Journal of Data-centric Machine Learning Research (2024) Submitted 2/24; Revised 3/24; Published —

Towards Efficient Active Learning in NLP via Pretrained Representations

Artem Vysogorets* AMV458@NYU.EDU
Center for Data Science
New York University
New York, NY, USA

Achintya Gopal AGOPAL6@BLOOMBERG.COM
Bloomberg LP
New York, NY, USA

Reviewed on OpenReview: *https://openreview.net/forum?id=nEF6d64WYR*

Editor:

Abstract

Fine-tuning Large Language Models (LLMs) is now a common approach for text classification in a wide range of applications. When labeled documents are scarce, active learning helps save annotation efforts but requires retraining of massive models on each acquisition iteration. We drastically expedite this process by using pretrained representations of LLMs within the active learning loop and, once the desired amount of labeled data is acquired, fine-tuning that or even a different pretrained LLM on this labeled data to achieve the best performance. As verified on common text classification benchmarks with pretrained BERT and RoBERTa as the backbone, our strategy yields similar performance to fine-tuning all the way through the active learning loop but is orders of magnitude less computationally expensive. The data acquired with our procedure generalizes across pretrained networks, allowing flexibility in choosing the final model or updating it as newer versions get released.

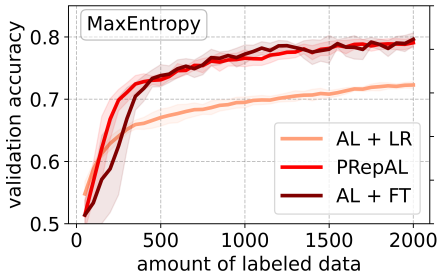
Keywords: Active Learning, Transfer Learning, NLP

1 Introduction

Text classification has a long history and numerous applications in the field of natural language processing (Jindal and Liu, 2007; Lai et al., 2015). Since the debut of Transformers (Vaswani et al., 2017), transfer learning using Large Language Models (LLMs) such as BERT, RoBERTa, and ELECTRA has become increasingly popular among practitioners (Devlin et al., 2018; Liu et al., 2019; Clark et al., 2020). Fine-tuning these models on text classification datasets including GLUE, MultiNLI, and IMDb significantly improved their state-of-the-art performance (Howard and Ruder, 2018; Devlin et al., 2018). However, in many practical scenarios, downstream text datasets are either scarcely labeled or are unlabeled at all, restricting supervised transfer learning. At the same time, manual labeling is often laborious and costly, which calls for a careful and targeted selection of examples for annotation using techniques such as active learning (Schröder and Niekler, 2020). The

∗. Work performed while interning at Bloomberg LP

standard active learning pipeline iterates over the following steps: (1) train a model on the labeled subset of data, (2) query this model to select unlabeled samples for annotation, and (3) label chosen samples and move them to the labeled split (Lewis and Gale, 1994). When used with LLMs, this procedure requires sequentially re-fine-tuning models with up to billions of parameters, which is very expensive, if feasible at all.



Phase	AL+LR	PRepAL	AL+FT
Precomputation (s)	261	261	—
Tot. Retraining (s)	5	5	79K
Tot. Acquisition (s)	356	356	12K
Final training (s)	0.9	7K	7K
Total (s)	0.5K	7.5K	98K
Accuracy (%)	72.27	79.05	79.60

Figure 1: Active learning with MaxEntropy acquisition function and BERT backbone on QNLI across different strategies over 39 labeling iterations. **Left**: validation performance after training on labeled data thus far. Error bands represent ± 1 standard deviation. **Right**: wall-clock time (in seconds) spent on each phase and validation accuracy of the final model trained on $2,000$ acquired samples.

Recently published studies attempt to mitigate this problem by either using just a single active learning iteration, introducing additional proxy-models, or carefully synchronizing model retraining and human labeling (Xie et al., 2021; Sanh et al., 2019; Nguyen et al., 2022). These methods, however, are either impractical, lack in performance, require considerable memory overhead, or suffer from all of the above. In this work, we propose an alternative approach to transfer learning with active learning that brings extraordinary speedups while avoiding these deficiencies.

Our method. We introduce *PRepAL (Pretrained Representation Active Learning)*, which precomputes data representations using a pretrained LLM and, on each active learning (AL) iteration, fits a simple linear classifier, e.g., Logistic Regression (LR), avoiding resource-consuming fine-tuning (FT) until the desired amount of data is labeled. This simple procedure yields surprisingly competitive performance with negligible additional resources and time per active learning iteration. For example, fitting Logistic Regression on $2,000$ QNLI representations extracted from a pretrained BERT model takes 0.2 seconds, whereas fine-tuning the entire BERT model takes nearly two hours. Thus, total time for model retraining in the 39-step active learning procedure in Figure 1 is just over 5 seconds with PRepAL, which is three orders of magnitude less than with standard fine-tuning (5 seconds vs. 79K seconds or 22 hours). In practice, this speedup helps avoid costly delays between active learning iterations associated with model retraining, allowing human annotators to complete all labeling in one sitting. At the same time, the quality of annotated data remains high: $2,000$ labeled QNLI samples acquired through PRepAL with MaxEntropy scoring function show only 0.55% drop in validation accuracy of the final fine-tuned BERT model compared to those selected with MaxEntropy and fine-tuning. The efficient retraining routine of PRepAL allows for sequential data labeling as opposed to batching and, as our

experiments in Section 4 demonstrate, this can further improve data quality in the early stages of active learning. The data acquired by PRepAL using one LLM as a backbone can successfully fine-tune a different pretrained LLM, as our experiments with BERT and RoBERTa indicate; this transferability allows switching between final model architectures without rerunning active learning. PRepAL can operate with virtually any acquisition function and, hence, is a general mechanism that improves the efficiency of active learning.

Contents. The remainder of the paper is organized as follows: in Section 2, we discuss related works. Section 3 gives a detailed description of PRepAL. Section 4 showcases its performance on classic text classification datasets in conjuction with different active learning methods. Sections 5 and 6 close the paper with discussion on PRepAL, its strengths and limitations, and offer avenues for future work.

2 Related Work

The surge of interest in active learning over the past few decades inspired a wealth of literature surveys (Settles, 2009; Fu et al., 2013; Aggarwal et al., 2014; Zhan et al., 2022). In Appendix A, we follow a recent taxonomy by Schröder and Niekler (2020) designed specifically for deep learning and describe relevant active learning methods in detail.

Active learning with proxy models. Akin to our study, a handful of works are concerned with accelerating the active learning routine common to all of the algorithms presented in Appendix A. Xie et al. (2021) utilize pretrained feature embeddings for one-shot label querying; however, this method is inferior to many existing baselines that iteratively retrain classifiers on newly labeled data and was evaluated on images only. Shelmanov et al. (2021) retrain a less bulky proxy model—DistillBERT (Sanh et al., 2019)—within the active learning loop and fine-tune BERT once on the final labeled dataset. This algorithm exhibits a discrepancy between BERT performance on the approximate and the baseline labeled datasets while bringing only marginal computational savings since DistillBERT has just 40% less parameters than BERT itself. Coleman et al. (2020) adhere to a similar approach by reducing the width and depth of the full models, which again does not enjoy nearly as much compute efficiency as our method. Like us, Jiao et al. (2021) use Logistic Regression on top of embeddings extracted from a pretrained model during data labeling but apply their method in a very specific medical imaging domain and only with entropy-based acquisition functions. Nguyen et al. (2022) resort to retraining both the main LLM and the proxy MiniLM (Wang et al., 2020) on each active learning iteration but do so in parallel with the current annotation step to save time. This method is even more computationally expensive than the original active learning routine and requires accurate synchronization between labeling and training to realize potential speedups. In contrast, our approach is general, conceptually simple, requires no additional proxy models, computationally cheap on each iteration, and incurs only negligible performance drop, if any (Section 4).

3 Method

As discussed in Section 2, previous works have used proxy models for data acquisition during active learning, hoping that labeled subsets transfer to more powerful but less efficient

training strategies like fine-tuning LLMs. Our method, PRepAL, follows the same paradigm but is simpler and more efficient. Given a pretrained LLM $\hat{\Phi}$ as the backbone, our method precomputes associated data representations $\hat{\Phi}(X)$ and uses them within the notoriously costly active learning loop, refitting a single-layer classifier ψ (e.g., Logistic Regression) on each active learning iteration (Algorithm 1). PRepAL makes AL iterations magnitudes more affordable and avoids unwanted delays caused by lengthy model retraining. For example, the total time spent retraining Logistic Regression on precomputed representations of labeled QNLI samples over 39 active learning iterations is just five seconds, while all 39 re-fine-tuning cycles take almost 22 hours (Figure 1). Most importantly, data labeled using PRepAL can be effectively used to fine-tune that same backbone LLM, achieving the best of both worlds: high efficiency and competitive performance.

Algorithm 1: PRepAL

Input: Acquisition function A, active learning iterations T, acquisition batch size b, classification dataset $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$, initially labeled indices $I_0 \subset [n]$, pretrained LLM $\hat{\Phi}$, classifier ψ, loss $\mathcal{L}$.;

Precompute representations: $\tilde{X} \leftarrow \hat{\Phi}(X)$;

for $t \in [T]$ **do**

$\psi_t \leftarrow \text{argmin}_\psi \mathcal{L}(\psi(\tilde{X}_{I_{t-1}}), Y_{I_{t-1}})$;

$I \leftarrow \text{Top}_b\{i \notin I_{t-1}, \text{key} = A(X_i, \psi_t)\}$;

$I_t \leftarrow I_{t-1} \cup I$;

end

Fine-tune LLM:

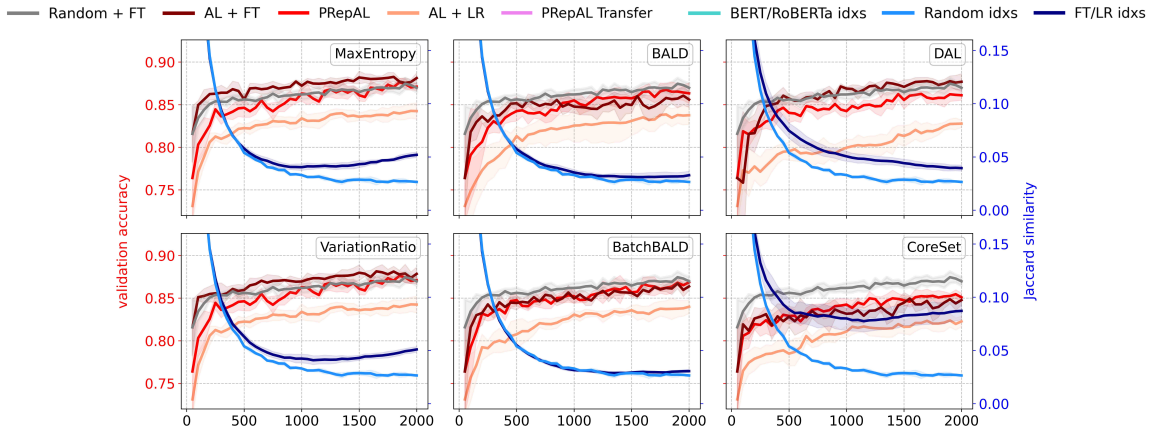
$\mu \leftarrow \text{argmin}_{\psi \circ \Phi} \mathcal{L}(\psi(\Phi(X_{I_T})), Y_{I_T})$;

Result: μ

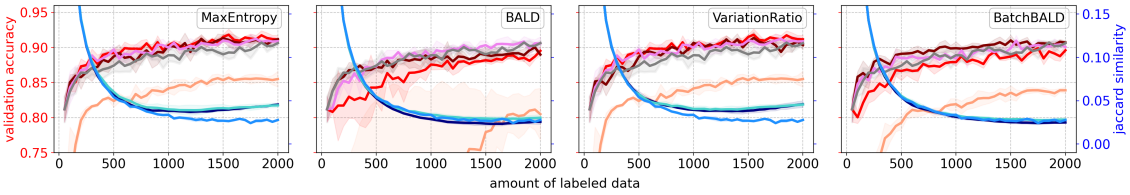
PRepAL can be applied in conjunction with virtually any acquisition function; however, not all of them fit its simplified retraining procedure equally well. Acquisition methods that operate on the representation space of the trained model, e.g., CoreSet and DAL, exhibit larger performance gaps when trained on data acquired through PRepAL compared to active learning with re-fine-tuning. These algorithms assume that feature extractors change as a result of retraining with more labeled data, which does not happen with the standard PRepAL pipeline where data representations are precomputed once and used for all active learning iterations. In principle, however, we can vary complexity of the model ψ (e.g., by adding hidden layers) that is retrained on static features $\tilde{X}$, trading off the increase in required resources for the dynamic embedding space and higher acquisition quality. Thus, PRepAL is not limited to any particular backbone LLM, classifier type or acquisition function; rather, it describes a flexible and universal strategy of using pretrained representations for more efficient active learning where fine-tuning pretrained models is state-of-the-art.

4 Experiments

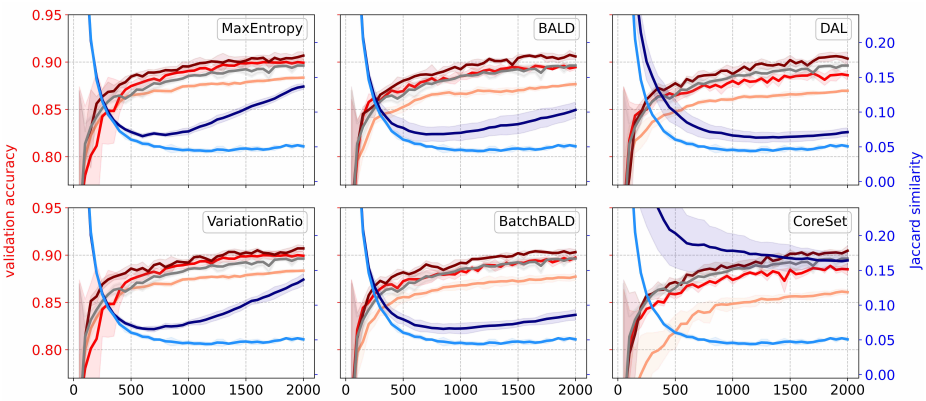
In this section, we present experimental results that benchmark several active learning methods for text classification using pretrained BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) to evaluate PRepAL in an extensive ablation study. We perform our evaluation across 8 different acquisition functions: Random, MaxEntropy, VariationRatio, BALD, BatchBALD, DAL, CoreSet (greedy), and EGL (see Appendix A for details) on 5 common text classification datasets: SST-2, IMDb, CoLa, QNLI, and AG News (see



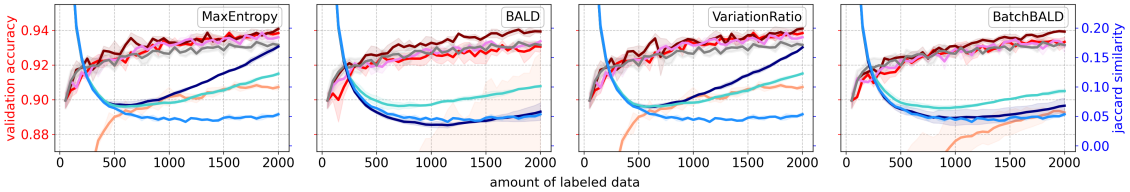
(a) BERT backbone on SST-2 dataset



(b) RoBERTa backbone on SST-2 dataset



(c) BERT backbone on IMDb dataset



(d) RoBERTa backbone on IMDb dataset

Figure 2: The red-toned curves and the grey curve show the validation accuracy of different models with different active learning protocols. The blue-toned curves indicate Jaccard similarity between subsets of data indices selected by different active learning protocols and the data indices selected by AL+FT. Error bands represent ±1 standard deviation.

Appendix B for implementation details). This section contains results for the first two datasets only, while the other three can be found in Appendix C.

Active learning protocols. For each dataset and acquisition function pair, we implement three active learning strategies—AL+LR, PRepAL, and AL+FT—that differ in the model retraining procedure during and after active learning. In particular, AL+LR and PRepAL refit a Logistic Regression model on each data labeling iteration, while AL+FT re-fine-tunes the entire LLM from its original pretrained parameters together with a linear classification head. Thus, discounting for randomness, data selected with AL+LR and PRepAL must be identically the same. When the desired amount of data is acquired, AL+LR still fits a Logistic Regression while PRepAL and AL+FT fine-tune the original pretrained LLM. The validation accuracy of these final models is recorded in Figure 2. Figures 2b and 2d show how well data acquired by refitting a Logistic Regression model on top of BERT representations transfers to fine-tune a RoBERTa model (PRepAL Transfer).

Main results. In general, PRepAL incurs minimal, if any, accuracy drops compared to AL+FT. When used with the most consistent and successful acquisition functions such as MaxEntropy and VariationRatio, PRepAL tends to close this performance gap as more data becomes available. AL+LR trails the other two strategies by quite a margin, suggesting that investing resources in post-labeling fine-tuning is essential to achieve best performance. We observe that random acquisition is a strong baseline in our setup, beating AL+FT with BALD and BatchBALD acquisition functions on several occasions (Figure 2a). Still, PRepAL is almost always better than random labeling (Table 1).

Selected data. Jaccard similarity shows a considerable overlap between data indices selected for labeling through querying Logistic Regression models and those obtained via fine-tuning entire backbone models, which is especially pronounced with MaxEntropy and VariationRatio (the blue-toned curves in Figure 2). Most importantly, in these cases, Jaccard similarity grows steadily with active learning iterations, indicating that PRepAL and AL+FT consistently select similar samples for labeling, which contributes to their even performance. While DAL and CoreSet also exhibit higher than random Jaccard similarity

Table 1: Validation accuracy (mean±std, in %) of the final models fine-tuned on $2,000$ labeled samples. Accuracy above random labeling is shown in bold.

Algorithm	Protocol	BERT		RoBERTa	
		SST-2	IMDb	SST-2	IMDb
Random	AL+FT	86.9 ± 0.5	89.6 ± 0.1	90.6 ± 0.3	93.1 ± 0.1
MaxEntropy	AL+LR	84.2 ± 0.9	88.3 ± 0.1	85.5 ± 1.1	90.7 ± 0.2
	PRepAL	$\mathbf{87.1 \pm 0.4}$	$\mathbf{89.9 \pm 0.4}$	$\mathbf{91.1 \pm 0.5}$	$\mathbf{93.8 \pm 0.2}$
	AL+FT	$\mathbf{88.1 \pm 0.6}$	$\mathbf{90.7 \pm 0.4}$	$\mathbf{91.1 \pm 0.1}$	$\mathbf{94.1 \pm 0.2}$
	Transfer	—	—	$\mathbf{90.9 \pm 0.4}$	$\mathbf{93.6 \pm 0.1}$
BatchBALD	AL+LR	84.0 ± 0.7	87.7 ± 0.4	83.9 ± 0.4	89.3 ± 0.2
	PRepAL	$\mathbf{86.9 \pm 0.2}$	$\mathbf{89.7 \pm 0.2}$	89.6 ± 0.5	$\mathbf{93.3 \pm 0.3}$
	AL+FT	86.3 ± 0.5	$\mathbf{90.3 \pm 0.3}$	$\mathbf{90.7 \pm 0.2}$	$\mathbf{93.9 \pm 0.1}$
	Transfer	—	—	90.1 ± 0.6	$\mathbf{93.1 \pm 0.4}$

between PRepAL and AL+FT indices, it is only observed over initial iterations and shrinks with more labeled data. This phenomenon may be an artifact of PRepAL's immutable data representations, which these two acquisition methods heavily rely on. We hypothesize that BERT's representation space does not change as much in the beginning of active learning, causing larger overlap between indices selected by PRepAL and by AL+FT in the first few iterations of DAL and CoreSet.

Transferability across models. Figures 2b and 2d show that data acquired by PRepAL with BERT can be successfully transferred to fine-tune a pretrained RoBERTa model (PRepAL Transfer). In fact, across all five datasets and training sizes, it performs no worse than PRepAL with RoBERTa backbone itself, sometimes even surpassing AL+FT (BALD and BatchBALD in Figure 2b). PRepAL indices transferred from BERT have considerable overlap with those selected by AL+FT using RoBERTa; interestingly, this overlap can be even higher than for PRepAL with RoBERTa itself (Figure 2d). These observations offer flexibility to choose the final model architecture after PRepAL is initially used. As updated versions of popular large pretrained models are released, one is not required to rerun PRepAL with the new backbone but can instead reuse data labeled previously using a different LLM and achieve a commensurate accuracy.

Reducing the batch size. One fundamental culprit of most score-based active learning procedures is the lack of diversity in the acquired data due to batching (Guo and Schuurmans, 2007; Beatty et al., 2018). Reducing the acquisition batch size often improves model performance (Brinker, 2003). However, the extreme costs associated with retraining models urge practitioners to increase the number of samples acquired on every iteration, sacrificing diversity and, hence, quality of the data. In response, recent works design a variety of algorithms to mitigate the negative consequences associated with large batch acquisition (Kirsch et al., 2019; Tan et al., 2021; Citovsky et al., 2021). The resource efficiency of retraining with PRepAL, on the other hand, allows us to abandon batching whatsoever and acquire training samples one by one—an unthinkable luxury for any other active learning procedure. In Figure 3, we compare the performance of RoBERTa fine-tuned on data acquired using batch sizes of 50 and 1 sample(s) per each of 39 and 1950 iterations, respectively. Interestingly, we observe that using sequential labeling improves the ultimate model only in the beginning and when labeled data is still limited. This may indicate that, while the

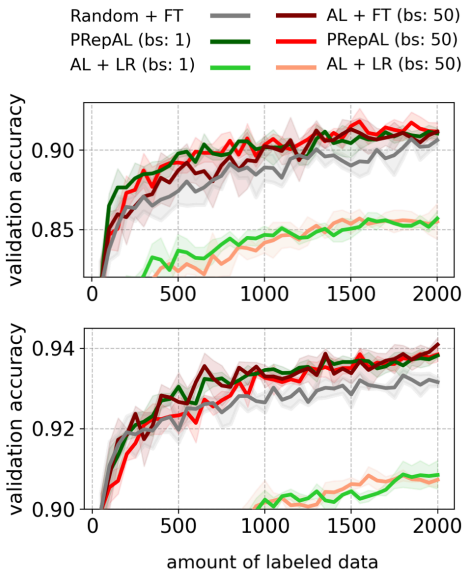


Figure 3: Validation accuracy of different active learning protocols based on MaxEntropy acquisition function that use a batch size (bs) of 50 samples per iteration (red-toned curves), or 1 sample per iteration (green-toned curves). **Top:** RoBERTa on SST-2; **Bottom:** RoBERTa on IMDb.

diversity within each individual batch is poor, different batches at different iterations are diverse enough to match the quality of the data acquired one by one.

5 Discussion

Motivated to reduce the computational costs of active learning when fine-tuning massive models such as LLMs is state-of-the-art, we propose PRepAL—a universal active learning protocol for quick and memory-efficient acquisition of high-quality data by leveraging pretrained representation spaces. Our method precomputes fixed representations of the unlabeled data using a pretrained LLM and retrains a linear classifier on each active learning iteration in seconds, avoiding unwanted delays between labeling phases. Finally, PRepAL fine-tunes the original LLM on the ultimate labeled data to reach best performance.

We empirically confirmed the effectiveness of our method using pretrained BERT and RoBERTa models across a variety of text classification datasets and active learning functions. As a byproduct, we benchmarked seven pool-based acquisition methods and found simple uncertainty-based scoring functions like MaxEntropy and VariationRatio to be particularly successful and consistent in this domain. Conveniently, PRepAL is most effective when used together with these functions, showing little performance drop compared to a more laborious data acquisition procedure (AL+FT), in which the entire LLM is re-fine-tuned on each active learning iteration. Fitting just a linear classifier on every active learning iteration allows labeling data points sequentially and not in batches, offering an improvement in quality of the data during the early stages of the active learning loop. The data acquired by the Logistic Regression model not only transfers to fine-tune the original pre-trained backbone architecture but also to other, potentially more advanced models as demonstrated by our experiments with BERT and RoBERTa.

6 Limitations & Future Research

We close the paper by discussing the limitations of our method and sketching the directions for future research. As mentioned in Section 3, not all acquisition functions work equally well with PRepAL. Some methods, like DAL and CoreSet, are more sensitive to having accurate representation spaces, which remain fixed throughout our active learning protocol. On the other hand, in Section 4 we found these algorithms inferior to other simpler baselines (e.g., MaxEntropy), for which PRepAL matches with its more sophisticated rival AL+FT in terms of the final model performance. In addition, it is trivial to modify our procedure to obtain dynamic representation spaces of retrained models by stacking hidden layers in the classifier attached to BERT and using them for feature extraction. It might be interesting to test whether this adjustment will lead to better performance for acquisition methods like DAL and CoreSet. Further research may explore how viable PRepAL is for other types of downstream tasks such as machine translation or even for applications in computer vision, where fine-tuning deep convolutional networks or Visual Transformers has become a popular practice (Huh et al., 2016; Dosovitskiy et al., 2021; Morid et al., 2021). Finally, the flying speed of retraining with PRepAL opens an opportunity to compare batch mode active learning with sequential labeling, which can potentially reveal how exactly the acquisition size impacts data diversity, quality, and the ultimate model performance.

7 Acknowledgements

The first author was partly supported by the NSF Award 1922658.

8 Reproducibility Statement

While the code is proprietary, we have made every effort to facilitate the reproduction of our empirical results. Our proposed algorithm is simple and thoroughly documented in Section 3, which includes pseudo-code (Algorithm 1). A rigorous description of all acquisition functions used for benchmarking is presented in Appendix A. In Appendix B, we precisely describe our datasets (Table 2) and the programming tools used in preparation of this work. In the same section, we list all essential hyper-parameters and motivate our choices by citing a study that inspired them. Lastly, Tables 3 and 4 report the exact estimates of model accuracy to facilitate cross-checking of the reproduced results.

References

Charu C Aggarwal, Xiangnan Kong, Quanquan Gu, Jiawei Han, and S Yu Philip. Active learning: A survey. In *Data Classification*, pages 599–634. Chapman and Hall/CRC, 2014.

Garrett Beatty, Ethan Kochis, and Michael Bloodgood. Impact of batch size on stopping active learning for text classification. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*. IEEE, January 2018.

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.

Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, page 59–66. AAAI Press, 2003. ISBN 1577351894.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. *Advances in Neural Information Processing Systems*, 34:11933–11944, 2021.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*, 2020.

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. In *ICLR 2020*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Liat Ein Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. Active learning for bert: an empirical study. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7949–7962, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Melanie Ducoffe and Frederic Precioso. Adversarial active learning for deep networks: a margin based approach. *arXiv preprint arXiv:1802.09841*, 2018.

William Falcon and The PyTorch Lightning team. PyTorch Lightning, 3 2019. URL `https://github.com/Lightning-AI/lightning`.

Linton C. Freeman. *Elementary applied statistics: for students in behavioral science*. Wiley, New York, 1965.

Yifan Fu, Xingquan Zhu, and Bin Li. A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283, 2013.

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.

Daniel Gissin and Shai Shalev-Shwartz. Discriminative active learning. *arXiv preprint arXiv:1907.06347*, 2019.

Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

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

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

Jiaji Huang, Rewon Child, Vinay Rao, Hairong Liu, Sanjeev Satheesh, and Adam Coates. Active learning for speech recognition: the power of gradients. *CoRR*, abs/1612.03226, 2016.

Minyoung Huh, Pulkit Agrawal, and Alexei Efros. What makes imagenet good for transfer learning? 08 2016.

Yiping Jiao, Jie Yuan, Yong Qiang, and Shumin Fei. Deep embeddings and logistic regression for rapid active learning in histopathological images. *Computer Methods and Programs in Biomedicine*, 212:106464, 2021.

Nitin Jindal and Bing Liu. Review spam detection. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, page 1189–1190, New York, NY, USA, 2007. Association for Computing Machinery.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

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.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI Conference on Artificial Intelligence*, 2015.

David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In Bruce W. Croft and C. J. van Rijsbergen, editors, *SIGIR '94*, pages 3–12, London, 1994. Springer London. ISBN 978-1-4471-2099-5.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

Mohammad Amin Morid, Alireza Borjali, and Guilherme Del Fiol. A scoping review of transfer learning research on medical image analysis using imagenet. *Computers in Biology and Medicine*, 128:104115, 2021. ISSN 0010-4825.

Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, page 79, 2004.

Minh Van Nguyen, Nghia Ngo, Bonan Min, and Thien Nguyen. FAMIE: A fast active learning framework for multilingual information extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, pages 131–139, Hybrid: Seattle, Washington + Online, July 2022. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

Fábio Perez, Rémi Lebret, and Karl Aberer. Cluster-based active learning. *arXiv preprint arXiv:1812.11780*, 2018.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Christopher Schröder and Andreas Niekler. A survey of active learning for text classification using deep neural networks, 2020.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR (Poster)*, 2018.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. *Advances in neural information processing systems*, 20, 2007.

Artem Shelmanov, Dmitri Puzyrev, Lyubov Kupriyanova, Denis Belyakov, Daniil Larionov, Nikita Khromov, Olga Kozlova, Ekaterina Artemova, Dmitry V Dylov, and Alexander Panchenko. Active learning for sequence tagging with deep pre-trained models and bayesian uncertainty estimates. *arXiv preprint arXiv:2101.08133*, 2021.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

Wei Tan, Lan Du, and Wray Buntine. Diversity enhanced active learning with strictly proper scoring rules. *Advances in Neural Information Processing Systems*, 34:10906–10918, 2021.

Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019. In the Proceedings of ICLR.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788, 2020.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*, 2018.

Yichen Xie, Masayoshi Tomizuka, and Wei Zhan. Towards general and efficient active learning. *arXiv preprint arXiv:2112.07963*, 2021.

Minjie Xu and Gary Kazantsev. Understanding goal-oriented active learning via influence functions. *arXiv preprint arXiv:1905.13183*, 2019.

Zhao Xu, Kai Yu, Volker Tresp, Xiaowei Xu, and Jizhi Wang. Representative sampling for text classification using support vector machines. In *European Conference on Information Retrieval*, pages 393–407. Springer, 2003.

Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.

Ye Zhang, Matthew Lease, and Byron Wallace. Active discriminative text representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Appendix A. Active Learning Algorithms: Literature Review

In this section, we follow a recent taxonomy by Schröder and Niekler (2020) designed specifically for deep learning and describe relevant active learning methods in more detail. Like Schröder and Niekler (2020) and Zhan et al. (2022), we focus on pool-based active learners as they are the most prevalent and are natural for text classification tasks. Active learning algorithms of this type have access to the entire pool of unlabeled data and make decisions via ranking samples with an ad-hoc acquisition function $A(x)$ (Xie et al., 2021).

Data-based. The methods in this category focus on the unlabeled data itself and are the most model-agnostic. Designed primarily for convolutional neural networks, CoreSet by Sener and Savarese (2018) acquires unlabeled data in a greedy manner as to best cover the dataset manifold within the representation space. That is, CoreSet selects instances that maximize the acquisition function $A(x) = \min_{x_j \in L} \|\Phi(x) - \Phi(x_j)\|_2$ where L is the current labeled dataset and where Φ is the current embedding mapping. A handful of methods enforce representativeness of selected samples through clustering. Nguyen and Smeulders (2004) use K-medoid clustering for sample density estimation; Xu et al. (2003) run K-means within the SVM margin and send cluster centroids for annotation. Perez et al. (2018) send entire clusters for inspection and labeling by a human.

Model-based. These methods rely on features of the learner. Settles et al. (2007) was first to use expected norm of the loss gradient with respect to learner's parameters to assess the potential influence of any unlabeled sample on training. This algorithm and its close adaptations are commonly referred to as Expected Gradient Length (EGL). Huang et al. (2016) apply EGL for speech recognition and discuss it from a variance reduction perspective. Formally, the acquisition function of EGL is $A(x) = \mathbb{E}_{\hat{y} \sim \hat{p}(y|x)} \|\nabla_\theta \ell(x, \hat{y}, \hat{\theta})\|_2^2$

where ℓ is the loss function, $\hat{\theta}$ are the current model parameters, and $\hat{p}(y|x)$ is the estimate of class probabilities at the unlabeled sample x. Applying convolutional neural networks for text classification, Zhang et al. (2017) score unlabeled samples by the length of the embedding space update weighted by the current class probability estimates as above. Tong and Koller (2001) take a margin-based approach and choose unlabeled points that lie closest to the decision hyperplane of SVM. Ducoffe and Precioso (2018) extend this idea to deep neural networks by choosing adversarial examples instead. Gissin and Shalev-Shwartz (2019) develop Discriminative Active Learning (DAL); they fit a separate classifier on the learned representations of the data that discriminates between labeled and unlabeled instances, and acquire those predicted unlabeled with higher confidence. Xu and Kazantsev (2019) introduce Goal-Oriented Active Learning (GORAL), which uses influence functions to approximate utility of labeling any datapoint with respect to a particular objective, e.g., negative validation loss or negative prediction entropy.

Prediction-based. The algorithms in this subclass utilize predictions of the current learner (ensemble of learners) to guide acquisition. A large body of studies choose to label samples with the maximum uncertainty as expressed by the model. In the context of text classification, Lewis and Gale (1994) measure uncertainty as entropy of the current class probabilities, i.e., $A(x) = \mathbb{H}(\hat{p}(y|x))$. Beluch et al. (2018) find that variation ratio $A(x) = 1 - \max_i \hat{p}(y_i|x)$, originally introduced by Freeman (1965), is competitive for active learning in image classification. Houlsby et al. (2011) propose Bayesian Active Learning via Disagreement (BALD) and estimate uncertainty as the mutual information between the predictions and the parameters of a Bayesian model, which they reformulate as $A(x) = \mathbb{H}(y|x) - \mathbb{E}_{p(\omega)}\mathbb{H}(y|\omega, x)$. Gal et al. (2017) model the posterior $p(\omega)$ as the dropout distribution for BALD when applied in the image classification domain. BALD may overestimate the mutual information between a batch of unlabeled samples and model parameters, making it less effective in batch-mode acquisition. Accounting for this shortcoming, Kirsch et al. (2019) introduce BatchBALD that scores selected points jointly with $A(X) = \mathbb{H}(Y) - \mathbb{E}_{p(\omega)}\mathbb{H}(Y|\omega, X)$ where $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$.

Appendix B. Experimental Details

In this section, we report the details of our implementation of the experiments presented in Section 4. Following Devlin et al. (2018), we use the Adam optimizer (Kingma and Ba, 2015) with cross-entropy criterion for 3 training epochs, batch size of 16, dropout rate of 0.1, early stopping, no weight decay, and 1e-6 and 2e-5 as learning rates of the backbone and the linear classifier, respectively. Each experiment starts with 50 randomly chosen labeled samples and acquires 50 more on each of the 39 subsequent AL iterations. Precomputed data representations are extracted from the last hidden state of the backbone LLM (`bert-base-uncased` or `roberta-base-cased`) and are reduced using average pooling (we tested other feature extraction setups but preliminary runs showed little dependence on these hyperparameters). We run EGL only on the smallest dataset (CoLa) due to time constraints (Figure 5). All experiments were set up in PyTorch (Paszke et al., 2019) and PyTorch Lightning (Falcon and The PyTorch Lightning team, 2019), run on an internal cluster with Tesla V100-SXM2-32GB GPUs installed, and repeated across five random seeds, taking approximately $12,500$ GPU hours total.

Table 2: Text classification datasets used in this study. The number of classes is denoted by K. AG News was downsampled to 12,500 documents per class due to limited resources and time.

Dataset	Task	Size	K	Prior	Source	License
SST-2	sentiment analysis	67,349	2	56%	Socher et al. (2013)	CC0
CoLa	acceptability	8,551	2	70%	Warstadt et al. (2018)	CC0
QNLI	question-answering	102,671	2	50%	Wang et al. (2019)	CC-BY-SA 4.0
IMDb	sentiment analysis	25,000	2	50%	Maas et al. (2011)	—
AG News	categorization	50,000	4	Unif.	Zhang et al. (2015)	—

Appendix C. Additional Experiments

Figure 4 depicts relative performance of different active learners across datasets and retraining procedures. A similar study was carried out by Dor et al. (2020) and, although their setup is slightly different, their results match ours where comparison is appropriate. In particular, we observe that the simplest uncertainty-based acquisition functions—MaxEntropy and VariationRatio—perform consistently well and are superior to other methods in most scenarios. Among others, only DAL consistently outperforms random acquisition when used together with the most powerful active learning protocol, AL+FT. Note, on the other hand, that DAL and especially CoreSet lose to Random in practically all setups when Logistic Regression is used for querying samples for acquisition. Indeed, as discussed in Section 3, these methods are not expected to shine when data representations remain unchanged across labeling iterations, as is the case with AL+LR and PRepAL, which share the same acquisition strategy. Figure 4 shows that it is crucial to fine-tune the final LLM to achieve best performance, regardless of the quantity of available labeled data and of the way it was obtained. This is more clearly shown in Figure 2 in Section 4.

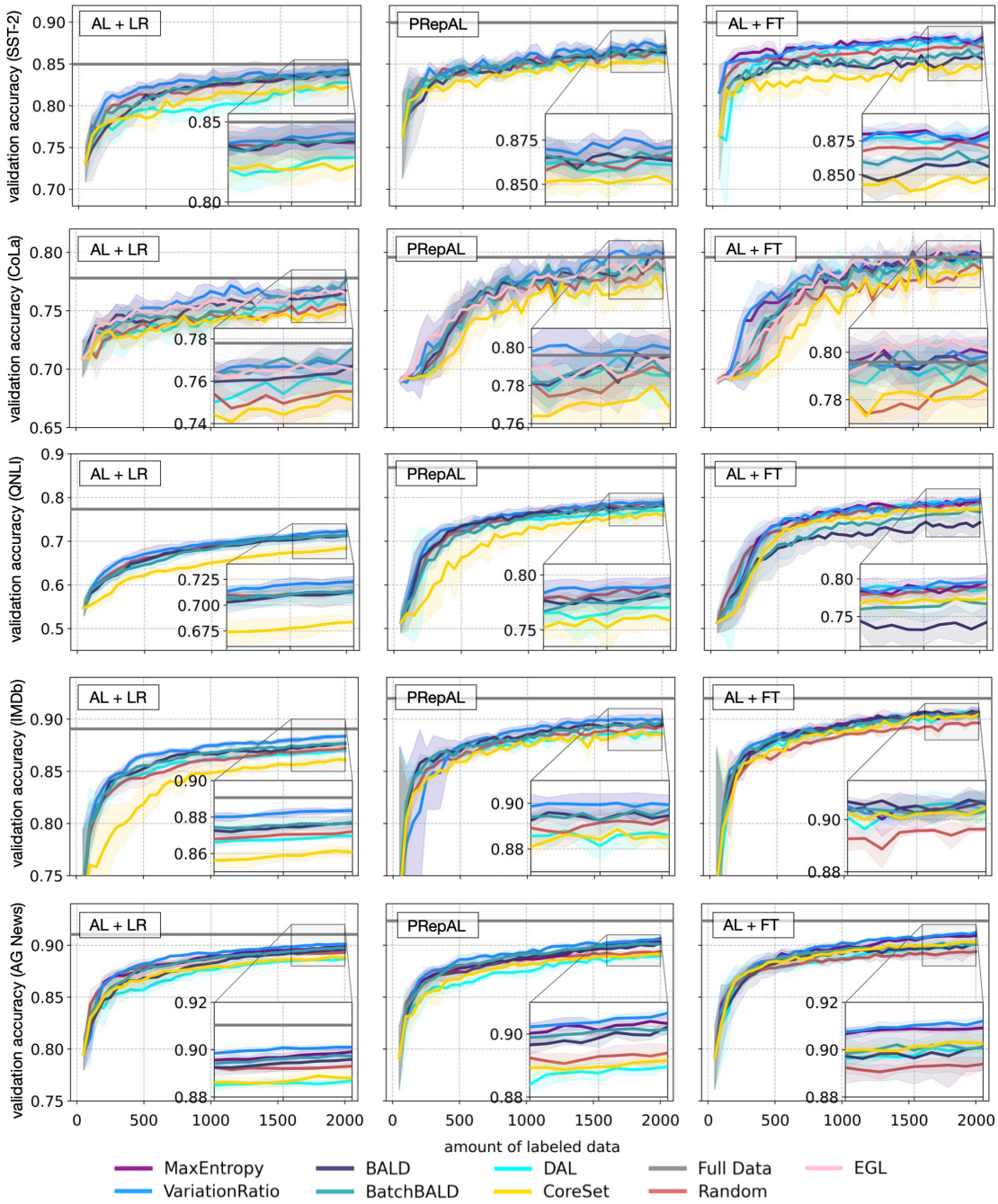
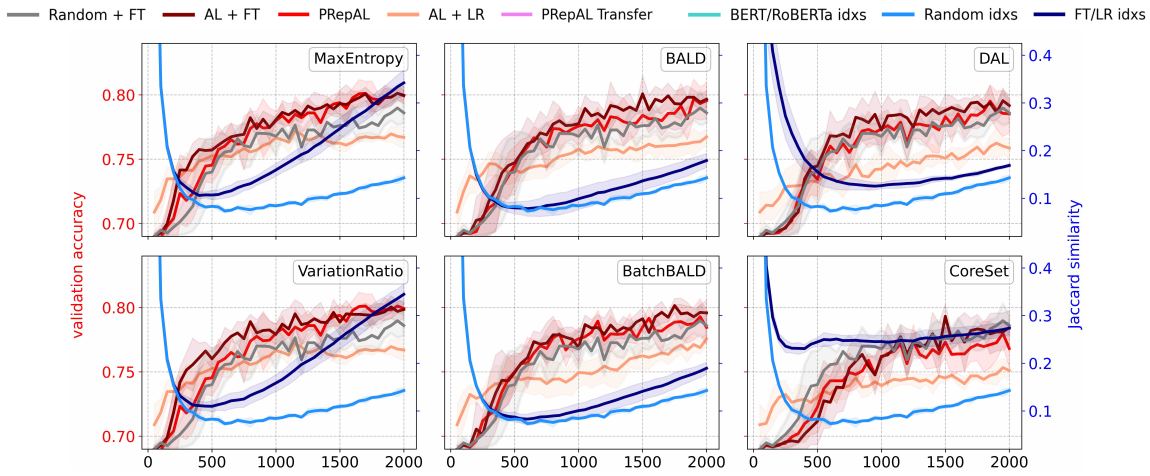
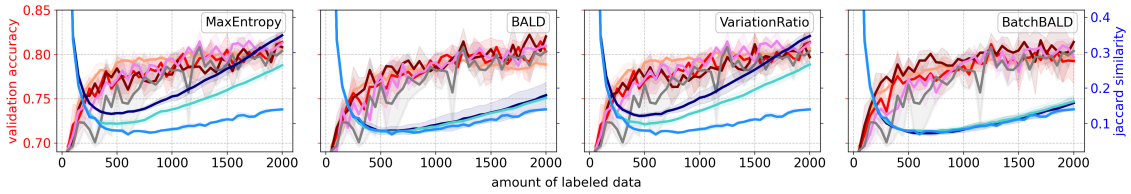


Figure 4: Validation accuracy of final models across different acquisition functions, retraining methods, and datasets. All use BERT as the backbone LLM. Error bands represent ± 1 standard deviation. Across the majority of datasets and active learning protocols, simple uncertainty-based acquisition functions like MaxEntropy and VariationRatio outperform all other methods. Note that the curves associated with MaxEntropy and VariationRatio overlap for AL+LR and PRepAL on datasets with two classes (all except AG News) because both procedures acquire the same exact data.

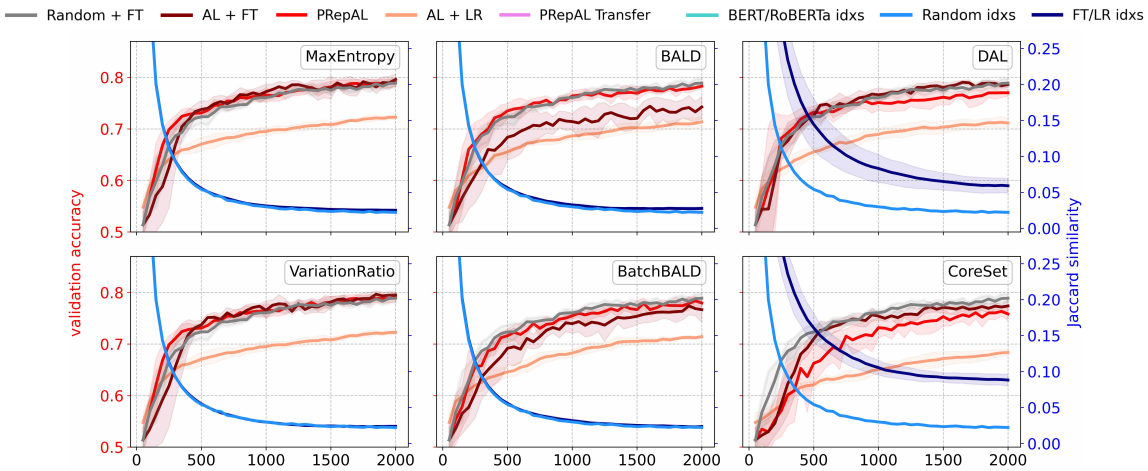


(a) BERT backbone

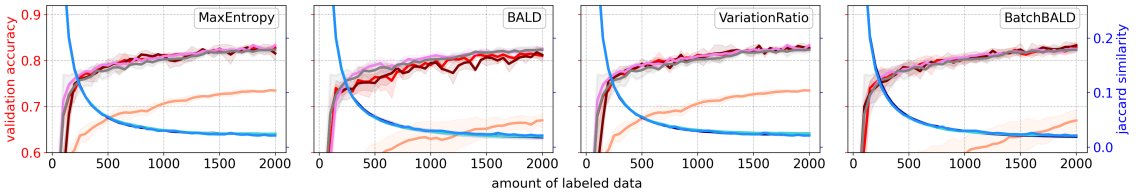


(b) RoBERTa backbone

Figure 5: CoLa dataset. The red-toned curves and the grey curve show the validation accuracy of different models with different active learning protocols. The blue-toned curves indicate Jaccard similarity between subsets of data indices selected by different active learning protocols and the data indices selected by AL+FT. Error bands represent ± 1 standard deviation. The high values of Jaccard similarity are partly due to the dataset size (only 8, 551 samples). Almost 40% of the final 2, 000 samples selected by AL+FT are in the 2,000 chosen by PRepAL (AL+LR) with MaxEntropy acquisition.

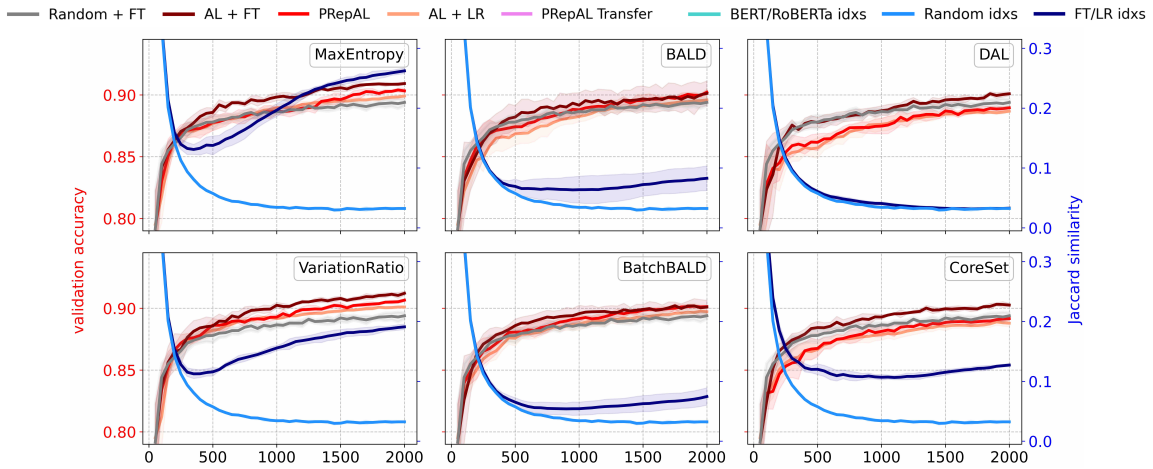


(a) BERT backbone

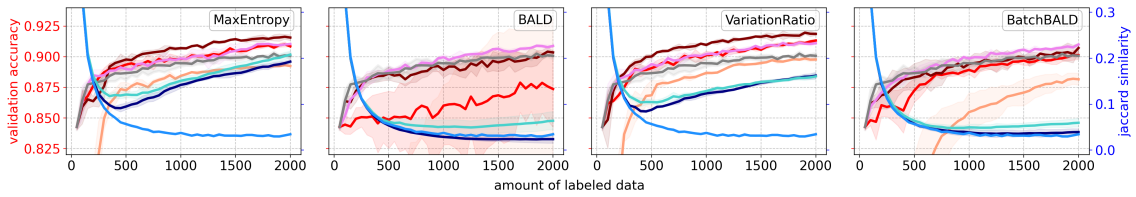


(b) RoBERTa backbone

Figure 6: QNLI dataset. The red-toned curves and the grey curve show the validation accuracy of different models with different active learning protocols. The blue-toned curves indicate Jaccard similarity between subsets of data indices selected by different active learning protocols and the data indices selected by AL+FT. Error bands represent ±1 standard deviation. PRepAL and AL+FT perform similarly, with PRepAL at an unexpected advantage for BALD and BatchBALD. Both significantly outperform AL+LR but are not better than random acquisition. The Jaccard similarity between indices associated with PRepAL and AL+FT is indistinguishable from random for uncertainty-based active learners, which is likely due to the dataset size (100K+ training samples).



(a) BERT backbone



(b) RoBERTa backbone

Figure 7: AG News dataset. The red-toned curves and the grey curve show the validation accuracy of different models with different active learning protocols. The blue-toned curves indicate Jaccard similarity between subsets of data indices selected by different active learning protocols and the data indices selected by AL+FT. Error bands represent ±1 standard deviation. Unlike other datasets, AL+LR closes in on the tight performance of PRepAL and AL+FT with BERT and even outperforms random acquisition with subsequent fine-tuning for 4/6 active learning functions.

Table 3: Validation accuracy (mean±std, in %) of the final BERT model fine-tuned on $2,000$ labeled samples selected by different acquisition functions with different retraining protocols. Accuracy above random labeling is shown in bold.

Algorithm	Protocol	SST-2	QNLI	CoLa	AG News	IMDb
Random	AL+FT	86.9 ± 0.5	78.8 ± 0.9	78.6 ± 0.8	89.4 ± 0.3	89.6 ± 0.1
Max Entropy	AL+LR	84.2 ± 0.9	72.2 ± 0.5	76.7 ± 0.8	$\mathbf{89.9 \pm 0.1}$	88.3 ± 0.1
	PRepAL	$\mathbf{87.1 \pm 0.4}$	$\mathbf{79.0 \pm 0.7}$	$\mathbf{79.9 \pm 0.4}$	$\mathbf{90.3 \pm 0.2}$	$\mathbf{89.9 \pm 0.4}$
	AL+FT	$\mathbf{88.1 \pm 0.6}$	$\mathbf{79.6 \pm 1.0}$	$\mathbf{80.0 \pm 0.7}$	$\mathbf{90.9 \pm 0.1}$	$\mathbf{90.7 \pm 0.4}$
Variation Ratio	AL+LR	84.2 ± 0.9	72.2 ± 0.5	76.7 ± 0.8	$\mathbf{90.1 \pm 0.1}$	88.3 ± 0.2
	PRepAL	$\mathbf{87.1 \pm 0.4}$	$\mathbf{79.0 \pm 0.7}$	$\mathbf{79.9 \pm 0.4}$	$\mathbf{90.6 \pm 0.1}$	$\mathbf{89.9 \pm 0.4}$
	AL+FT	$\mathbf{87.8 \pm 0.8}$	$\mathbf{79.5 \pm 0.6}$	$\mathbf{79.8 \pm 0.6}$	$\mathbf{91.2 \pm 0.1}$	$\mathbf{90.7 \pm 0.3}$
BALD	AL+LR	83.7 ± 1.0	71.3 ± 1.1	76.7 ± 0.6	$\mathbf{89.6 \pm 0.2}$	87.6 ± 0.3
	PRepAL	86.3 ± 1.2	78.2 ± 0.4	$\mathbf{79.5 \pm 1.3}$	$\mathbf{90.2 \pm 0.2}$	89.5 ± 0.2
	AL+FT	85.6 ± 0.9	74.2 ± 2.0	$\mathbf{79.6 \pm 0.6}$	$\mathbf{90.1 \pm 1.0}$	$\mathbf{90.6 \pm 0.3}$
BatchBALD	AL+LR	84.0 ± 0.7	71.4 ± 0.4	77.6 ± 0.6	$\mathbf{89.7 \pm 0.1}$	87.7 ± 0.4
	PRepAL	$\mathbf{86.9 \pm 0.2}$	77.9 ± 0.9	78.4 ± 1.3	$\mathbf{90.1 \pm 0.1}$	$\mathbf{89.7 \pm 0.2}$
	AL+FT	86.3 ± 0.5	76.6 ± 2.4	$\mathbf{79.6 \pm 1.0}$	$\mathbf{90.1 \pm 0.5}$	$\mathbf{90.3 \pm 0.3}$
DAL	AL+LR	82.7 ± 0.4	71.1 ± 0.9	75.8 ± 0.8	88.7 ± 0.1	86.9 ± 0.2
	PRepAL	86.1 ± 0.5	77.0 ± 0.1	78.5 ± 1.5	88.9 ± 0.1	88.6 ± 0.5
	AL+FT	$\mathbf{87.7 \pm 0.9}$	78.6 ± 0.5	$\mathbf{79.1 \pm 0.5}$	$\mathbf{90.1 \pm 0.1}$	$\mathbf{90.3 \pm 0.7}$
CoreSet	AL+LR	82.2 ± 0.6	68.3 ± 0.5	75.1 ± 0.5	88.8 ± 0.2	86.1 ± 0.2
	PRepAL	85.1 ± 0.5	75.8 ± 1.2	76.8 ± 1.4	89.2 ± 0.2	88.5 ± 0.6
	AL+FT	84.7 ± 0.6	77.4 ± 1.2	78.4 ± 1.3	$\mathbf{90.2 \pm 0.1}$	$\mathbf{90.4 \pm 0.5}$

Table 4: Validation accuracy (mean±std, in %) of the final RoBERTa model fine-tuned on $2,000$ labeled samples selected by different acquisition functions with different retraining protocols. Accuracy above random labeling is shown in bold.

Algorithm	Protocol	SST-2	QNLI	CoLa	AG News	IMDb
Random	AL+FT	90.6 ± 0.3	82.3 ± 0.5	80.4 ± 0.5	90.0 ± 0.2	93.1 ± 0.1
Max Entropy	AL+LR	85.5 ± 1.1	73.5 ± 0.2	$\mathbf{80.7 \pm 0.2}$	89.2 ± 0.1	90.7 ± 0.2
	PRepAL	$\mathbf{91.1 \pm 0.5}$	$\mathbf{82.8 \pm 0.6}$	$\mathbf{81.4 \pm 1.4}$	$\mathbf{90.8 \pm 0.0}$	$\mathbf{93.8 \pm 0.2}$
	AL+FT	$\mathbf{91.1 \pm 0.1}$	81.6 ± 1.6	$\mathbf{80.8 \pm 0.1}$	$\mathbf{91.5 \pm 0.2}$	$\mathbf{94.1 \pm 0.2}$
	Transfer	$\mathbf{90.9 \pm 0.4}$	$\mathbf{83.4 \pm 0.3}$	$\mathbf{81.3 \pm 0.2}$	$\mathbf{91.0 \pm 0.1}$	$\mathbf{93.6 \pm 0.1}$
Variation Ratio	AL+LR	85.5 ± 1.1	73.5 ± 0.2	$\mathbf{80.7 \pm 0.2}$	89.8 ± 0.1	90.7 ± 0.2
	PRepAL	$\mathbf{91.1 \pm 0.5}$	$\mathbf{82.8 \pm 0.6}$	$\mathbf{81.4 \pm 1.4}$	$\mathbf{91.3 \pm 0.1}$	$\mathbf{93.8 \pm 0.2}$
	AL+FT	90.3 ± 0.6	$\mathbf{82.6 \pm 0.2}$	79.7 ± 1.1	$\mathbf{91.8 \pm 0.1}$	$\mathbf{94.1 \pm 0.2}$
	Transfer	$\mathbf{90.9 \pm 0.3}$	$\mathbf{83.4 \pm 0.3}$	$\mathbf{81.3 \pm 0.2}$	$\mathbf{91.1 \pm 0.1}$	$\mathbf{93.6 \pm 0.1}$
BALD	AL+LR	80.5 ± 3.6	67.0 ± 1.5	78.9 ± 0.7	72.8 ± 9.3	81.9 ± 9.2
	PRepAL	89.5 ± 0.6	81.1 ± 0.5	80.3 ± 2.0	87.3 ± 3.1	$\mathbf{93.1 \pm 0.3}$
	AL+FT	89.0 ± 0.3	81.2 ± 1.4	$\mathbf{82.0 \pm 0.1}$	$\mathbf{90.3 \pm 0.1}$	$\mathbf{93.9 \pm 0.1}$
	Transfer	$\mathbf{90.6 \pm 0.9}$	$\mathbf{82.6 \pm 0.1}$	$\mathbf{81.0 \pm 1.0}$	$\mathbf{90.9 \pm 0.1}$	$\mathbf{93.4 \pm 0.1}$
BatchBALD	AL+LR	83.9 ± 0.4	67.0 ± 2.0	79.4 ± 0.5	88.1 ± 0.5	89.3 ± 0.2
	PRepAL	89.6 ± 0.5	$\mathbf{82.9 \pm 0.9}$	79.2 ± 2.5	$\mathbf{90.1 \pm 0.3}$	$\mathbf{93.3 \pm 0.3}$
	AL+FT	$\mathbf{90.7 \pm 0.2}$	$\mathbf{83.3 \pm 0.5}$	$\mathbf{81.4 \pm 0.3}$	$\mathbf{90.7 \pm 0.4}$	$\mathbf{93.9 \pm 0.1}$
	Transfer	90.1 ± 0.6	$\mathbf{82.5 \pm 0.4}$	79.7 ± 0.7	$\mathbf{91.0 \pm 0.1}$	$\mathbf{93.1 \pm 0.4}$