Active Curriculum Learning

Borna Jafarpour[1], **Nicolai Pogrebnyakov**[1,2], **Dawn Sepehr**[1]
[1] Thomson Reuters, Toronto, Canada
[2] Copenhagen Business School, Frederiksberg, Denmark
{firstname.lastname}@thomsonreuters.com

Abstract

This paper investigates and reveals the relationship between two closely related machine learning disciplines, namely Active Learning (AL) and Curriculum Learning (CL), from the lens of several novel curricula. This paper also introduces Active Curriculum Learning (ACL) which improves AL by combining AL with CL to benefit from the dynamic nature of the AL informativeness concept as well as the human insights used in the design of the curriculum heuristics. Comparison of the performance of ACL and AL on two public datasets for the Named Entity Recognition (NER) task shows the effectiveness of combining AL and CL using our proposed framework.

1 Introduction

Modern deep learning architectures predominantly need large amounts of labeled data to achieve high levels of performance. In the presence of a large unlabeled corpus, data points are usually chosen randomly to be annotated. However, annotation can be a costly task and not all the annotations are equally beneficial. Active Learning (AL) aims to reduce the number of annotations required to train a machine learning model by choosing the most "informative" unlabeled data for annotation. The informativeness is determined by querying a model or a set of models trained on the available annotated data (Settles 2012). Algorithm 1 shows AL more formally.

Several categories of informativeness score have been developed in the literature. For example, uncertainty metrics select unlabeled data for which the model has the highest uncertainty of label prediction (Settles and Craven 2008). Examples of uncertainty measures for a classification task are the difference of the probability of prediction for the first and second most likely classes (i.e., the margin of the prediction probability) and the entropy of prediction over all classes (i.e., $-\sum_{i=1}^{c} p_i \log p_i$ where c is the number of classes). Lower values of margin and higher values of entropy metrics are associated with higher uncertainty and consequently informativeness. Some other examples of informativeness scoring methods for unlabeled data are the amount of prediction disagreement in a committee of models (Melville and Mooney 2004) and the amount of expected change to model weights (Zhang, Lease, and Wallace 2017) or loss value (Long et al. 2014).

Curriculum Learning (CL), on the other hand, attempts to mimic how humans learn and uses that knowledge to train better models (Bengio et al. 2009; Soviany et al. 2021). Complex topics are taught to humans based on a curriculum which takes into account the level of difficulty of the material presented to the learner. CL borrows this idea and engages the human experts to design a metric that is used to sort the annotated training data from "easy" to "hard" to be presented to the model during training (Bengio et al. 2009). The

1. Seed labeled data $\boldsymbol{D^L} = \{(x_1, y_1), \ldots, (x_k, y_k)\}$
2. Unlabeled data $\boldsymbol{D^U} = \{x_{k+1}, \ldots, x_m\}$
3. While the stopping criterion is not met:
   3.1. Fine-tune or train model $\boldsymbol{M}$ on $\boldsymbol{D^L}$
   3.2. $\boldsymbol{I} :=$ the set of i most informative data samples in $\boldsymbol{D^U}$ according to $\boldsymbol{M}$
   3.3. $\boldsymbol{D^U} := \boldsymbol{D^U} \setminus \boldsymbol{I}$; $\boldsymbol{D^L} := \boldsymbol{D^L} \cup \boldsymbol{L(I)}$

Algorithm 1: Steps of the AL algorithm where $L(\boldsymbol{I})$ denotes the set $\boldsymbol{I}$ after annotation. An example of stopping criterion can be a minimum value for accuracy.

1. Training data $\boldsymbol{D^T} = \{\}$
2. Available data $\boldsymbol{D^A} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$
3. Repeat until $\boldsymbol{D^A}$ is empty:
   3.1. $\boldsymbol{I} :=$ the set of $\boldsymbol{k}$ easiest examples in $\boldsymbol{D^A}$ according to *a fixed curriculum*
   3.2. $\boldsymbol{D^T} := \boldsymbol{D^T} \cup \boldsymbol{I}$; $\boldsymbol{D^A} := \boldsymbol{D^A} \setminus \boldsymbol{I}$
   3.3. Fine-tune existing model $\boldsymbol{M}$ on $\boldsymbol{D^T}$

Algorithm 2: Steps of the CL algorithm.

Proceedings of the First Workshop on Interactive Learning for Natural Language Processing , pages 40–45

goal of CL is to find a better local optimum faster compared to randomly presenting the data to the model by smoothing the loss function in early stages of training. CL algorithm is presented in Algorithm 2. CL has been investigated in computer vision (Gui, Baltrusaitis, and Morency 2017), Natural Language Processing (NLP) (Rao, Anuranjana, and Mamidi 2020), and speech recognition (Braun, Neil, and Liu 2016) among others (Soviany et al. 2021). Specifically within NLP, CL has been used on tasks such as question answering (Sachan and Xing 2016), natural language understanding (Xu et al. 2020), as well as learning word representations (Tsvetkov et al. 2016). Different curriculum designs has been investigated by considering heuristics such as sentence length, word frequency, language model score, and parse tree depth (Tsvetkov et al. 2016; Platanios et al. 2019).

Other related approaches such as self-paced learning (SPL) (Kumar, Packer, and Koller 2010) and self-paced curriculum learning (Jiang et al. 2015) have also been proposed to show the efficacy of a designed curriculum which adapts dynamically to the pace at which the learner progresses. Other attempts at improving an AL strategy include self-paced active learning (Tang and Huang 2019) in which the authors introduce practical techniques to consider informativeness, representativeness, and easiness of samples while querying for labels. Such methods that only focus on designing a curriculum miss, in general, the opportunity to also leverage the ability of the predictive model which progresses as new labeled data becomes available.

The addition of CL injects human expertise into learning manifested in the design of a curriculum. This is in contrast with previous studies that combined AL with SPL (Tang and Huang 2019; Lin et al. 2018). SPL is inspired by CL but, similarly to AL, relies on querying the model being trained to select instances for labeling.

Our contributions in this paper are twofold: (i) we shed light on the relationship between AL and CL by investigating if AL enforces (or follows) a curriculum. To this end, we monitor and visualize a variety of novel curricula during the AL simulation loop; (ii) We propose a novel method which we call Active Curriculum Learning (ACL). ACL takes advantage of the benefits of both CL (i.e., designing a curriculum for the model to follow) and AL (i.e., choosing samples based on the enhanced ability of the predictive model) at the same time to improve AL. Our preliminary experiments show that the performance of an AL strategy will be improved by deliberately combining AL and CL concepts. This article presents the foundation of this method accompanied by the preliminary results and in our future work we will explore its effectiveness more extensively by implementing more experiments and performing hyper parameter tuning as well as exploring other NLP tasks beyond NER.

2 Novel Curricula

Other than the most explored curriculum features such as sentence length and word frequency some other curricula for measuring diversity, simplicity, and prototypicality of the samples are proposed in (Tsvetkov et al. 2016). Our conjecture is that large-scale language models and also linguistic features can be used to design NLP curricula. We design seven novel curricula which assign a score to a sentence indicating its level of difficulty for a specific NLP task. Then, to acquire a curriculum, sentences are sorted by their corresponding scores. Other than our 7 novel curricula, we also experiment with the following commonly used curricula:

1. **SENT_LEN:** Number of words in a sentence.
2. **WORD_FREQ:** Average of frequency of the words in a sentence (e.g., frequency of the word A is calculated by $\frac{N_A}{\sum_{w \in V} N_w}$ where V is the set of the unique vocabulary of the labeled dataset, and N_w is the number of times the word w has appeared in the labeled dataset).

Our seven novel curricula are as follows:

1. **PARSE_CHILD:** Average of the number of children of words in the sentence parse tree.
2. **GPT_SCORE:** Sentence score according to the GPT2 language model (Radford et al. 2019) calculated as follows: $\sum_k \log(p(w_k))$ where $p(w_k)$ is the probability of kth word of the sentence according to the GPT2 model.
3. **LL_LOSS:** Average loss of the words in a sentence from the Longformer language model (Beltagy, Peters, and Cohan 2020)

For the following four novel curricula, we use the spaCy library (Honnibal and Montani 2017) to replace a word in a sentence with one of its linguistic features. The curriculum value for a sentence is then calculated exactly in the same way

as word frequency but with one of the linguistic features instead of the word itself:

4. **POS**: Simple universal part-of-speech tag such as *PROPN*, *AUX* or *VERB*.
5. **TAG**: Detailed part-of-speech tag such as *NNP*, *VBZ*, *VBG*.
6. **SHAPE**: Shape of the word. For example, shapes of "Apple" and "12a." are "Xxxxx" and "ddx." respectively.
7. **DEP**: Syntactic relation connecting the word to its parent in the dependency parse tree of the sentence (e.g., *amod*, and *compound)*.

3 The Relationship between AL and CL and the Experimental Setup

We set out to answer the following question: *what is the relationship between AL and CL from the lens of the nine curricula?* To answer this question, we simulate two AL strategies as well as random strategy and monitor the curriculum metrics on the most informative samples (from the unlabeled data) chosen for annotation by each sampling strategy and compare them. We use the following two informativeness measures for unlabeled sentences in our AL strategies: (i) min-margin: minimum of margin of the prediction probability for the sentence tokens is considered as the AL score for that sentence. Sentences with lower scores are preferred, (ii) max-entropy: maximum of entropy of the prediction probability for the sentence tokens are considered as the AL score for that sentence and sentences with higher scores are preferred.

For the experiments, we use a single layer Bi-LSTM model (Lample et al. 2016) with the hidden state size of 768, enhanced with a 2-layer feed-forward network in which the number of hidden and output layers' nodes are equal to the number of classes in the dataset. The input to the LSTM model is the word2vec embedding (Mikolov et al. 2013) of sentence words. We use ADAM optimizer (Kingma and Ba 2017) with the batch size of 64 and the learning rate of 5e-4. We experiment with two publicly available English-language NER datasets: *OntoNotes5*[1], and *CoNLL 2003*[2] and use early stopping on the loss of the provided validation sets. Furthermore, we start with 500 randomly selected sentences as the seed data and choose 500 sentences to be labeled in each iteration for a total of 15 iterations.

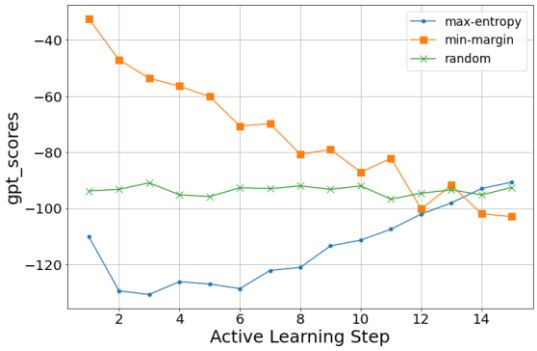

Figure 1: Comparison of the mean of GPT score of sentences added to training data in each iteration between random, min-margin and max-entropy AL strategies for the CoNLL dataset (average of 3 runs).

Figure 1 illustrates the experimental results of monitoring GPT score during AL loop. This figure clearly shows that GPT score of sentences chosen by max-entropy tends to have lower values (i.e., more complex sentences) and min-margin tends to choose sentences with higher values (i.e., simpler sentences) compared to a random strategy. Similar figures for other curricula reveal peculiarities of the different AL strategies compared to the random strategy and other AL strategies. Due to space limitations, instead of including such figures for different strategies, we calculate the following metric which we call Mean Normalized Difference (MND) to quantify how an AL selection strategy differs from a random strategy in choosing the most informative unlabeled data based on a curriculum. This metric is defined as follows:

$$MND = \sum_{i=1}^{n}\sum_{j=1}^{k}\frac{N\left(\psi^{CL}(RN_{ij})\right)-N\left(\psi^{CL}(AL_{ij})\right)}{n\times k} \quad (1)$$

where n is the number of iterations where we add k newly labeled sentences to the labeled dataset, ψ^{CL} calculates the value of the curriculum feature for a sentence, RN_{ij} and AL_{ij} are the j^{th} sentence out of k chosen for annotation in the i^{th} step of the random and active strategies, respectively, $N(x) := \frac{x - r^{CL}_{min}}{r^{CL}_{max} - r^{CL}_{min}}$, $r^{CL}_{min} := \min_{i \in [1,n]} \frac{\sum_{j=1}^{k}\psi^{CL}(R_{ij})}{k}$, and $r^{CL}_{max} := \max_{i \in [1,n]} \sum_{j=1}^{k}\frac{\psi^{CL}(R_{ij})}{k}$. In theory, the MND score can take any value. If the MND score of an AL strategy for a curriculum is close to zero, it means the curriculum values (ψ^{CL}) of the data chosen for

[1] Available at https://catalog.ldc.upenn.edu/LDC2013T19

[2] Available at https://www.clips.uantwerpen.be/conll2003/ner/

annotation are close to that of the random strategy. This, however, does not imply that the same unlabeled data is chosen by the two techniques. Furthermore, large values of the MND score indicate that AL chooses unlabeled data for annotation that have different curriculum scores compared to the random strategy. Since MND is normalized, we can compare the MND score of any two combinations of AL strategy and curriculum score to compare the degree to which they diverge from random strategy.

Experimental Results: Results of the MND scores for different curriculum features on the two experimental datasets are reported in Table 1. In most of these experiments, we observe that there is a difference between how random strategy and AL choose unlabeled dataset from the lens of MND as if AL is mimicking curriculum learning. We also observe that not all AL strategies consistently have the same MND sign for a curriculum on OntoNotes5 and CoNLL 2003 datasets but a noticeable divergence from the random strategy is evident. Table 1 also shows that the largest difference between active and random strategies in following curricula in our experiments is DEP/Min-Margin combination and the smallest difference between them is POS/Max-Entropy combination, both for OntoNotes5 dataset.

	CoNLL 2003		OntoNotes5	
	Min-Margin	Max-Entropy	Min-Margin	Max-Entropy
DEP	-16.7	2	-66.3	-5.5
POS	-18.2	-0.1	-4.2	-5.9
SHAPE	4.1	-3	12.5	4.7
TAG	-14.3	0.3	-4.3	-8.7
GPT_SCORE	-3.3	3.5	-9.0	6.3
LL_LOSS	-1.5	1.1	-18.1	1.7
PARSE_CHILD	3.1	-1.7	18.1	-0.9
SENT_LEN	4.7	-3.9	10.7	-6.2
WORD_FREQ	1.9	-2.4	-0.7	-0.1

Table 1: Mean Normalized Difference of min-margin and max-entropy for the two datasets CoNLL 2003 and OntoNotes5 (average of 15 steps and 3 runs).

4 Active Curriculum Learning (ACL)

To improve the performance of the AL strategies, we introduce a simple yet effective method leveraging both advantages of AL and CL which we call Active Curriculum Learning (ACL). The goal of this proposed method is to benefit from the dynamic nature of AL data selection metric while utilizing experts' knowledge in designing a fixed curriculum. To this end, in each step of the ACL loop, we use the following linear combination of the AL and CL scores to choose the most informative unlabeled data:

$$\psi^{ACL}(s, M_i) := \alpha \frac{\psi^{CL}(s)}{\max_{s \in \boldsymbol{D_i^U}} |\psi^{CL}(s)|} + \beta \frac{\psi^{AL}(s, M_i)}{\max_{s \in \boldsymbol{D_i^U}} |\psi^{AL}(s, M_i)|} \quad (2)$$

where $\boldsymbol{D_i^U}$ is the set of unlabeled sentences in step i of the ACL loop, α and β are the two parameters that control the combination of AL and CL scores, $\psi^{AL}(s, M_i)$ is the AL score (i.e., informativeness) of sentence s according to the predictive model M_i trained on $\boldsymbol{D_i^L}$ at step i.

The overall steps of the ACL algorithm are presented in Algorithm 3. Similar to the AL algorithm, the min-margin based strategy favors sentences with lower ψ^{ACL} for annotation and the opposite is true for the max-entropy based approach.

1. Seed labeled data $\boldsymbol{D^L} = \{(x_1, y_1), \dots, (x_m, y_m)\}$
2. Unlabeled data $\boldsymbol{D^U} = \{x_{m+1}, \dots, x_n\}$
3. While the stopping criterion is not met:
   - 3.1. $\boldsymbol{I} :=$ the set of k examples in $\boldsymbol{D^U}$ with the best score based on $\boldsymbol{\psi^{ACL}}\}$
   - 3.2. $\boldsymbol{D^U} := \boldsymbol{D^U} \setminus \boldsymbol{I}$; $\boldsymbol{D^L} := \boldsymbol{D^L} \cup \boldsymbol{L(I)}$
   - 3.3. Fine-tune or train the model $\boldsymbol{M_i}$ on $\boldsymbol{D^L}$

Algorithm 3. Steps of the ACL algorithm where $L(\boldsymbol{I})$ denotes the set $\boldsymbol{I}$ after annotation.

Experimental Results: We use the training setup of section 3 and perform token classification on CoNLL 2003 and OntoNotes5 datasets using the ACL algorithm. To evaluate the performance of ACL, for each AL metric and dataset combination, we run 18 ACL experiments where $\alpha = 1$, $\beta = 0.5$ or $\beta = -0.5$ for the 9 curricula, and also one AL experiment where $\alpha = 1$ and $\beta = 0$. Since the main focus of this article is to demonstrate if the introduction of a curriculum adds value to the performance of the active strategies, we select these hyper parameters in such a way that the effects of the active strategies are still dominant in the proposed model.

In each step of the ACL loop, we measure the token-level F1 score (for higher granularity) of the provided test set using the trained model in that step. Table 2 reports the average of F1 scores for the top 5 ACL combinations as well as the active learner (α = 1, β = 0) across all runs (3) and steps (15). In all of our experiments, the top 5 ACL

OntoNotes5					
Min-Margin			Max-Entropy		
CM	β	F1	CM	β	F1
GPT_SCORE	0.5	0.4	LL_LOSS	-0.5	0.48
PARSE_CHILD	-0.5	0.4	DEP	-0.5	0.45
SENT_LEN	-0.5	0.38	POS	-0.5	0.43
LL_LOSS	0.5	0.37	WORD_FREQ	-0.5	0.43
TAG	-0.5	0.33	SENT_LEN	-0.5	0.43
-	0	0.23	-	0	0.36
CoNLL 2003					
Min-Margin			Max-Entropy		
CM	β	F1	CM	β	F1
LL_LOSS	0.5	0.65	SENT_LEN	0.5	0.67
GPT_SCORE	0.5	0.63	LL_LOSS	0.5	0.66
PARSE_CHILD	-0.5	0.63	WORD_FREQ	-0.5	0.66
SENT_LEN	-0.5	0.62	PARSE_CHILD	0.5	0.66
DEP	0.5	0.61	GPT_SCORE	-0.5	0.66
-	0	0.57	-	0	0.64

Table 2: ACL results for OntoNotes5 and CoNLL datasets. The last row for each experiment corresponds to the AL strategy. Curriculum Metric is denoted by CM, F1 is the average of F1 score across all 15 steps and 3 runs. For all experiments we have $\alpha = 1$.

combinations always outperformed AL for that dataset. In particular our curricula based on deep language models (GPT_SCORE and LL_LOSS) are appearing frequently in Table 2 indicating their utility.

5 Conclusions and Future Work

To the best of our knowledge, this is the first work to investigate and reveal the relationship between two closely related machine learning techniques namely, AL and CL. We observed that AL in fact follows a curriculum as it progresses through its iterations compared to the random strategy.

This is also the first work to take advantage of the benefits of both CL (i.e., designing a curriculum for the model to learn) and AL (i.e., choosing samples based on the improved ability of the predictive model) to improve AL in a unified model.

In our future work, we are interested in understanding in detail how CL helps AL, and exploring model-based techniques of combining AL and CL rather than a fixed set of weights for α and β. Another interesting question to investigate is to conduct similar experiments for other NLP tasks or using multiple curricula together with AL can be beneficial in reducing the annotation cost. We are also interested in investigating our novel curricula on their own in an isolated CL setting.

References

Beltagy, Iz, Matthew E. Peters, and Arman Cohan. 2020. "Longformer: The Long-Document Transformer." *ArXiv:2004.05150 [Cs]*, December. http://arxiv.org/abs/2004.05150.

Bengio, Yoshua, Jerome Louradour, Ronan Collobert, and Jason Weston. 2009. "Curriculum Learning." In *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. Https://Doi.Org/10.1145/1553374.1553380. Montreal, Quebec, Canada: Association for Computing Machinery. https://doi.org/10.1145/1553374.1553380.

Braun, Stefan, Daniel Neil, and Shih-Chii Liu. 2016. "A Curriculum Learning Method for Improved Noise Robustness in Automatic Speech Recognition." *ArXiv:1606.06864 [Cs]*, September. http://arxiv.org/abs/1606.06864.

Gui, Liangke, Tadas Baltrusaitis, and Louis-Philippe Morency. 2017. "Curriculum Learning for Facial Expression Recognition." In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 505–11. Washington, DC, DC, USA: IEEE. https://doi.org/10.1109/FG.2017.68.

Honnibal, Matthew, and Ines Montani. 2017. "SpaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing."

Jiang, Lu, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander G. Hauptmann. 2015. "Self-Paced Curriculum Learning." In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2694–2700. AAAI'15. Austin, Texas: AAAI Press.

Kingma, Diederik P., and Jimmy Ba. 2017. "Adam: A Method for Stochastic Optimization." *ArXiv:1412.6980 [Cs]*, January. http://arxiv.org/abs/1412.6980.

Kumar, M., Benjamin Packer, and Daphne Koller. 2010. "Self-Paced Learning for Latent Variable Models." In *Advances in Neural Information Processing Systems*. Vol. 23. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2010/file/e57c6b956a6521b28495f2886ca0977a-Paper.pdf.

Lample, Guillaume, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. "Neural Architectures for Named Entity Recognition." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 260–70. San Diego, California: Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-1030.

Lin, Liang, Keze Wang, Deyu Meng, Wangmeng Zuo, and Lei Zhang. 2018. "Active Self-Paced Learning for Cost-Effective and Progressive Face Identification." *IEEE Transactions on Pattern*

Analysis and Machine Intelligence 40 (1): 7–19. https://doi.org/10.1109/TPAMI.2017.2652459.

Long, Bo, Jiang Bian, Olivier Chapelle, Ya Zhang, Yoshiyuki Inagaki, and Yi Chang. 2014. "Active Learning for Ranking through Expected Loss Optimization." *IEEE Transactions on Knowledge and Data Engineering* 27 (5): 1180–91.

Melville, Prem, and Raymond J. Mooney. 2004. "Diverse Ensembles for Active Learning." In *Twenty-First International Conference on Machine Learning - ICML '04*, 74. Banff, Alberta, Canada: ACM Press. https://doi.org/10.1145/1015330.1015385.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Efficient Estimation of Word Representations in Vector Space." *ArXiv Preprint ArXiv:1301.3781*.

Platanios, Emmanouil Antonios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. 2019. "Competence-Based Curriculum Learning for Neural Machine Translation." In *Proceedings of the 2019 Conference of the North {A}merican Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1162–117. Minneapolis, Minnesota: Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1119.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. "Language Models Are Unsupervised Multitask Learners." *Ilya* (blog). 2019.

Rao, Vijjini Anvesh, Kaveri Anuranjana, and Radhika Mamidi. 2020. "A Sentiwordnet Strategy for Curriculum Learning in Sentiment Analysis." In *Natural Language Processing and Information Systems*, edited by Elisabeth Métais, Farid Meziane, Helmut Horacek, and Philipp Cimiano, 12089:170–78. Lecture Notes in Computer Science. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-51310-8_16.

Sachan, Mrinmaya, and Eric Xing. 2016. "Easy Questions First? A Case Study on Curriculum Learning for Question Answering." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 453–63. Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-1043.

Settles, Burr. 2012. "Active Learning." *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6 (1): 1–114.

Settles, Burr, and Mark Craven. 2008. "An Analysis of Active Learning Strategies for Sequence Labeling Tasks." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing - EMNLP '08*, 1070. Honolulu, Hawaii: Association for Computational Linguistics. https://doi.org/10.3115/1613715.1613855.

Soviany, Petru, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2021. "Curriculum Learning: A Survey." *ArXiv:2101.10382 [Cs]*, January. http://arxiv.org/abs/2101.10382.

Tang, Ying-Peng, and Sheng-Jun Huang. 2019. "Self-Paced Active Learning: Query the Right Thing at the Right Time." In *Proceedings of the AAAI Conference on Artificial Intelligence*, 5117–24. https://doi.org/10.1609/aaai.v33i01.33015117.

Tsvetkov, Yulia, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. 2016. "Learning the Curriculum with Bayesian Optimization for Task-Specific Word Representation Learning." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 130–39. Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-1013.

Xu, Benfeng, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. 2020. "Curriculum Learning for Natural Language Understanding." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6095–6104. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.542.

Zhang, Ye, Matthew Lease, and Byron C. Wallace. 2017. "Active Discriminative Text Representation Learning." In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 3386–92. AAAI'17. AAAI Press.