Active Learning for Abstractive Text Summarization via LLM-Determined Curriculum and Certainty Gain Maximization

Dongyuan Li[1], **Ying Zhang**[2,*], **Zhen Wang**[1], **Shiyin Tan**[1], **Satoshi Kosugi**[1], **Manabu Okumura**[1]
[1]Tokyo Institute of Technology, Tokyo, Japan
[2]RIKEN Center for Advanced Intelligence Project
{lidy, wzh, tanshiyin, kosugi, oku}@lr.pi.titech.ac.jp, ying.zhang@riken.jp

Abstract

For abstractive text summarization (ATS), laborious data annotation and time-consuming model training become two high walls, hindering its further progress. Active Learning (AL), selecting a few informative instances for annotation and model training, sheds light on solving these issues. However, only few AL-based studies focus on ATS and suffer from low stability, effectiveness, and efficiency. To solve the problems, we propose a novel LLM-determined curriculum active learning framework. Firstly, we design a prompt to ask large language models to rate the difficulty of instances, which guides the model to train on from easier to harder instances. Secondly, we design a novel AL strategy, *i.e.*, Certainty Gain Maximization, enabling to select instances whose distribution aligns well with the overall distribution. Experiments show that our method can improve the stability, effectiveness, and efficiency of the ATS backbones. Code is available on Github [1].

1 Introduction

Abstractive text summarization (ATS) aims to condense a lengthy document into a concise yet informative summary, preserving the essential information of the original document (Sun et al., 2023). Benefiting from the Transformer (Vaswani et al., 2017) and pre-training strategies (Lewis et al., 2020; Zhang et al., 2020), a new paradigm, *i.e.*, pre-training and fine-tuning Transformer-based large language models (LLMs), achieves state-of-the-art results in ATS (Xia et al., 2024). However, as the number of parameters in LLMs increases, processing LLMs for ATS becomes impractical due to the huge time and computational resources required. Moreover, similar to other natural language generation (NLG) tasks, fine-tuning LLMs for ATS requires large-scale datasets, which always necessitate labor intensive cost and expert knowledge, further impeding ATS (Tsvigun et al., 2023).

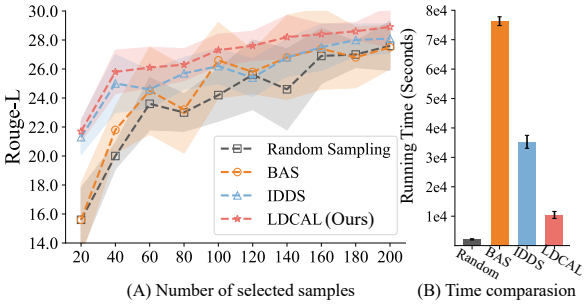

Figure 1: BART-base model with various AL methods on the AESLC dataset. (A) Performance on the test set, where AL methods select instances with a query size of 20 for training. (B) Running time over 200 instances.

Active learning (AL) sheds light on alleviating these issues (Settles, 2009). To reduce human labor and time consumption, AL aims to select a few unlabeled yet informative instances for annotation and model training (Li et al., 2024a). Specifically, BAS (Gidiotis et al., 2024) and IDDS (Tsvigun et al., 2023) are recent best-performing AL methods for ATS. BAS measures the uncertainty of a model's prediction for each instance and annotates an instance with the highest uncertainty. Given a model and an input sentence, BAS randomly drops certain parameters in the model to generate multiple outputs. The variance of the BLEU scores (Papineni et al., 2002) between the input and each output is considered as the uncertainty of the given input instance. In contrast, IDDS (Tsvigun et al., 2023) aims to acquire representative instances for model training. It selects instances that are dissimilar to those already selected but are the most representative among unselected instances.

Despite the great success of the AL methods in ATS, there are still three issues. *i) Unstable Performance*. As shown in Figure 1(A), as the number of selected instances increases, previous methods ex-

*Corresponding author.
[1]https://github.com/Clearloveyuan/EMNLP-LDCAL/tree/main

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 8959–8971
November 12-16, 2024

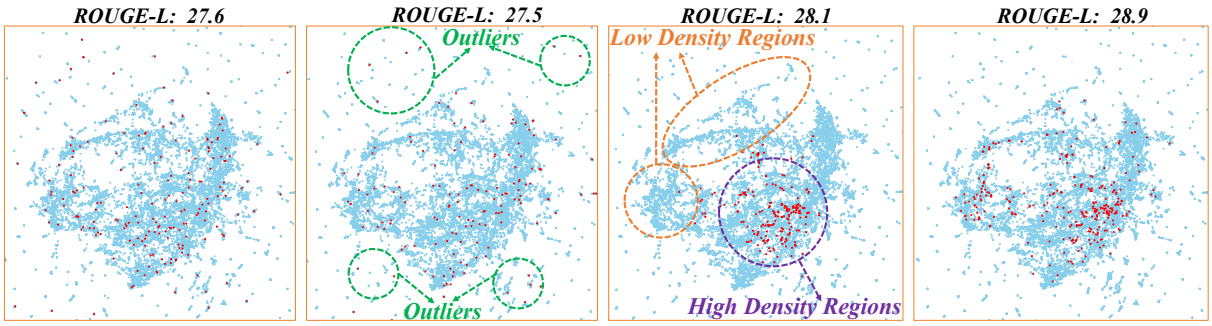

Figure 2: UMAP visualization (McInnes et al., 2020) of selected instances from BART with various AL methods on the AESLC dataset. Blue and red points represent 14,200 unselected and 200 selected instances, respectively.

hibit unstable fluctuations on model performance. This instability often prevents the model from being fully optimized, thereby hindering its performance. *ii) Misaligned Distribution*. As shown in Figure 2, BAS selects outliers because such instances have high uncertainty. While IDDS is designed to avoid outliers, it neglects instances in low density regions. Because the distribution of selected instances differs from that of the entire dataset, performance improvement is hampered. *iii) High Computational Cost*. In Figure 1(B), among all AL methods, BAS is the most time-consuming one, because it visits the remaining unselected instances in each iteration. Although IDDS visits all instances in advance without inference at each iteration, its time complexity is $\mathcal{O}(n^2)$ with n being the number of instances, which is a huge overhead for large-scale datasets.

To address the above issues, we make two contributions. ***Firstly***, to address unstable performance, we propose LLM-Determined Curriculum Active Learning (LDCAL). We found that the AL-based training process is highly sensitive to the training order of instances, which easily leads to the model's unstable performance. Drawing inspiration from the way humans realize learning stability by gradually increasing the difficulty of learning content, we employ curriculum learning by gradually increasing the complexity of instances from easy to hard during training. Unlike previous studies that always determine document difficulty based on traditional curriculum, we design a prompt to directly ask LLMs for document difficulty. By following the curriculum determined by LLMs, stable performance can be achieved. ***Secondly***, to address the misaligned distribution and high computational cost, we propose a novel AL strategy named Certainty Gain Maximization. Certainty measures how well unselected instances are represented by selected ones, and we select instances that maximize the certainty gain for the unselected instances. By using Certainty Gain Maximization, we can select instances whose distribution aligns well with the overall distribution in Figure 2(D). In addition, the time complexity of Certainty Gain Maximization is $\mathcal{O}(nk)$ ($k \ll n$), where n and k represent the number of unselected and selected instances, respectively. It is significantly lower than that of existing methods in Figure 1(B). The main contributions of this study are summarized as follows:

- To the best of our knowledge, we are the first to propose a curriculum active learning framework for ATS, where LLM-determined curriculum enhances the stability and effectiveness of the AL processes.
- We design a certainty gain-based AL strategy to select diverse instances with uneven distribution, further improving effectiveness and efficiency.
- Experiments conducted on three ATS benchmarks demonstrate the stability, effectiveness, and efficiency of our method LDCAL.

2 Related Work

Abstractive Text Summarization. Seq2seq models (Sutskever et al., 2014) along with the attention mechanism (Bahdanau et al., 2015) have made neural networks a primary tool for ATS. Then, pre-trained language models, based on Transformer, achieve SOTA performance on various tasks (Zhang et al., 2024; Li et al., 2023b; You et al., 2022). Pre-training and fine-tuning has become a mainstream paradigm for ATS (Lewis et al., 2020; Zhang et al., 2020; Qi et al., 2020; Guo et al., 2021). We aim to explore an efficient and effective method based on this paradigm. Although LLM-based zero-shot and in-context learning (Chhabra et al., 2024; Jain et al., 2023) inspired many interests recently, which is out scope of this work.

Active Learning (AL). AL selects only a few informative instances to annotate for model training, aiming to reduce human annotation efforts and model computation costs (Margatina and Aletras, 2023). Recent work almost leverages AL for text classification (Wu et al., 2022; Yu et al., 2022; Schröder et al., 2023) and sequence tagging tasks (Radmard et al., 2021; Yuan et al., 2022), with little focus on NLG tasks. Current AL-based methods for NLG can be broadly classified into two categories: *uncertainty-based* and *diversity-based* methods (Zhang et al., 2022b). Specifically, *uncertainty-based* methods identify uncertain instances as those that prompt models to produce diverse outputs, assuming these instances offer more valuable information for model training than others (Raj and Bach, 2022). For example, for each source instance, Wang et al. (2019) repeat the following two processes several times: 1) randomly drop certain model parameters and 2) infer the translation probability between the source and the generated translation. After repetition, the variance of these translation probabilities is considered as the uncertainty for the source instance. Following their work, Xiao et al. (2020) use the variance of the BLEU score as an uncertainty measure in the German-English translation, and Gidiotis et al. (2024) adopt this for ATS. *Diversity-based methods* assume that diverse instances can approximate the original data distribution, suggesting the selection of the most representative instances for model training (Kim et al., 2006). For example, to avoid outliers, Tsvigun et al. (2023) select instances that are dissimilar to already selected ones while ensuring similarity to unselected instances. Xia et al. (2024) calculate the average Jensen-Shannon divergence between each unselected instance and selected instances as a diversity score, to alleviate hallucinations. Unlike these methods, LDCAL involves curriculum learning and incorporates a new certainty gain maximization strategy to select instances from various regions of the ATS dataset.

Curriculum Learning (CL). As a training strategy, CL gradually increases the complexity of the data, *i.e.*, easy-to-hard, during the training process for faster convergence and better performance (Bengio et al., 2009). CL achieves great success in various NLG tasks, *e.g.*, machine translation (Zhang et al., 2021; Mohiuddin et al., 2022), medical report generation (Liu et al., 2021; Zhang et al., 2022a), dialogue generation (Zhu et al., 2021), and language modeling (Mi, 2023). *"How to define the difficulty of training instances"* is a fundamental issue in CL. For example, sentence length (Kocmi and Bojar, 2017), word rarity (Zhang et al., 2018), and norm of word embedding (Liu et al., 2020) are considered as difficulty measurements for machine translation. For ATS, sentence length (Kano et al., 2021), self-defined confidence scores (Sotudeh et al., 2022; Sun et al., 2023), and ROUGE scores (Magooda and Litman, 2021) are used. Instead of defining text difficulty from classic curriculum, we design prompts to directly ask LLMs for the difficulty.

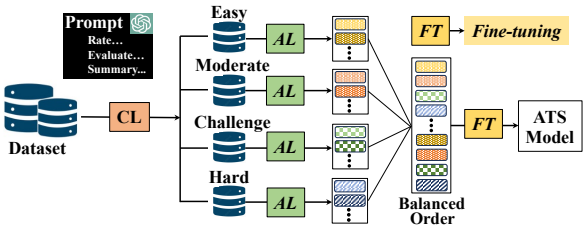

Figure 3: Overall pipeline of the LDCAL framework.

3 Methodologies

We first introduce the detailed process of the proposed LDCAL framework in Section 3.1. Then, we introduce the LLM-determined curriculum learning strategy for difficulty measurement in Section 3.2. Finally, in Section 3.3, we describe the novel AL strategy, *i.e.*, the certainty gain maximization.

3.1 Curriculum Active Learning Framework

The overview of our LDCAL for ATS is shown in Figure 3. We first introduce the necessary notations and then present the detailed process.

Notations. We define $\mathcal{D}_{\textbf{pool}}=\{\boldsymbol{x}^{(i)}\}_{i=0}^{N}$ as a dataset pool containing N unlabeled documents, where $\boldsymbol{x}^{(i)}=(x_1^{(i)},\cdots,x_n^{(i)})$ represents the i-th unlabeled document with n tokens. Using an instance difficulty measurement function $\mathcal{I}()$ and an AL-based instance acquisition function $\mathcal{A}()$, we select a few instances from $\mathcal{D}_{\textbf{pool}}$ for annotation and then add them to a labeled pool $\mathcal{D}_{\textbf{lab}} = \{(\boldsymbol{x}^{(i)},\boldsymbol{y}^{(i)})\}_{i=0}^{M}$, where $\boldsymbol{y}^{(i)}=(y_1^{(i)},\cdots,y_m^{(i)})$ is the i-th annotated summary with m tokens and M denotes the size of $\mathcal{D}_{\textbf{lab}}$. Given an LLM with weights $\boldsymbol{W}$, we fine-tune it with $\mathcal{D}_{\textbf{lab}}$ in a balanced easy-to-hard order.

LDCAL Framework. LDCAL mainly contains four processes, and we will introduce them in order.

1) **Curriculum Learning with Instance Division.** Based on the designed documents' difficulty

Prompt

You are tasked with evaluating the ease of generating a summary for a given English text. Rate the difficulty of summarizing the text on a scale from 1.0 to 4.0. Provide a comprehensive rationale for your assessment. Consider factors such as the complexity of the language and the coherence of the text. Explain how these elements influence the difficulty level of the generating a summary.

Figure 4: LLM-determined instance difficulty for CL.

measurement function $\mathcal{I}()$, we set four difficulty levels for documents and separate $\mathcal{D}_{\textbf{pool}}$ into four corresponding sub-pools, formulated by:

$$\mathcal{D}_{\textbf{pool}}^{\textbf{Easy}}, \mathcal{D}_{\textbf{pool}}^{\textbf{Mode.}}, \mathcal{D}_{\textbf{pool}}^{\textbf{Chal.}}, \mathcal{D}_{\textbf{pool}}^{\textbf{Hard}} = \mathcal{I}(\mathcal{D}_{\textbf{pool}}), \quad (1)$$

where "Mode." and "Chal." are abbreviations for moderate and challenging, respectively. For more details on $\mathcal{I}()$, please refer to Section 3.2.

2) **Active Learning for Instance Selection.** We design a diversity-based acquisition function $\mathcal{A}()$ to select a few representative instances in parallel from four unlabeled sub-pools as follows:

$$\{\boldsymbol{x}^{(i)}\}_{i=1}^{k} = \mathcal{A}(\mathcal{D}_{\textbf{pool}}^{*}, M^{*}), \quad (2)$$

where $* \in \{\text{Easy, Mode., Chal., Hard}\}$, M^{*} is the number of selected instances from $\mathcal{D}_{\textbf{pool}}^{*}$ and k is the total query size. For more details on $\mathcal{A}()$, please refer to Section 3.3.

3) **Human Annotation.** Each selected instance $\boldsymbol{x}$ is annotated with an abstractive text summary $\boldsymbol{y}$ by human annotators. Then, the selected document $\boldsymbol{x}$ is removed from the unlabeled pool $\mathcal{D}_{\textbf{pool}}^{*}$ and is added to the labeled pool as:

$$\mathcal{D}_{\textbf{pool}}^{*} := \mathcal{D}_{\textbf{pool}}^{*} \backslash \boldsymbol{x}; \quad \mathcal{D}_{\textbf{lab}} := \mathcal{D}_{\textbf{lab}} \cup \{(\boldsymbol{x}, \boldsymbol{y})\}. \quad (3)$$

4) **Model Fine-tuning.** $\mathcal{D}_{\textbf{lab}}$ is used to update the weights of an LLM pre-trained language model in a balanced easy-to-hard order as follows:

$$\widetilde{\boldsymbol{W}} = \arg\min_{\boldsymbol{W}} \mathcal{L}\left(\mathcal{D}_{\textbf{lab}}, \boldsymbol{W}\right), \quad (4)$$

where $\mathcal{L}$ is the loss used for fine-tuning, *e.g.*, supervised fine-tuning objective (Li et al., 2024b).

Then, we introduce two core modules of LDCAL.

3.2 CL with Instance Division

Humans' learning process typically occurs in an easy-to-hard order, as directly learning a difficult concept may exceed their abilities to understand it. To ensure that the model learns as much knowledge as possible from limited instances, we design two CL strategies to mimic this process, based on understanding the difficulty of text from the perspectives of *Classic manners* and *LLMs*, respectively.

LLM-Determined CL. Considering that LLMs increasingly outperform humans in many tasks (Minaee et al., 2024) and can understand difficulty of text from a model's perspective, we design a LLM-determined CL strategy. Specifically, we create a prompt, as shown in Figure 4, to directly ask an LLM, *i.e.*, GPT-3.5, to rate the difficulty in understanding each document. We then use these scores to measure document difficulty. Considering that LLM resources may be limited in reality, we also propose a Classic CL and use it as a baseline (CCAL) to compare with LLM-determined CL.

Classic CL. Current studies determine the difficulty of understanding a document based on reading experience like sentence length or manually defined metrics like Perplexity, achieving improvements in many tasks (Gao et al., 2024). Thus, we summarize traditional difficulty-based CL methods for measuring document difficulty in three aspects.

1) **Lexical Complexity.** i) Word_freq: Average frequency of all words in the document; ii) POS: Average frequency of simple universal part-of-speech tags in the document, such as PROPN and VERB; iii) TAG: Average frequency of detailed part-of-speech tags in the document like NNP, VBZ, and VBG, annotated using spaCy.[2]

2) **Syntactic Structure Complexity.** i) Sent_len: Number of words in the text; ii) Parse_child: Average of the number of children of words in the sentence parse tree. iii) DEP: Syntactic relation connecting a word to its parent in the dependency parse tree of the sentence, *e.g.*, amod and compound (Jafarpour et al., 2021).

3) **Contextual Semantic Complexity.** i) LL_loss: Average loss of words in sentences from Longformer (Beltagy et al., 2020); ii) GPT_score: It is based on the likelihood of the sentence being generated by GPT-2, with lower scores indicating more difficulty (Radford et al., 2019).

Finally, we use the normalized weighted sum of above-mentioned scores as the difficulty score for each document, with smaller scores being simpler.

[2] https://github.com/explosion/spaCy.

Instance Division. After obtaining document difficulty scores by CL, we divide unlabeled instances $\mathcal{D}_{\textbf{pool}}$ into four sub-pools, as formulated in Eq.(1). In Figure 3, we further partition selected instances of these four sub-pools into several blocks, where each block contains an equal number of easy, moderate, challenging, and hard instances. We utilize such balanced blocks for model training, with instances in each block sorted from easy to hard. We refer to this as the *balanced easy-to-hard order*.

3.3 Active Learning for Instance Selection

We argue that uncertainty-based methods tend to select outliers that have small value for model training. Diversity-based methods overlook uneven density of instances, repeatedly selecting instances from high-density areas while neglecting those from low-density areas. To relieve these issues, we propose the certainty gain maximization, aiming to maximize the diversity of selected instances and avoid outliers, *i.e.*, select representative instances from both high-density and low-density regions.

Specifically, we first obtain the embedding of each instance $\boldsymbol{x}^{(u)}$ using an encoder $\Phi()$, denoted as $\Phi(\boldsymbol{x}^{(u)})=\boldsymbol{e}^{(u)}$, *e.g.*, using the [CLS] pooled sequence embedding from BERT (Devlin et al., 2019). Then, a *certainty score* (CER) for each unlabeled instance $\boldsymbol{x}^{(u)}$ is defined as how well it can be represented by those already labeled instances as:

$$\text{CER}(\boldsymbol{x}^{(u)}) = \max_{\boldsymbol{x}^{(\ell)} \in \mathcal{D}_{\textbf{lab}}} \text{Sim}(\boldsymbol{x}^{(u)}, \boldsymbol{x}^{(\ell)}), \quad (5)$$

where $\text{Sim}()$ can be any similarity measure, *e.g.*, a scalar product $\text{Sim}(\boldsymbol{x}^{(u)}, \boldsymbol{x}^{(\ell)}) = <\boldsymbol{e}^{(u)}, \boldsymbol{e}^{(\ell)}>$. We assume the higher the certainty score between $\boldsymbol{e}^{(u)}$ and $\boldsymbol{e}^{(\ell)}$, the better $\boldsymbol{x}^{(u)}$ can be represented by $\boldsymbol{x}^{(\ell)}$.

To ensure that the final $\mathcal{D}_{\textbf{lab}}$ maximizes the overall CER, we first define the *Certainty Gain* (CERG) for each unlabeled instance. Specifically, if any one unlabeled instance $\boldsymbol{x}^{(s)}$ is selected for annotation, its impact on the certainty gain of any other unlabeled instances $\boldsymbol{x}^{(u)}$ can be formulated by:

$$\text{CERG}(\boldsymbol{x}^{(s)}, \boldsymbol{x}^{(u)}) = \max\{\text{Sim}(\boldsymbol{x}^{(s)}, \boldsymbol{x}^{(u)}) - \text{CER}(\boldsymbol{x}^{(u)}), 0\}, \quad (6)$$

where we avoid negative certainty gain by setting 0 to represent that there is no certainty gain for $\boldsymbol{x}^{(u)}$.

Due to uneven instance distribution in ATS, solely focusing on selecting instances to maximize the certainty gain of all unlabeled instances would bias the sampling towards high-density regions. To address this issue, we introduce the average certainty gain maximization (ACERG) as follows:

$$\text{ACERG}(\boldsymbol{x}^{(s)}) = \frac{1}{L} \sum_{\boldsymbol{x}^{(u)} \in \mathcal{D}_{\textbf{pool}}} \text{CERG}(\boldsymbol{x}^{(s)}, \boldsymbol{x}^{(u)}), \quad (7)$$

where L represents the total number of unlabeled instances with certainty gain greater than 0, *i.e.*, $\text{CERG}(\boldsymbol{x}^{(s)}, \boldsymbol{x}^{(u)}) > 0$. Since outliers tend to have a relatively small sum of certainty gain scores, we introduce a threshold $\beta = 50$ to avoid selecting outliers. In practice, we select the same number of instances that have the highest ACERG from each sub-pool (Eq.(2)), which is simple yet effective.

Time Complexity. Compared to uncertainty-based acquisition strategies, LDCAL does not require retraining an acquisition model during each AL iteration, as instance representations and similarities can be calculated by Eq.(5) before starting the AL annotation process. Compared to existing diversity-based acquisition strategies, such as IDDS and coreset (Sener and Savarese, 2017), which have a time complexity of $\mathcal{O}(n^2)$ with n being the number of unlabeled instances, LDCAL has an almost linear time complexity of $\mathcal{O}(nk)$, where k is the number of annotated instances with $k \ll n$.

Table 1: Dataset statistics of the used Abstract Text Summarization Datasets. Specifically, # Ins. represents number of instances, Doc. len. represents the averaged length of documents and Sum. len. represents the averaged length of summaries.

Dataset	Subset	# Ins.	Doc. len.	Sum. len.
Gigaword	Train	200	40.8	13.3
	Test	2K	38.6	12.5
AESLC	Train	14.4K	142.4	7.8
	Test	1.9K	143.8	7.9
WikiHow	Train	184.6K	377.5	77.2
	Test	1K	386.9	77.0
Pubmed	Train	119.1K	495.4	263.9
	Test	6.7K	509.5	268.0

4 Experiments and Discussions

4.1 Experimental Settings

Datasets. Following Tsvigun et al. (2023), we evaluated ATS backbones using three widely-used datasets. AESLC contains *short* emails with their subject lines as summaries (Zhang and Tetreault, 2019). WikiHow contains *medium-sized* articles with their headlines as summaries (Koupaee and Wang, 2018). PubMed contains *long* scientific articles with their abstracts as summaries (Cohan et al.,

2018). Table 1 displays the numbers of instances in both training and test sets, along with the average token counts for documents and summaries. All datasets are in English. In line with Tsvigun et al. (2023), we have refined the WikiHow dataset by removing noisy instances. Specifically, we exclude those with documents containing ten or fewer tokens and summaries with three or fewer tokens.

Different human-based curriculums have different ranges. For example, in the AESLC dataset, the range of word length is 25 to 3136, while the range of the POS score is 3 to 948. Therefore, we first normalize each curriculum score and then add them up as the difficulty score for the document. Finally, we arranged the samples in ascending order according to the difficulty score of the document and evenly divided them into four sub-pools. For example, we have total number of 14,436 samples in the AESLC dataset and we divide it as $|\mathcal{D}_{\textbf{pool}}^{\textbf{Easy}}| = 3,609$, $|\mathcal{D}_{\textbf{pool}}^{\textbf{Mode.}}| = 3,609$, $|\mathcal{D}_{\textbf{pool}}^{\textbf{Chal.}}| = 3,609$, $|\mathcal{D}_{\textbf{pool}}^{\textbf{Hard}}| = 3,609$. For LDCAL, we follow the same way to equally separate instances into four equal sizes according to their difficulty scores, which is obtained from GPT-3.5.

Baselines and Evaluation Metrics. We selected six classic and state-of-the-art (SOTA) AL baselines for comparison, including *uncertainty-based methods*: NSP (Xiao et al., 2020), ENSP (Lyu et al., 2020), and BAS (Gidiotis et al., 2024); *diversity-based methods*: core-set (Sener and Savarese, 2017) and IDDS (Tsvigun et al., 2023); Random Sampling, and Classic CL with certainty gain-based AL strategy, *i.e.*, CCAL. Since there are no hybrid active learning baselines for text summarization, we do not compare with those general active learning baselines in the classification field (Li et al., 2023a, 2024a). To evaluate the quality of generated summaries, we used the commonly adopted ROUGE metrics (Lin, 2004). Considering that hallucination of the generated summaries is one of the most crucial problems in ATS (Nan et al., 2021; Goyal et al., 2022), we measured the factual consistency of the generated summaries with the original documents by SummaC$_{ZS}$ (Laban et al., 2022).

Implementation Details. For uncertainty-based methods, we first randomly selected ten annotated instances to fine-tune the ATS model. We then used the model to infer the uncertainty scores of unlabeled instances for the first AL iteration. Other baselines do not require fine-tuning the model to make a query for the first AL iteration. In each AL iteration, we selected the top 20 instances (query size) from $\mathcal{D}_{\textbf{pool}}$ based on the uncertainty or diverse score obtained by an acquisition strategy. The selected instances with their ground-truth summaries are added to $\mathcal{D}_{\textbf{lab}}$. Following previous work on annotation emulation (Shen et al., 2017; Shelmanov et al., 2021), we used ground-truth summaries to emulate human-annotated summaries. Finally, we fine-tuned an ATS model from scratch and evaluated it on a held-out test set. We ran the AL loop for 10 iterations with 200 instances in total for each experiment. We also report the performance of ATS backbones in each iteration to show the dynamics of the model performance, depending on the invested annotation effort. To ensure a fair comparison with previous studies, we used either off-the-shelf software packages or the code provided by the respective authors. Each model was run ten times, and the average performance across these runs is reported as the final result.

Backbones and Hyperparameters. We conducted experiments using the SOTA ATS backbones: BART-base (Lewis et al., 2020) and PEGASUS-large (Zhang et al., 2020). We tuned the hyperparameters by the ROUGE-L score on the subset of the Gigaword dataset (Graff et al., 2003).

4.2 Quantitative Evaluation

4.2.1 Comparison with SOTA Baselines

High Effectiveness. In Table 2, to measure the quality and factual consistency of the generated abstractive summaries, we report ROUGE and SummaC$_{ZS}$ scores for baselines. We have the following three main findings. **Firstly**, LDCAL and CCAL achieve the best and second-best performance compared to all baselines, showing the effectiveness of combing CL with AL. **Secondly**, NSP, ENSP, and BAS underperform Random Sampling since they often infer outliers with high uncertainty scores and select the outliers for model training. It is demonstrated in Section 4.3.3. **Thirdly**, IDDS outperforms Random Sampling because it can avoid outliers. However, IDDS underperforms LDCAL and CCAL, as it tends to select redundant instances from high-density regions and neglects instances from low-density regions. As mentioned in § 4.3.3, when we add a few instances from low-density regions to IDDS, its performance improves.

High Efficiency. In Figure 5, when BART-base is selected as the backbone, LDCAL and CCAL show high efficiency compared to all baselines.

Model	Method	AESLC				WikiHow				PubMed			
		ROUGE-1	ROUGE-2	ROUGE-L	SummaC$_{ZS}$	ROUGE-1	ROUGE-2	ROUGE-L	SummaC$_{ZS}$	ROUGE-1	ROUGE-2	ROUGE-L	SummaC$_{ZS}$
BART Base	NSP	25.4±3.1	13.9±1.7	25.4±1.0	0.11±0.02	28.0±0.3	8.8±0.1	20.3±0.1	-0.43±0.04	26.4±0.2	9.6±0.1	17.0±0.3	-0.32±0.03
	ENSP	26.4±3.2	14.6±1.7	27.1±0.9	0.10±0.03	27.9±0.2	8.8±0.1	20.3±0.2	-0.45±0.04	28.0±0.2	10.1±0.1	17.3±0.4	-0.33±0.04
	BAS	28.1±2.2	14.7±1.8	27.5±1.5	0.13±0.01	27.3±0.4	9.5±0.1	20.7±0.1	-0.44±0.03	27.7±0.2	9.8±0.2	17.1±0.1	-0.35±0.04
	core-set	26.0±0.8	13.6±0.2	25.6±0.4	0.13±0.02	27.8±0.1	9.3±0.2	20.2±0.0	-0.42±0.02	27.2±0.1	9.5±0.1	16.9±0.1	-0.28±0.02
	IDDS	28.6±1.1	15.8±0.6	28.1±1.0	0.15±0.01	29.2±0.2	10.0±0.2	21.3±0.2	-0.45±0.02	30.0±0.1	11.0±0.1	18.0±0.3	-0.22±0.02
	Random	28.3±4.5	14.8±1.8	27.6±1.7	0.12±0.02	28.2±0.2	9.8±0.2	21.0±0.2	-0.44±0.03	28.0±0.2	10.0±0.1	17.2±0.5	-0.32±0.03
	CCAL	29.0±0.5	16.2±0.7	28.5±0.8	0.16±0.01	29.8±0.2	10.2±0.2	21.7±0.1	-0.43±0.02	30.4±0.1	11.4±0.1	18.6±0.1	-0.20±0.03
	LDCAL	**29.4±0.4**	**16.5±0.2**	**28.9±1.4**	**0.18±0.01**	**30.2±0.1**	**10.5±0.1**	**22.0±0.1**	**-0.40±0.02**	**30.8±0.1**	**11.8±0.1**	**18.9±0.1**	**-0.18±0.02**
PEGASUS Large	NSP	26.2±3.6	14.3±2.2	25.8±3.4	0.13±0.03	28.2±0.5	9.0±0.1	20.4±0.2	-0.38±0.04	26.7±0.1	9.8±0.1	17.3±0.2	-0.30±0.04
	ENSP	27.0±2.0	14.9±1.7	27.5±2.2	0.11±0.04	28.0±0.1	9.2±0.3	20.5±0.1	-0.39±0.03	28.5±0.1	10.4±0.1	17.5±0.1	-0.29±0.03
	BAS	28.5±2.0	15.1±2.8	27.5±1.6	0.15±0.02	27.4±0.2	9.7±0.2	20.9±0.3	-0.40±0.04	28.2±0.1	10.0±0.0	17.4±0.2	-0.32±0.05
	core-set	25.2±0.4	14.0±0.4	25.9±0.6	0.14±0.03	27.9±0.1	9.5±0.1	20.5±0.0	-0.41±0.03	27.8±0.1	9.8±0.1	17.0±0.0	-0.26±0.02
	IDDS	29.3±1.3	16.2±0.9	28.4±1.8	0.16±0.02	29.4±0.2	10.4±0.1	21.7±0.1	-0.36±0.02	30.4±0.1	11.3±0.1	18.3±0.2	-0.18±0.03
	Random	28.4±2.8	15.2±1.7	27.9±4.2	0.13±0.04	28.4±0.5	9.7±0.3	21.3±0.2	-0.40±0.04	28.4±0.3	10.4±0.1	17.5±0.1	-0.30±0.04
	CCAL	29.8±0.6	16.7±0.7	28.8±0.3	0.18±0.02	30.1±0.2	10.7±0.1	22.2±0.1	-0.35±0.02	30.8±0.1	11.9±0.1	18.7±0.2	-0.17±0.03
	LDCAL	**30.1±0.7**	**16.9±0.6**	**29.3±0.8**	**0.20±0.01**	**30.5±0.1**	**10.9±0.1**	**22.4±0.1**	**-0.34±0.02**	**31.0±0.1**	**12.1±0.1**	**19.2±0.1**	**-0.15±0.02**

Table 2: Main results of summarization in ROUGE and SummaC$_{ZS}$ metrics with 200 instances annotations across models and datasets, where the best results are highlighted in **bold** and the second-best are underscored.

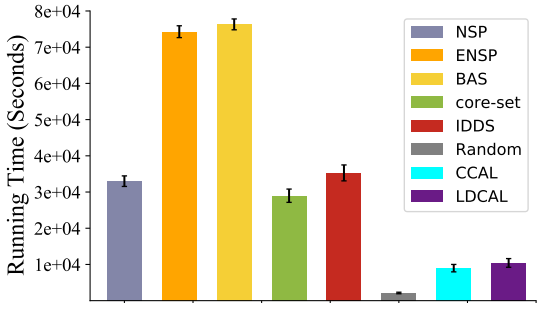

Figure 5: Running Time on the AESLC dataset.

Uncertainty-based methods NSP, ENSP, and BAS need to use models to infer the uncertainty score for each instance in each AL iteration, which is time-consuming. Core-set and IDDS strategies need to calculate the distance between each pair of instances during the diverse instance selection process, which has $\mathcal{O}(n^2)$ time complexity for a ATS dataset with the number of n documents. This is computationally expensive for larger-scale datasets. In contrast, LDCAL and CCAL approximate the linear time complexity, $\mathcal{O}(nk)$ with $k \ll n$, *e.g.*, $n = 14,400$ and $k = 200$ for the AESLC dataset.

Good Stability. Figure 6 shows the trend of the baselines' performance with increasing number of instances. Specifically, the unstable performance of NSP, ENSP, and BAS is due to their frequent selection of outliers for model training. IDDS can avoid outliers. However, few selected training data often lead to underfitting of the model, resulting in unstable performance. Zhao et al. (2021) also found the same phenomenon. In contrast, LDCAL and CCAL show good stability, which can be attributed to CL. In Figure 7, we further show the role of CL in enhancing the stability of AL performance by combining CL with other baselines.

Method	AESLC	WikiHow	PubMed
Random Sampling	26.7±1.7	21.0±0.2	17.2±0.5
LDCAL	**28.9±1.4**	**22.0±0.1**	**18.9±0.2**
w/o ACERG	26.9±1.3	21.1±0.2	17.5±0.1
w/o CL	28.3±0.7	21.5±0.1	18.3±0.1
w Classic CL	28.5±0.8	21.7±0.1	18.6±0.1

Table 3: Ablation Study for LDCAL. We trained BART with 200 annotated instances and report ROUGE-L.

4.2.2 Ablation Studies

Table 3 verifies the effectiveness of CL and AL modules in LDCAL through ablation studies. We first introduce three variants of LDCAL as follows. 1) w/o ACERG represents LDCAL without the certainty gain maximization strategy, that randomly selects 200 instances for model training from easy to hard without the AL loop. 2) w/o CL represents LDCAL without the CL process, that selects instances solely based on ACERG without considering the difficulty level. 3) w Classic CL replaces the original LLM-determined CL with Classic CL, introduced in Section 3.2.

As shown in Table 3, we derive the following three main findings. **Firstly,** when we remove the certainty gain strategy, the performance of LDCAL

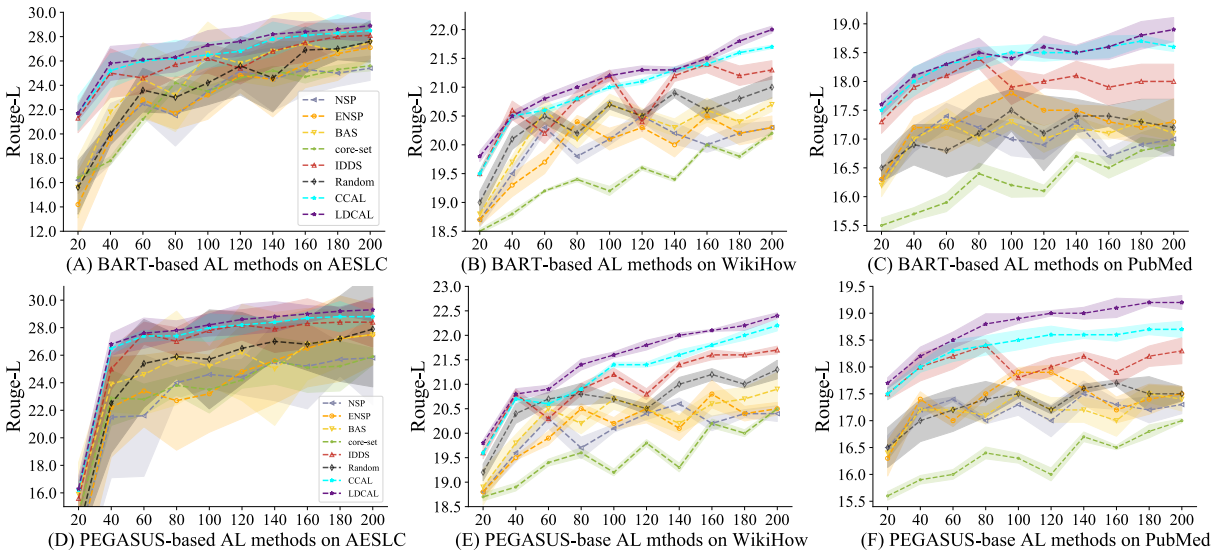

Figure 6: Number of labeled instances vs. ROUGE-L. LDCAL is compared with other AL strategies, using BART and PEGASUS as backbones, on the AESLC, WikiHow, and PubMed datasets.

drops considerably, showing the importance of the AL process. **Secondly**, w/o ACERG performs better than Random Sampling, demonstrating the effectiveness of CL even in the scenarios with only a few randomly selected instances. **Thirdly,** LDCAL with Classic or LLM-determined CL achieves superior scores compared to Random Sampling, showing the effectiveness of CL. Moreover, LLM-determined CL performs better than Classic CL, demonstrating that LLMs have their own understanding of document difficulty. We hope LDCAL can inspire more LLM-determined CL methods.

Method	AESLC	WikiHow	PubMed
NSP	25.4±1.0	20.3±0.1	17.0±0.3
NSP+CL	25.3±1.2(↓)	20.4±0.1(↑)	17.0±0.2(−)
BAS	27.5±1.5	20.6±0.2	17.1±0.1
BAS+CL	27.5±1.3(−)	20.7±0.1(↑)	17.3±0.1(↑)
core-set	25.6±0.4	20.2±0.0	16.9±0.1
core-set+CL	25.8±0.3 (↑)	20.5±0.1(↑)	17.2±0.1(↑)
IDDS	28.1±1.0	21.3±0.2	18.0±0.3
IDDS+CL	28.2±0.7(↑)	21.5±0.2(↑)	18.2±0.1(↑)
LDCAL w/o CL	28.3±0.7	21.5±0.1	18.3±0.1
LDCAL	**28.9±1.4**	**22.0±0.1**	**18.9±0.2**

Table 4: CL for other BART-based baselines. We report ROUGE-L on the AESCL dataset with 200 instances.

4.3 Qualitative Evaluation

4.3.1 Effect of CL for Other Baselines

To further explore the effectiveness of CL, we combined it with various baselines, as shown in Table 4. We observe that, when adding the LLM-determined CL, almost all baselines improve the performance significantly, demonstrating a well-defined training order of instances is effective for the AL process.

To explore the effect of CL on the stability of the AL process, we show the performance changes of the baselines as the number of instances increases in Figure 7. We find that when CL is added, the stability of the baselines is greatly enhanced.

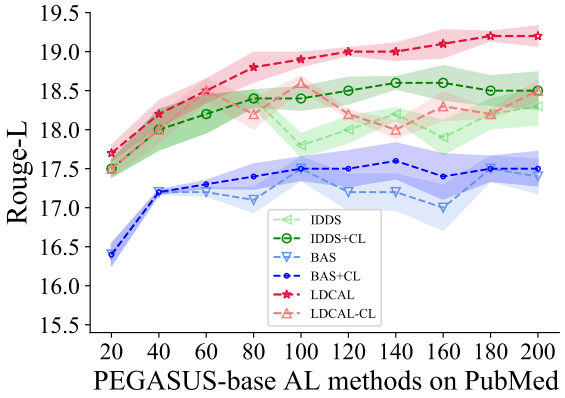

Figure 7: Combining CL with other AL strategies.

4.3.2 Effect of CL Training Order

The training order of instances is the core issue for CL. To determine the optimal training order, we conducted experiments with three types of training orders: **1) Random order**, where instances of varying difficulty levels are randomly sorted. **2) Easy-to-hard order**, where the dataset is completely sorted by the difficulty of instances. **3) Balanced order** (used in LDCAL), where the dataset is di-

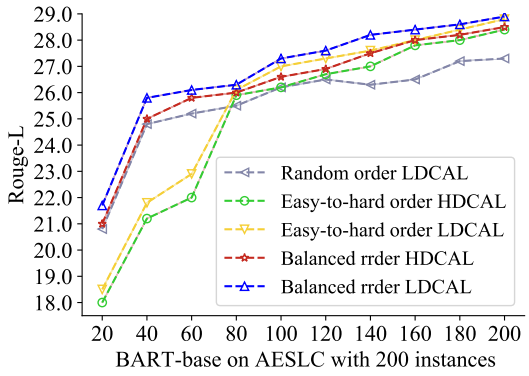

Figure 8: CL process with diverse training orders.

vided into several equal blocks, each containing equal number of easy, moderate, challenging, and hard instances. Within each block, instances are sorted from easy to hard. In Figure 8, we have three findings. **Firstly,** a simple easy-to-hard order may result in underfitting due to the initial use of simple instances. **Secondly,** although random ordering can yield good results in the early stages, it may become less effective as the number of instances increases. **Thirdly,** the balanced order more quickly achieves good performance than the simple easy-to-hard method in the early stages and continues to improve steadily during later training, because it allows the model to encounter harder instances earlier and periodically review simpler instances.

Method	AESLC	WikiHow
NSP	25.4±1.0	20.3±0.1
NSP w/o outliers	26.2±1.3(↑)	20.8±0.2(↑)
BAS	27.5±1.5	20.6±0.2
BAS w/o outliers	27.8±1.2(↑)	20.8±0.3(↑)
IDDS	28.1±1.0	21.3±0.2
IDDS + low-density ins.	28.3±1.2(↑)	21.6±0.2(↑)

Table 5: Experimental results after removing outliers for NSP and BAS, and replacing instances for IDDS.

4.3.3 Outliers and Low-Density Instances

To demonstrate the harm caused by outliers, we removed outliers from a total of 200 training instances for NSP and BAS. Table 5 shows a significant improvement in performance after removing outliers, indicating that the outliers can reduce baseline performance. To highlight the importance of low-density instances for model training, we replaced the last 20 selected instances of IDDS with the last 20 selected instances of LDCAL, *i.e.*, 20 representative instances from low-density regions. As in Table 5, IDDS + low density instances improves performance, demonstrating the importance of selecting instances in low-density regions.

5 Conclusion

This research presented the first study of the curriculum active learning framework for ATS. Extensive experiments showed that LLM-determined CL helps to improve model's stability and performance. Our AL strategy, *i.e.*, certainty gain maximization, could select diverse instances in uneven distribution scenes, further enhancing model's effectiveness and efficiency. In future work, we want to design more effective prompts for CL and investigate LDCAL in other NLG tasks beyond ATS.

Limitations

Despite the benefits, there are still several limitations of this study. **Firstly**, AL strategies directly use the annotations provided by the dataset, rather than annotating instances by humans in an human-in-the-loop manner. Thus, the effectiveness of LDCAL in real scenarios with human annotations still needs further exploration. **Secondly**, although we have achieved good results with the BART and PEGASUS backbones, our effectiveness in other recent LLMs, such as Llama (Touvron et al., 2023), still needs further validation.

Ethical Considerations

Active learning (AL) inherently involves a biased sampling process, potentially resulting in annotated datasets that reflect this bias. Consequently, one can intentionally use AL to amplify the bias within datasets. Our research enhances the effectiveness of AL, which in turn could streamline the introduction of additional bias. Additionally, we acknowledge that our approach relies on pre-trained language models, which are themselves typically biased. This inherent bias in pre-trained models can inadvertently influence the selection of instances for annotation in AL, impacting all applications that utilize these models.

Acknowledgement

First, we would like to thank all the reviewers for their valuable suggestions, which helped us improve the quality of our manuscript. Then, Dongyuan Li acknowledges the support from the China Scholarship Council (CSC).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, volume 382 of *ACM International Conference Proceeding Series*, pages 41–48. ACM.

Anshuman Chhabra, Hadi Askari, and Prasant Mohapatra. 2024. Revisiting zero-shot abstractive summarization in the era of large language models from the perspective of position bias. *Preprint*, arXiv:2401.01989.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

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.

Shen Gao, Zhengliang Shi, Minghang Zhu, Bowen Fang, Xin Xin, Pengjie Ren, Zhumin Chen, Jun Ma, and Zhaochun Ren. 2024. Confucius: Iterative tool learning from introspection feedback by easy-to-difficult curriculum. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18030–18038.

Alexios Gidiotis, Grigorios Tsoumakas, et al. 2024. Bayesian active summarization. *Comput. Speech Lang.*, 83:101553.

Tanya Goyal, Jiacheng Xu, Junyi Jessy Li, and Greg Durrett. 2022. Training dynamics for text summarization models. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2061–2073. Association for Computational Linguistics.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

Mandy Guo, Joshua Ainslie, David C. Uthus, Santiago Ontañón, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. Longt5: Efficient text-to-text transformer for long sequences. *CoRR*, abs/2112.07916.

Borna Jafarpour, Dawn Sepehr, and Nick Pogrebnyakov. 2021. Active curriculum learning. In *Proceedings of the First Workshop on Interactive Learning for Natural Language Processing*, pages 40–45, Online. Association for Computational Linguistics.

Sameer Jain, Vaishakh Keshava, Swarnashree Mysore Sathyendra, Patrick Fernandes, Pengfei Liu, Graham Neubig, and Chunting Zhou. 2023. Multi-dimensional evaluation of text summarization with in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8487–8495, Toronto, Canada. Association for Computational Linguistics.

Ryuji Kano, Takumi Takahashi, Toru Nishino, Motoki Taniguchi, Tomoki Taniguchi, and Tomoko Ohkuma. 2021. Quantifying appropriateness of summarization data for curriculum learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1395–1405, Online. Association for Computational Linguistics.

Seokhwan Kim, Yu Song, Kyungduk Kim, Jeongwon Cha, and Gary Geunbae Lee. 2006. Mmr-based active machine learning for bio named entity recognition. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 4-9, 2006, New York, New York, USA*. The Association for Computational Linguistics.

Tom Kocmi and Ondřej Bojar. 2017. Curriculum learning and minibatch bucketing in neural machine translation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 379–386.

Mahnaz Koupaee and William Yang Wang. 2018. Wikihow: A large scale text summarization dataset. *CoRR*, abs/1810.09305.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Dongyuan Li, Yusong Wang, Kotaro Funakoshi, and Manabu Okumura. 2023a. After: Active learning based fine-tuning framework for speech emotion recognition. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8.

Dongyuan Li, Yusong Wang, Kotaro Funakoshi, and Manabu Okumura. 2023b. Joyful: Joint modality fusion and graph contrastive learning for multimoda emotion recognition. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16051–16069, Singapore. Association for Computational Linguistics.

Dongyuan Li, Zhen Wang, Yankai Chen, Renhe Jiang, Weiping Ding, and Manabu Okumura. 2024a. A survey on deep active learning: Recent advances and new frontiers. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21.

Dongyuan Li, Ying Zhang, Yusong Wang, Kotaro Funakoshi, and Manabu Okumura. 2024b. Active learning with task adaptation pre-training for speech emotion recognition. *Journal of Natural Language Processing*, 31(3):825–867.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Fenglin Liu, Shen Ge, and Xian Wu. 2021. Competence-based multimodal curriculum learning for medical report generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3001–3012, Online. Association for Computational Linguistics.

Xuebo Liu, Houtim Lai, Derek F. Wong, and Lidia S. Chao. 2020. Norm-based curriculum learning for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 427–436, Online. Association for Computational Linguistics.

Zhihao Lyu, Danier Duolikun, Bowei Dai, Yuan Yao, Pasquale Minervini, Tim Z Xiao, and Yarin Gal. 2020. You need only uncertain answers: Data efficient multilingual question answering. *Workshop on Uncertainty and Ro-Bustness in Deep Learning*.

Ahmed Magooda and Diane Litman. 2021. Mitigating data scarceness through data synthesis, augmentation and curriculum for abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2043–2052, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Katerina Margatina and Nikolaos Aletras. 2023. On the limitations of simulating active learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4402–4419, Toronto, Canada. Association for Computational Linguistics.

Leland McInnes, John Healy, and James Melville. 2020. Umap: Uniform manifold approximation and projection for dimension reduction. *Preprint*, arXiv:1802.03426.

Maggie Mi. 2023. Mmi01 at the BabyLM challenge: Linguistically motivated curriculum learning for pretraining in low-resource settings. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 269–278, Singapore. Association for Computational Linguistics.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *Preprint*, arXiv:2402.06196.

Tasnim Mohiuddin, Philipp Koehn, Vishrav Chaudhary, James Cross, Shruti Bhosale, and Shafiq Joty. 2022. Data selection curriculum for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1569–1582, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Feng Nan, Cícero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Kathleen R. McKeown, Ramesh Nallapati, Dejiao Zhang, Zhiguo Wang, Andrew O. Arnold, and Bing Xiang. 2021. Improving factual consistency of abstractive summarization via question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6881–6894. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2401–2410. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Puria Radmard, Yassir Fathullah, and Aldo Lipani. 2021. Subsequence based deep active learning for named entity recognition. In *Proceedings of the 59th Annual*

Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 4310–4321, Online. Association for Computational Linguistics.

Anant Raj and Francis R. Bach. 2022. Convergence of uncertainty sampling for active learning. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 18310–18331. PMLR.

Christopher Schröder, Lydia Müller, Andreas Niekler, and Martin Potthast. 2023. Small-text: Active learning for text classification in python. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 84–95, Dubrovnik, Croatia. Association for Computational Linguistics.

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

Burr Settles. 2009. Active learning literature survey. Computer sciences technical report.

Artem Shelmanov, Dmitri Puzyrev, Lyubov Kupriyanova, Denis Belyakov, Daniil Larionov, Nikita Khromov, Olga Kozlova, Ekaterina Artemova, Dmitry V. Dylov, and Alexander Panchenko. 2021. Active learning for sequence tagging with deep pre-trained models and Bayesian uncertainty estimates. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1698–1712, Online. Association for Computational Linguistics.

Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep active learning for named entity recognition. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 252–256, Vancouver, Canada. Association for Computational Linguistics.

Sajad Sotudeh, Nazli Goharian, Hanieh Deilamsalehy, and Franck Dernoncourt. 2022. Curriculum-guided abstractive summarization for mental health online posts. In *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 148–153, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Shichao Sun, Ruifeng Yuan, Jianfei He, Ziqiang Cao, Wenjie Li, and Xiaohua Jia. 2023. Data selection curriculum for abstractive text summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7990–7995, Singapore. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Akim Tsvigun, Ivan Lysenko, Danila Sedashov, Ivan Lazichny, Eldar Damirov, Vladimir Karlov, Artemy Belousov, Leonid Sanochkin, Maxim Panov, Alexander Panchenko, Mikhail Burtsev, and Artem Shelmanov. 2023. Active learning for abstractive text summarization. *Preprint*, arXiv:2301.03252.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Shuo Wang, Yang Liu, Chao Wang, Huanbo Luan, and Maosong Sun. 2019. Improving back-translation with uncertainty-based confidence estimation. 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 791–802, Hong Kong, China. Association for Computational Linguistics.

Junda Wu, Rui Wang, Tong Yu, Ruiyi Zhang, Handong Zhao, Shuai Li, Ricardo Henao, and Ani Nenkova. 2022. Context-aware information-theoretic causal de-biasing for interactive sequence labeling. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3436–3448, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yu Xia, Xu Liu, Tong Yu, Sungchul Kim, Ryan A. Rossi, Anup Rao, Tung Mai, and Shuai Li. 2024. Hallucination diversity-aware active learning for text summarization. *Preprint*, arXiv:2404.01588.

Tim Z. Xiao, Aidan N. Gomez, and Yarin Gal. 2020. Wat zei je? detecting out-of-distribution translations with variational transformers. *CoRR*, abs/2006.08344.

Jingyi You, Dongyuan Li, Hidetaka Kamigaito, Kotaro Funakoshi, and Manabu Okumura. 2022. Joint learning-based heterogeneous graph attention network for timeline summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4091–4104, Seattle, United States. Association for Computational Linguistics.

Yue Yu, Lingkai Kong, Jieyu Zhang, Rongzhi Zhang, and Chao Zhang. 2022. AcTune: Uncertainty-based active self-training for active fine-tuning of pretrained language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1422–1436, Seattle, United States. Association for Computational Linguistics.

Michelle Yuan, Patrick Xia, Chandler May, Benjamin Van Durme, and Jordan Boyd-Graber. 2022. Adapting coreference resolution models through active learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7533–7549, Dublin, Ireland. Association for Computational Linguistics.

Hongkuan Zhang, Saku Sugawara, Akiko Aizawa, Lei Zhou, Ryohei Sasano, and Koichi Takeda. 2022a. Cross-modal similarity-based curriculum learning for image captioning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7599–7606, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Mingliang Zhang, Fandong Meng, Yunhai Tong, and Jie Zhou. 2021. Competence-based curriculum learning for multilingual machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2481–2493, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rui Zhang and Joel R. Tetreault. 2019. This email could save your life: Introducing the task of email subject line generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 446–456. Association for Computational Linguistics.

Xuan Zhang, Gaurav Kumar, Huda Khayrallah, Kenton Murray, Jeremy Gwinnup, Marianna J Martindale, Paul McNamee, Kevin Duh, and Marine Carpuat. 2018. An empirical exploration of curriculum learning for neural machine translation. *arXiv preprint arXiv:1811.00739*.

Ying Zhang, Dongyuan Li, and Manabu Okumura. 2024. Reconsidering token embeddings with the definitions for pre-trained language models. *arXiv preprint arXiv:2408.01308*.

Zhisong Zhang, Emma Strubell, and Eduard Hovy. 2022b. A survey of active learning for natural language processing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6166–6190, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Qingqing Zhu, Xiuying Chen, Pengfei Wu, JunFei Liu, and Dongyan Zhao. 2021. Combining curriculum learning and knowledge distillation for dialogue generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1284–1295, Punta Cana, Dominican Republic. Association for Computational Linguistics.