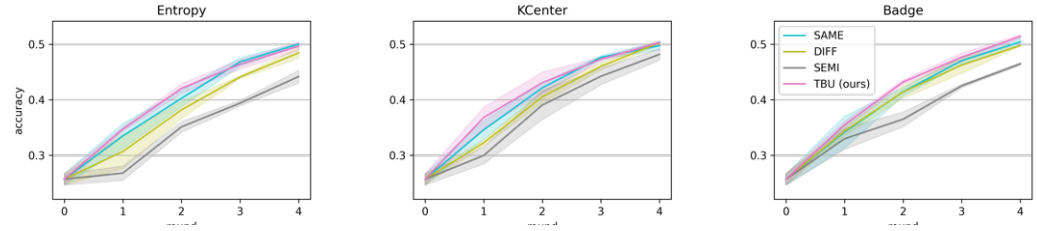
(b) Brightness



(c) Contrast



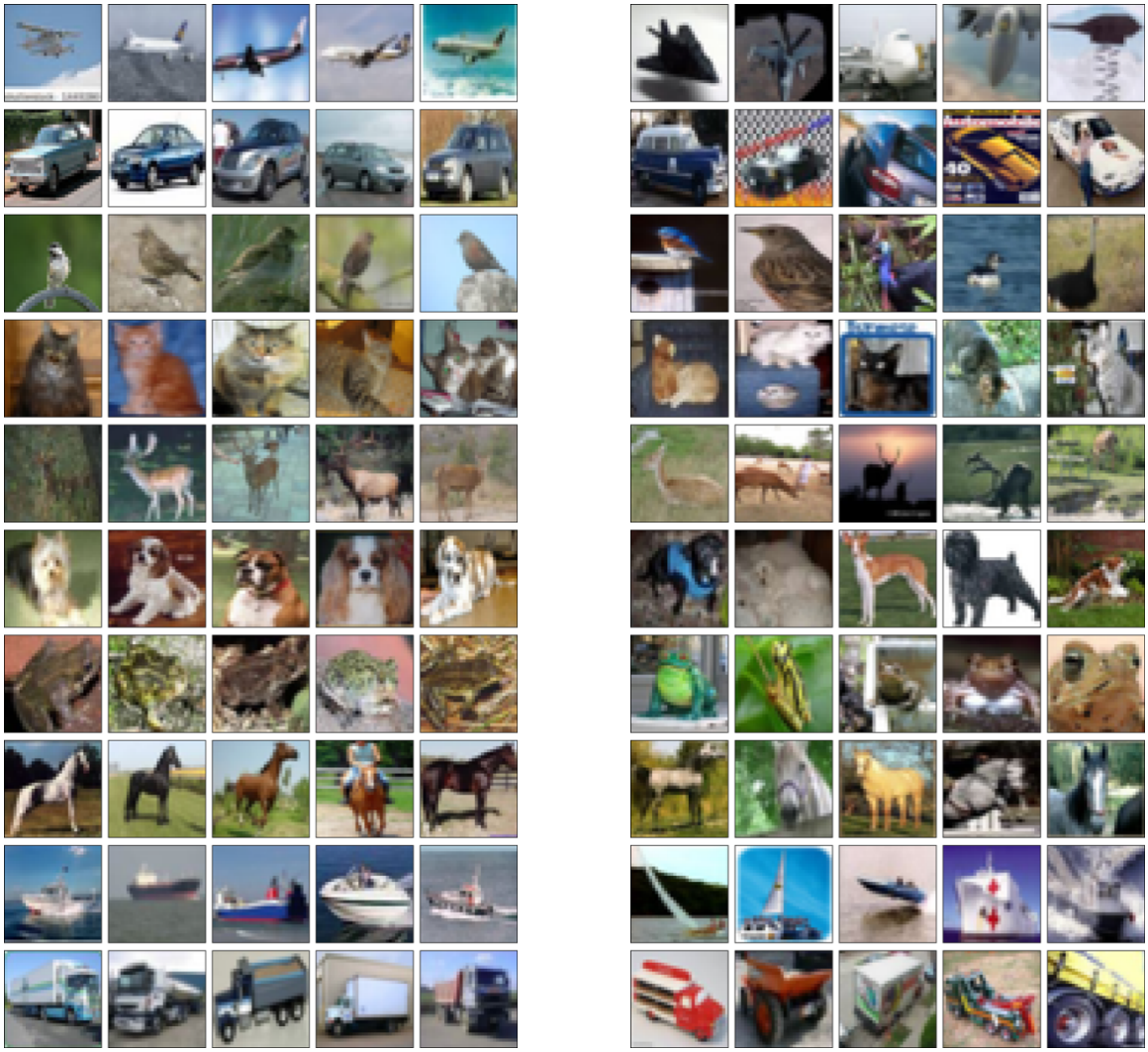
(d) Fog



(e) Saturate

Figure 7: Per-round accuracy of the target model set to ResNet-18 in CIFAR-100-C.

and SVHN. Commonly, as the active learning cycle proceeds, LE instances increase and HA instances decrease. As a result, the remaining instances, neither LE nor HA, can gradually accommodate the instances regarded as HA instances in previous rounds.



(a) LE instances (b) HA instances

Figure 8: Example images of the LE and HA instances from CIFAR-10.

Data	Round	Entropy		CoreSet		Badge	
		LE	HA	LE	HA	LE	HA
CIFAR-10	1st	8.0 ± 1.2	**34.3 ± 6.9**	8.0 ± 1.2	**34.3 ± 6.9**	8.0 ± 1.2	**34.3 ± 6.9**
	2nd	10.4 ± 1.2	12.9 ± 2.2	10.0 ± 0.6	9.3 ± 0.6	10.2 ± 0.5	13.5 ± 1.3
	3rd	13.5 ± 0.8	10.5 ± 1.2	12.0 ± 0.3	9.1 ± 0.7	12.3 ± 0.4	10.3 ± 1.2
	4th	**16.0 ± 0.9**	9.1 ± 1.2	**13.1 ± 1.1**	8.8 ± 0.8	**13.2 ± 0.6**	8.3 ± 1.7
CIFAR-100	1st	5.2 ± 0.3	7.0 ± 0.4	5.2 ± 0.3	7.0 ± 0.4	5.2 ± 0.3	7.0 ± 0.4
	2nd	8.1 ± 0.1	**7.1 ± 0.2**	7.8 ± 0.1	**7.5 ± 0.3**	7.9 ± 0.2	7.0 ± 0.5
	3rd	11.8 ± 0.4	7.0 ± 0.6	10.8 ± 0.6	7.1 ± 0.7	10.9 ± 0.3	**7.1 ± 0.6**
	4th	**15.5 ± 0.4**	5.4 ± 0.6	**12.9 ± 0.2**	6.5 ± 0.1	**14.3 ± 0.5**	5.9 ± 0.2
SVHN	1st	17.7 ± 1.1	**18.0 ± 2.3**	17.7 ± 1.1	**18.0 ± 2.3**	17.7 ± 1.1	**18.0 ± 2.3**
	2nd	28.0 ± 0.6	11.6 ± 0.7	22.8 ± 1.6	9.6 ± 1.1	26.0 ± 1.2	9.2 ± 2.0
	3rd	36.2 ± 0.6	4.2 ± 0.4	28.7 ± 1.5	5.1 ± 0.2	31.3 ± 1.5	5.1 ± 0.8
	4th	**40.7 ± 0.6**	2.1 ± 0.1	**31.9 ± 0.8**	2.7 ± 0.7	**32.5 ± 1.4**	2.3 ± 0.1

Table 4: Per-round percentage of the LE and HA instances in the unlabeled data.