Uncertainty-based Query Strategies for Active Learning with Transformers

Christopher Schröder*, Andreas Niekler, and Martin Potthast
Leipzig University

Abstract

Active learning is the iterative construction of a classification model through targeted labeling, enabling significant labeling cost savings. As most research on active learning has been carried out before transformer-based language models ("transformers") became popular, despite its practical importance, comparably few papers have investigated how transformers can be combined with active learning to date. This can be attributed to the fact that using state-of-the-art query strategies for transformers induces a prohibitive runtime overhead, which effectively cancels out, or even outweighs aforementioned cost savings. In this paper, we revisit uncertainty-based query strategies, which had been largely outperformed before, but are particularly suited in the context of fine-tuning transformers. In an extensive evaluation on five widely used text classification benchmarks, we show that considerable improvements of up to 14.4 percentage points in area under the learning curve are achieved, as well as a final accuracy close to the state of the art for all but one benchmark, using only between 0.4% and 15% of the training data.

1 Introduction

Collecting labeled data for machine learning can be costly and time-consuming. A key technique to minimize labeling costs has been active learning, where an oracle (e.g., a human expert) is queried to label selected problem instances deemed to be most informative to the learning algorithm's next iteration, according to a query strategy.

Active learning is characterized by the real-world machine learning scenario in which large amounts of training data are unavailable, which may explain why comparably little research has investigated deep learning in this context. The recent, widely successful transformer-based language models can circumvent the limitations imposed by small training datasets (Vaswani et al., 2017; Devlin et al., 2019). Pre-trained on large amounts of unlabeled text, they can be fine-tuned to a given task using far less training data than when trained from scratch. However, their high number of model parameters renders them computationally highly expensive, for query strategies that are targeted at neural networks or text classification (Settles et al., 2007; Zhang et al., 2017), resulting in prohibitive turnaround times between labeling steps.

In this paper, we systematically investigate uncertainty-based query strategies as a computationally inexpensive alternative. Despite their relative disadvantages in traditional active learning, when paired with transformers, they are highly effective as well as efficient. Our extensive experiments encompass the state-of-the-art transformer models ELECTRA (Clark et al., 2020), BERT (Devlin et al., 2019), and DistilRoBERTa (Sanh et al., 2019), and five well-known sentence classification benchmarks for active learning.

2 Related Work

Uncertainty-based query strategies used to be the most common choice in active learning, using uncertainty scores obtained from the learning algorithm (Lewis and Gale, 1994), estimates obtained via ensembles (Krogh and Vedelsby, 1994; RayChaudhuri and Hamey, 1995), or prediction entropy (Holub et al., 2008). More recently—predating transformers—neural network-based active learning predominantly employed query strategies that select problem instances according to (1) the magnitude of their backpropagation-induced gradients (Settles et al., 2007; Zhang et al., 2017), where instances causing a high-magnitude gradient inform the model better, and (2) representativity-based criteria (e.g., coresets (Sener and Savarese, 2018)), which select instances from a vector space to geometrically represent the full set.

* Corresponding author:
christopher.schroeder@uni-leipzig.de

For today's deep neural networks, ensembles are too computationally expensive, and prediction entropy has been observed to be overconfident (Guo et al., 2017; Lakshminarayanan et al., 2017). The exception are flat architectures, where among others Prabhu et al. (2019) showed fastText (Joulin et al., 2017) to be effective, well-calibrated, and computationally efficient. Prior to transformers, query strategies relying on expected gradient length (Settles et al., 2007) achieved the best results on many active learning benchmarks for text classification (Zhang et al., 2017). Gradients depend on the current model, which means, when used for a query strategy, they scale with the vast number of a transformer's parameters, and moreover, they need to be computed per-instance instead of batch-wise, thereby becoming computationally expensive.

The cost of ensembles, the adverse scaling of network parameters in gradient-based strategies, and a history of deeming neural networks to be overconfident effectively rules out the most predominantly used query strategies. This might explain why transformers, despite the success of fine-tuning them for text classification (Howard and Ruder, 2018; Yang et al., 2019; Sun et al., 2020), have only very recently been considered at all in combination with active learning (Lu and MacNamee, 2020; Yuan et al., 2020; Ein-Dor et al., 2020; Margatina et al., 2021). All of the related works mitigate the computationally complex query strategies by subsampling the unlabeled data before querying (Lu and MacNamee, 2020; Ein-Dor et al., 2020), by performing fewer queries with larger sample sizes (Yuan et al., 2020; Margatina et al., 2021), or by tailoring to less expensive learning settings, namely binary classification (Ein-Dor et al., 2020). Uncertainty-based query strategies have not been systematically analyzed in connection with transformers. Our work not only closes this gap, but does so in an experimental setup that allows for a direct comparison of transformer-based active learning to traditional active learning on standard benchmarks, which, too, has been largely neglected by the aforementioned related work.

3 Transformer-based Active Learning

The goal of active learning is to minimize the labeling costs of training data acquisition while maximizing a model's performance (increase) with each newly labeled problem instance. In contrast to regular supervised text classification ("passive learning"), it operates iteratively, where in each iteration (1) an algorithm (i.e., query strategy) selects new instances for labeling according to an estimation of their informativeness, (2) an oracle (e.g., a human expert) provides the respective label, and (3) a learning algorithm either uses the newly labeled instance for its next learning step, or a model is retrained from scratch using all previously labeled instances. This work considers pool-based active learning (Lewis and Gale, 1994), where the query strategies have access to all unlabeled data.

Query Strategies We consider two uncertainty-based query strategies, and a baseline: (1) Prediction Entropy (PE; Roy and McCallum (2001); Schohn and Cohn (2000)) selects instances with the highest entropy in the predicted label distribution with the aim to reduce overall entropy. (2) Breaking Ties (BT; Luo et al. (2005)) takes instances with the minimum margin between the top two predicted classes' confidence scores and is equivalent to PE in the binary case. In recent years, BT has only been used as a baseline for deep image classification (Ash et al., 2020) but has otherwise barely received any attention. (3) Random Sampling (RS), is a commonly used baseline in active learning.

Oracle The oracle is usually operationalized using the training datasets of existing benchmarks: To ensure comparability with the literature, we pick some of the standard text classification tasks.

Learning Algorithms We fine-tune the ELECTRA discriminator model (Clark et al., 2020), which provided superior results to BERT (Devlin et al., 2019) on several natural language understanding datasets, but is not yet evaluated for active learning and text classification. As a point of reference, and due to nonexistent results on the benchmark datasets, we also include BERT as strong and—apart from active learning—well-researched transformer baseline. BERT and ELECTRA both have 24-layers, hidden units of size 1024 and 336M parameters in total. As a more parameter-efficient alternative, we also investigate DistilRoBERTa (Sanh et al., 2019), which consists of merely six layers, hidden units of size 768, and 82M parameters.

The classification model consists of the respective transformer, on top of which we add a fully connected projection layer, and a final softmax output layer. We use the element-wise average of the token vectors (Lu and MacNamee, 2020) which are computed by the transformer. Regarding fine-

(a)

Dataset Name (ID)	Type	Classes	Training	Test
AG's News (AGN)	News Topics	4	120,000	(*) 7,600
Customer Reviews (CR)	Sentiment	2	3,397	378
Movie Reviews (MR)	Sentiment	2	9,596	1,066
Subjectivity (SUBJ)	Sentiment	2	9,000	1,000
TREC-6 (TREC-6)	Questions	6	5,500	(*) 500

	(d)			(e)	
Dataset	Model	Acc.	Data Use	Model	AUC
AGN	D.RoBERTa (BT)	0.894	0.4%	D.RoBERTa (BT)	0.875
	ELECTRA (passive)	0.948	100.00%		–
	XLNet[1]	0.955	100.00%		–
CR	ELECTRA (PE)	0.938	15.45%	ELECTRA (RS)	0.884
	ELECTRA (passive)	0.932	100.00%	CNN[6]	0.743
	Capsule+ELMo[2]	0.889	100.00%		–
MR	ELECTRA (PE)	0.909	0.547%	ELECTRA (RS)	0.851
	ELECTRA (passive)	0.929	100.00%	CNN[6]	0.707
	SimCSE-RoBERTa[3]	0.884	100.00%		–
SUBJ	ELECTRA (PE)	0.969	5.83%	BERT (RS)	0.939
	ELECTRA (passive)	0.979	100.00%	CNN[6]	0.856
	AdaSent[4]	0.955	100.00%		–
TREC-6	ELECTRA (BT)	0.958	9.55%	D.RoBERTa (BT)	0.864
	ELECTRA (passive)	0.956	100.00%		–
	RCNN[5]	0.962	100.00%		–

		(b) Accuracy per Strategy			(c) AUC per Strategy		
Dataset / Model		PE	BT	RS	PE	BT	RS
AGN	SVM	0.804	0.804	0.801	0.693	0.705	0.699
	KimCNN	0.871	0.874	0.866	0.753	0.791	0.810
	DistilRoBERTa	0.892	0.894	0.879	0.855	**0.875**	0.855
	BERT	0.896	**0.904**	0.884	0.858	0.872	0.849
	ELECTRA	0.864	0.882	0.879	0.743	0.782	0.807
CR	SVM	0.757		0.763	0.717		0.718
	KimCNN	0.765		0.745	0.713		0.705
	DistilRoBERTa	0.906		0.886	0.874		0.870
	BERT	0.904		0.896	0.877		0.868
	ELECTRA	**0.938**		0.932	0.861		**0.884**
MR	SVM	0.674		0.641	0.612		0.597
	KimCNN	0.719		0.720	0.674		0.677
	DistilRoBERTa	0.819		0.809	0.784		0.783
	BERT	0.857		0.846	0.833		0.827
	ELECTRA	**0.909**		0.901	0.845		**0.851**
SUBJ	SVM	0.843		0.839	0.801		0.797
	KimCNN	0.897		0.896	0.859		0.864
	DistilRoBERTa	0.944		0.926	0.924		0.902
	BERT	0.957		0.949	**0.939**		0.933
	ELECTRA	**0.969**		0.965	0.930		0.933
TREC-6	SVM	0.740	0.758	0.742	0.491	0.648	0.619
	KimCNN	0.840	0.836	0.792	0.711	0.714	0.688
	DistilRoBERTa	0.942	0.950	0.918	0.840	**0.864**	0.856
	BERT	0.932	0.947	0.921	0.789	0.844	0.828
	ELECTRA	0.946	**0.958**	0.942	0.786	0.832	0.841

Table 1: (a) Key information about the datasets. (*): Predefined test sets were available and adopted. (b) Accuracy after the final iteration per query strategy (averaged over five runs). (c) Area under the learning curve per query strategy (averaged over five runs). (d) Final accuracy compared to (our) passive classification and previously reported state-of-the-art text classification: [1]Yang et al. (2019), [2]Zheng et al. (2019), [3]Gao et al. (2021), [4]Zhao et al. (2015), [5]Tay et al. (2018). (e) Best AUC scores compared to [6]Zhang et al. (2017).

tuning, we adopt the combination of discriminative fine-tuning, slanted triangular learning rates, and gradual unfreezing, as described by Howard and Ruder (2018). The main active learning routine is then as follows: (1) The query strategy, either using the model from the previous iteration, or sampling randomly, selects 25 instances. (2) The oracle provides labels for these instances. (3) The next model is trained using all data labeled so far.

Baselines. For comparison, we consider a linear SVM, and KimCNN (Kim, 2014), which have been used extensively in text classification, disregarding active learning. We adopted the KimCNN parameters from Kim (2014) and Zhang et al. (2017).

4 Evaluation

We evaluate fine-tuning ELECTRA, BERT and DistilRoBERTa models during active learning.

Datasets and Experimental Setup Table 1a shows the five datasets employed, which have previously been used to evaluate active learning: AG's News (AGN; Zhang et al. (2015)),

Customer Reviews (CR; Hu and Liu (2004)), Movie Reviews (MR; Pang and Lee (2005)), Subjectivity (SUBJ; Pang and Lee (2004)), and TREC-6 (Li and Roth (2002)). These datasets encompass binary and multi-class classification in different domains, and they are class-balanced, except for TREC-6. Where available, we employed the pre-existing test sets, or else a random sample of 10%. We follow the experiment setup of Zhang et al. (2017): 25 training instances to train the first model, followed by 20 active learning iterations, during each of which 25 instances are queried and labeled. Using 10% of the so far labeled data as validation set, transformers are trained for 50 epochs on AGN, and for 15 epochs otherwise. We stop training early (Duong et al., 2018) when accuracy surpasses 98%, or the validation loss does not increase for five epochs. Moreover, we carry over the model weights as initialization for the next model.

Results For each combination of dataset, model, and query strategy, Tables 1b and c report the final iteration's classification performance over five experiment runs in terms of average accuracy and average area under the learning curve (AUC). From Table 1b it can be observed that configurations based on transformers achieve major improvements in accuracy—similarly as previously reported for text classification (Howard and Ruder, 2018; Sun et al., 2020). The ELECTRA models outperform the baselines in every configuration by a large mar-

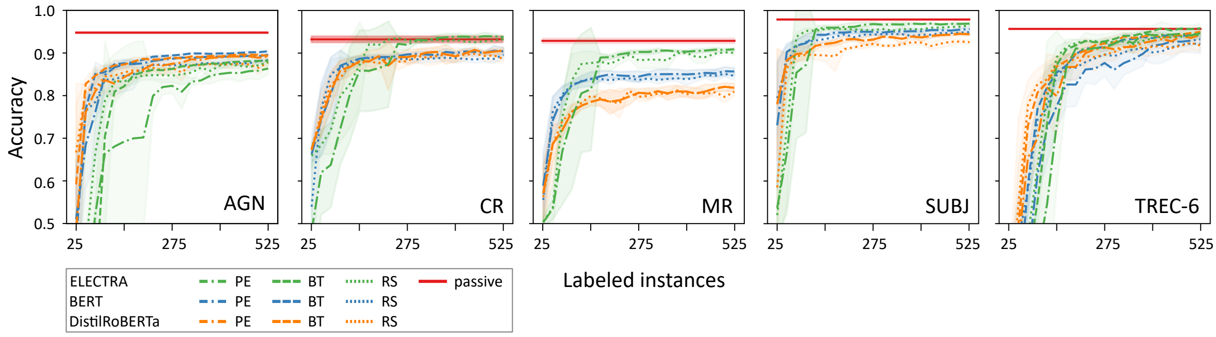

Figure 1: Active learning curves of three transformer-based models when combined with two (inner plots) or three (outer plots) query strategies: Prediction Entropy (PE), Breaking Ties (BT), and Random Sampling (RS). We report the mean and standard deviation over five runs. For comparison, the horizontal line depicts the best (passive) text classification result, where ELECTRA always achieved the best accuracy scores.

gin of up to 26 percentage points (on MR) in accuracy. Moreover, even compared to DistilRoBERTa, accuracy gains of up to 9.2 percentage points (on MR) can be observed, except for AGN, for which BERT reached the best result with an additional 2.2 points. As for query strategies, PE always resulted in highest accuracies for binary classification problems and likewise BT succeeds for multi-class.

Figure 1 shows the learning curves for all transformer configurations. The respective final scores can be read from Table 1c. For comparison, we also show the best model's score when trained on the full dataset (horizontal line). RS is a strong contender for binary classification, where it reaches the highest AUC scores. Still, this is first due to the datasets being balanced, and second, stems mostly from a steep rise during the earlier iterations. In terms of accuracy it is later outperformed by PE and BT in almost all cases. For imbalanced datasets, Ein-Dor et al. (2020) have shown RS to be less effective, which we can confirm for BERT and DistilRoBERTa on TREC-6. ELECTRA, however, seems to struggle during initial iterations as shown by a large standard deviation in Figure 1, but still reaches the highest final accuracy in most cases. In the multi-class setting, BT reaches the highest AUC scores both per dataset and per model, while simultaneously scoring the best accuracy results.

Table 1d lists our best transformer model trained via active learning per dataset, which we compare against passive text classification, namely (1) our own model when trained on the full training set and (2) state-of-the-art results. The largest discrepancy between active learning and passive text classification is observed on AGN, which is also the largest dataset from which the active learning models use less than 1% for training. Otherwise, all models are close (or even surpass the state of the art), using only between 0.4% and 14% of the training data.

In Table 1e, we report the best AUC scores per dataset and, where applicable, compare them to previous work. ELECTRA achieves the best scores on CR and MR, while BERT is best for SUBJ; all of them use RS and show a considerable increase in AUC compared to Zhang et al. (2017).

In conclusion, these results show that ELECTRA's previously reported performance in natural language understanding (Clark et al., 2020) is transferable to active learning for text classification. Although ELECTRA outperforms DistilRoBERTa and BERT in many configurations, the latter seem to be more stable in early iterations. Finally, we observe BT to be superior to the more widely-used PE strategy for multi-class problems.

5 Conclusions

An investigation of the effectiveness of ELECTRA, BERT, and DistilRoBERTa in active learning for text classification on several sentence classification datasets shows that transformers can outperform previous models by a large margin. As it turns out, uncertainty-based query strategies, which have been largely ignored in recent work, still perform well. In particular, the breaking ties strategy is superior to the well-known prediction entropy strategy for multi-class active learning. On four out of five datasets, we achieve results close to state-of-the-art text classification, using only a fraction of the training data. This shows that uncertainty-based strategies demand renewed attention in the context of transformers, and moreover, that transformers in turn can yet again shine, where previous methods were limited by small amounts of training data.

Ethical Considerations

Research on active learning improves the labeling of data, by efficiently supporting the learning algorithm with targeted information, so that overall less data has to be labeled. This could contribute to creating machine learning models, which would otherwise be infeasible, either due to limited budget, or time. Active learning can be used for good or bad, and our contributions would—in both cases– show how to make this process more efficient.

Moreover, we use pre-trained models, which can contain one or more types of bias. Bias, however, affects all approaches based on fine-tuning pre-trained language models, but therefore this has to be kept in mind and mitigated all the more.

References

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2020. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Long Duong, Hadi Afshar, Dominique Estival, Glen Pink, Philip Cohen, and Mark Johnson. 2018. Active learning for deep semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 43–48, Melbourne, Australia. Association for Computational Linguistics.

Liat Ein-Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2020. Active Learning for BERT: An Empirical Study. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7949–7962, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330, International Convention Centre, Sydney, Australia. PMLR.

A. Holub, P. Perona, and M. C. Burl. 2008. Entropy-based active learning for object recognition. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. Association for Computing Machinery.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Anders Krogh and Jesper Vedelsby. 1994. Neural network ensembles, cross validation and active learning. In *Proceedings of the 7th International Conference on Neural Information Processing Systems*, NIPS'94, pages 231–238, Cambridge, MA, USA. MIT Press.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30*, pages 6402–6413. Curran Associates, Inc.

David D Lewis and William A Gale. 1994. A sequential algorithm for training text classifiers. In *SIGIR'94*, pages 3–12. Springer.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics*, volume 1 of *COLING '02*, pages 1–7, USA. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jinghui Lu and Brian MacNamee. 2020. Investigating the effectiveness of representations based on pretrained transformer-based language models in active learning for labelling text datasets. *arXiv preprint arXiv:2004.13138*.

Tong Luo, Kurt Kramer, Dmitry B. Goldgof, Lawrence O. Hall, Scott Samson, Andrew Remsen, and Thomas Hopkins. 2005. Active learning to recognize multiple types of plankton. *J. Mach. Learn. Res.*, 6:589–613.

Katerina Margatina, Loic Barrault, and Nikolaos Aletras. 2021. Bayesian active learning with pretrained language models. *arXiv preprint arXiv:2104.08320*.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Ameya Prabhu, Charles Dognin, and Maneesh Singh. 2019. Sampling bias in deep active classification: An empirical study. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4058–4068, Hong Kong, China. Association for Computational Linguistics.

T. RayChaudhuri and L. G. C. Hamey. 1995. Minimisation of data collection by active learning. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 3, pages 1338–1341.

Nicholas Roy and Andrew McCallum. 2001. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML 01, pages 441–448. Morgan Kaufmann Publishers Inc.

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

Greg Schohn and David Cohn. 2000. Less is more: Active learning with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000)*, pages 839–846. Morgan Kaufmann.

Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings*.

Burr Settles, Mark Craven, and Soumya Ray. 2007. Multiple-instance active learning. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 1289–1296, Red Hook, NY, USA. Curran Associates Inc.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2020. How to Fine-Tune BERT for Text Classification? *arXiv preprint arXiv:1905.05583*.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018. Recurrently controlled recurrent networks. In *Advances in Neural Information Processing Systems 31*, pages 4731–4743. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5753–5763. Curran Associates, Inc.

Michelle Yuan, Hsuan-Tien Lin, and Jordan Boyd-Graber. 2020. Cold-start active learning through self-supervised language modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7935–7948, Online. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 649–657. Curran Associates, Inc.

Ye Zhang, Matthew Lease, and Byron C. Wallace. 2017. Active discriminative text representation learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 3386–3392. AAAI Press.

Han Zhao, Zhengdong Lu, and Pascal Poupart. 2015. Self-adaptive hierarchical sentence model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 4069–4076. AAAI Press.

Wanshan Zheng, Zibin Zheng, Hai Wan, and Chuan Chen. 2019. Dynamically route hierarchical structure representation to attentive capsule for text classification. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5464–5470. International Joint Conferences on Artificial Intelligence Organization.

Appendix

In the following, we summarize important details for reproducing the experiments.

A Technical Environment

All experiments were conducted within a Python 3.8 environment. The system had CUDA 10.2 installed and was equipped with an NVIDIA GeForce RTX 2080 Ti (11GB VRAM). Computations for fine-tuning transformers and training KimCNN were performed on the GPU, while the SVM-based runs were executed on the CPU.

B Implementation Details

Our experiments are built using well-known machine learnign libraries: PyTorch[1], huggingface transformers[2], scikit-learn[3], scipy[4], and numpy[5].

[1]https://pytorch.org/, 1.7.1

C Experiments

Each experiment configuration represents a combination of model, dataset and query strategy, and has been run for five times. We use a class-balanced initial set to support the warm start of the first model for the imbalanced TREC-6 dataset, whose rarest class would otherwise only rarely be encountered if sampled randomly.

C.1 Pre-Trained Models

We fine-tuned DistilRoBERTa (distilroberta-base), BERT-large (bert-base-uncased), and ELECTRA (google/electra-large-discriminator). All of them have been published to the huggingface model repository, and are therefore easily obtainable.

C.2 Datasets

Our experiments used datasets that are well-known benchmarks in text classification and active learning. All datasets have been made accessible to the Python ecosystem by existing Python APIs, which

[2]https://github.com/huggingface/transformers, 4.11.0
[3]https://scikit-learn.org/, 0.24.0
[4]https://www.scipy.org/, 1.6.0
[5]https://numpy.org/, 1.19.5

Dataset	Model	PE	BT
AGN	SVM	1.87 ± 0.37	0.92 ± 0.18
	KimCNN	7.33 ± 1.47	6.26 ± 1.25
	DistilRoBERTa	98.42 ± 19.69	97.30 ± 19.46
	BERT	533.97 ± 106.89	508.30 ± 101.67
	ELECTRA	507.54 ± 101.52	508.81 ± 101.77
CR	SVM	0.01 ± 0.00	–
	KimCNN	0.19 ± 0.04	–
	DistilRoBERTa	1.96 ± 0.39	–
	BERT	12.23 ± 2.45	–
	ELECTRA	12.50 ± 2.50	–
MR	SVM	0.01 ± 0.00	–
	KimCNN	0.53 ± 0.11	–
	DistilRoBERTa	7.63 ± 1.53	–
	BERT	41.83 ± 8.37	–
	ELECTRA	40.57 ± 8.11	–
SUBJ	SVM	0.01 ± 0.00	–
	KimCNN	0.48 ± 0.10	–
	DistilRoBERTa	5.27 ± 1.05	–
	BERT	31.63 ± 6.33	–
	ELECTRA	32.83 ± 6.58	–
TREC-6	SVM	0.09 ± 0.02	0.04 ± 0.01
	KimCNN	0.29 ± 0.06	0.25 ± 0.05
	DistilRoBERTa	2.96 ± 0.59	2.91 ± 0.58
	BERT	14.72 ± 2.94	14.68 ± 2.94
	ELECTRA	14.69 ± 2.94	15.44 ± 3.09

Table 2: Additional runtime information about the experiments (in seconds). The reported values are the average and standard deviation over five runs. As prediction entropy (PE) and breaking ties (BT) yield the same result in binary classification, we omitted the latter.

Dataset	Model	PE	BT	RS
AGN	SVM	0.804 ± 0.000	0.804 ± 0.000	0.801 ± 0.006
	KimCNN	0.871 ± 0.004	0.874 ± 0.005	0.866 ± 0.007
	DistilRoBERTa	0.892 ± 0.002	0.894 ± 0.003	0.879 ± 0.008
	BERT	0.896 ± 0.003	0.904 ± 0.002	0.884 ± 0.003
	ELECTRA	0.864 ± 0.017	0.882 ± 0.007	0.879 ± 0.008
CR	SVM	0.767 ± 0.000		0.763 ± 0.025
	KimCNN	0.765 ± 0.012		0.745 ± 0.014
	DistilRoBERTa	0.906 ± 0.007		0.886 ± 0.007
	BERT	0.904 ± 0.010		0.896 ± 0.008
	ELECTRA	0.938 ± 0.004		0.932 ± 0.006
MR	SVM	0.674 ± 0.000		0.641 ± 0.010
	KimCNN	0.719 ± 0.011		0.720 ± 0.013
	DistilRoBERTa	0.819 ± 0.012		0.809 ± 0.011
	BERT	0.857 ± 0.009		0.846 ± 0.011
	ELECTRA	0.909 ± 0.003		0.901 ± 0.008
SUBJ	SVM	0.843 ± 0.000		0.839 ± 0.012
	KimCNN	0.897 ± 0.004		0.896 ± 0.009
	DistilRoBERTa	0.944 ± 0.004		0.926 ± 0.005
	BERT	0.957 ± 0.004		0.949 ± 0.003
	ELECTRA	0.969 ± 0.002		0.965 ± 0.003
TREC-6	SVM	0.740 ± 0.000	0.758 ± 0.000	0.742 ± 0.031
	KimCNN	0.840 ± 0.016	0.836 ± 0.012	0.792 ± 0.020
	DistilRoBERTa	0.942 ± 0.008	0.950 ± 0.009	0.918 ± 0.016
	BERT	0.932 ± 0.010	0.947 ± 0.014	0.921 ± 0.025
	ELECTRA	0.946 ± 0.011	0.958 ± 0.011	0.942 ± 0.008

Table 3: Mean and standard deviation of the final iteration's accuracy over five runs.

provide fast access to the raw text of those datasets. We obtain CR and SUBJ using gluonnlp, and AGN, MR, and TREC using huggingface datasets.

C.3 Evaluation Metrics

Active learning was evaluated using standard active learning metrics, namely accuracy und area under the learning curve. For both metrics, the respective scikit-learn implementation was used.

C.4 Hyperparameters

Maximum Sequence Lenght For both KimCNN and transformers, we set the maximum sequence length to the minimum multiple of ten for which 95% of the respective dataset's sentences contain less than or an equal number of token.

Dataset	Max. Seq. Length
AGN	60
CR	50
MR	60
SUBJ	50
TREC	40

Table 4: Hyperparameter settings for the maximum sequence length (as number of tokens) per dataset.

C.4.1 Transformers

AGN is trained for 50 epochs, and all other datasets for 15 epochs. We use AdamW (Loshchilov and Hutter, 2019) as optimizer, with a base learning rate of $2e{-}5$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e{-}8$. Training is done with a batch size of 12.

C.4.2 KimCNN

We stick to the parameters reported by Zhang et al. (2017), i.e., filter heights of $(3, 4, 5)$ and 50 filters.

D Further Details

We report the query strategies' runtimes in Table 2 and the standard deviations for the final accuracy in the main experiment in Table 3.