Active Learning for Abstractive Text Summarization

Akim Tsvigun[1,2], Ivan Lysenko[2], Danila Sedashov[2], Ivan Lazichny[1], Eldar Damirov[2,4], Vladimir Karlov[2], Artemy Belousov[2], Leonid Sanochkin[1,2], Maxim Panov[7], Alexander Panchenko[3], Mikhail Burtsev[1,5], Artem Shelmanov[1,6,8]

[1]AIRI, [2]HSE, [3]Skoltech, [4]SberDevices, [5]MIPT, [6]MBZUAI, [7]TII,
[8]ISP RAS Research Center for Trusted Artificial Intelligence
{tsvigun, shelmanov}@airi.net, artem.shelmanov@mbzuai.ac.ae

Abstract

Construction of human-curated annotated datasets for abstractive text summarization (ATS) is very time-consuming and expensive because creating each instance requires a human annotator to read a long document and compose a shorter summary that would preserve the key information relayed by the original document. Active Learning (AL) is a technique developed to reduce the amount of annotation required to achieve a certain level of machine learning model performance. In information extraction and text classification, AL can reduce the amount of labor up to multiple times. Despite its potential for aiding expensive annotation, as far as we know, there were no effective AL query strategies for ATS. This stems from the fact that many AL strategies rely on uncertainty estimation, while as we show in our work, uncertain instances are usually noisy, and selecting them can degrade the model performance compared to passive annotation. We address this problem by proposing the first effective query strategy for AL in ATS based on diversity principles. We show that given a certain annotation budget, using our strategy in AL annotation helps to improve the model performance in terms of ROUGE and consistency scores. Additionally, we analyze the effect of self-learning and show that it can further increase the performance of the model.

1 Introduction

Abstractive text summarization (ATS) aims to compress a document into a brief yet informative and readable summary, which would retain the key information of the original document. State-of-the-art results in this task are achieved by neural seq-to-seq models (Lewis et al., 2020; Zhang et al., 2020; Qi et al., 2020; Guo et al., 2021; Liu and Liu, 2021) based on the Transformer architecture (Vaswani et al., 2017). Training a model for ATS requires a dataset that contains pairs of original documents and their short summaries, which are usually written by human annotators. Manually composing a summary is a very tedious task, which requires one to read a long original document, select crucial information, and finally write a small text. Each of these steps is very time-consuming, resulting in the fact that constructing each instance in annotated corpora for text summarization is very expensive.

Active Learning (AL; Cohn et al. (1996)) is a well-known technique that helps to substantially reduce the amount of annotation required to achieve a certain level of machine learning model performance. For example, in tasks related to named entity recognition, researchers report annotation reduction by 2-7 times with a loss of only 1% of F1-score (Settles and Craven, 2008a). This makes AL especially important when annotation is expensive, which is the case for ATS.

AL works iteratively: on each iteration, (1) a model is trained on the so far annotated dataset; (2) the model is used to select some informative instances from a large unlabeled pool using a *query strategy*; (3) informative instances are presented to human experts, which provide gold-standard annotations; (4) finally, the instances with annotations are added to the labeled dataset, and a new iteration begins. Traditional AL query strategies are based on uncertainty estimation techniques (Lewis and Gale, 1994; Scheffer et al., 2002). The hypothesis is that the most uncertain instances for the model trained on the current iteration are informative for training the model on the next iteration. We argue that uncertain predictions of ATS models (uncertain summaries) are not more useful than randomly selected instances. Moreover, usually, they introduce more noise and detriment to the performance of summarization models. Therefore, it is not possible to straightforwardly adapt the uncertainty-based approach to AL in text summarization.

In this work, we present the first effective query strategy for AL in ATS, which we call in-domain diversity sampling (IDDS). It is based on the idea

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 5128–5152
December 7-11, 2022 ©2022 Association for Computational Linguistics

of the selection of diverse instances that are semantically dissimilar from already annotated documents but at the same time similar to the core documents of the considered domain. The empirical investigation shows that while techniques based on uncertainty cannot overcome the random sampling baseline, IDDS substantially increases the performance of summarization models. We also experiment with the self-learning technique that leverages a training dataset expanded with summaries automatically generated by an ATS model trained only on the human-annotated dataset. This approach shows improvements when one needs to generate short summaries. The code for reproducing the experiments is available online[1]. The contributions of this paper are the following:

- We propose the first effective AL query strategy for ATS that beats the random sampling baseline.
- We conduct a vast empirical investigation and show that in contrast to such tasks as text classification and information extraction, in ATS, uncertainty-based AL query strategies cannot outperform the random sampling baseline.
- To our knowledge, we are the first to investigate the effect of self-learning in conjunction with AL for ATS and demonstrate that it can substantially improve results on the datasets with short summaries.

2 Related Work

Abstractive Text Summarization. The advent of seq2seq models (Sutskever et al., 2014) along with the development of the attention mechanism (Bahdanau et al., 2015) consolidated neural networks as a primary tool for ATS. The attention-based Transformer (Vaswani et al., 2017) architecture has formed the basis of many large-scale pre-trained language models that achieve state-of-the-art results in ATS (Lewis et al., 2020; Zhang et al., 2020; Qi et al., 2020; Guo et al., 2021). Recent efforts in this area mostly focus on minor modifications of the existing architectures (Liu and Liu, 2021; Aghajanyan et al., 2021; Liu et al., 2022).

Active Learning in Natural Language Generation. While many recent works leverage AL for text classification or sequence-tagging tasks (Yuan et al., 2020; Zhang and Plank, 2021; Shelmanov et al., 2021; Margatina et al., 2021), little attention has been paid to natural language generation tasks. Among the works in this area, it is worth mentioning (Haffari et al., 2009; Ambati, 2012; Ananthakrishnan et al., 2013). These works focus on neural machine translation (NMT) and suggest several uncertainty-based query strategies for AL. Peris and Casacuberta (2018) successfully apply AL in the interactive machine translation. Liu et al. (2018) exploit reinforcement learning to train a policy-based query strategy for NMT. Although there is an attempt to apply AL in ATS (Gidiotis and Tsoumakas, 2021), to the best of our knowledge, there is no published work on this topic yet.

Uncertainty Estimation in Natural Language Generation. A simple yet effective approach for uncertainty estimation in generation is proposed by Wang et al. (2019). They use a combination of expected translation probability and variance of the translation probability, demonstrating that it can handle noisy instances better and noticeably improve the quality of back-translation. Malinin and Gales (2021) investigate the ensemble-based measures of uncertainty for NMT. Their results demonstrate the superiority of these methods for OOD detection and for identifying generated translations of low-quality. Xiao et al. (2020) propose a method for uncertainty estimation of long sequences of discrete random variables, which they dub “BLEU Variance”, and apply it for OOD sentence detection in NMT. It is also shown to be useful for identifying instances of questionable quality in ATS (Gidiotis and Tsoumakas, 2022). In this work, we investigate these uncertainty estimation techniques in AL and show that they do not provide any benefits over annotating randomly selected instances.

Diversity-based Active Learning. Along with the uncertainty-based query strategies, a series of diversity-based methods have been suggested for AL (Kim et al., 2006; Sener and Savarese, 2018; Ash et al., 2019; Citovsky et al., 2021). The most relevant work among them is (Kim et al., 2006), where the authors propose to use a Maximal Marginal Relevance (MMR; Carbonell and Goldstein (1998))-based function as a query strategy in AL for named entity recognition. This function aims to capture uncertainty and diversity and selects instances for annotation based on these two perspectives. We adapt this strategy for the ATS task and compare the proposed method with it.

[1] https://github.com/AIRI-Institute/al_ats

3 Uncertainty-based Active Learning for Text Generation

In this section, we give a brief formal definition of the AL procedure for text generation and uncertainty-based query strategies. Here and throughout the rest of the paper, we denote an input sequence as $\mathbf{x} = (x_1 \dots x_m)$ and the output sequence as $\mathbf{y} = (y_1 \dots y_n)$, with m and n being lengths of $\mathbf{x}$ and $\mathbf{y}$ respectively.

Let $\mathcal{D} = \{(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})\}_{k=1}^{K}$ be a dataset of pairs (documents, summaries). Consider a probabilistic model $p_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$ parametrized by a vector $\mathbf{w}$. Usually, $p_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$ is a neural network, while the parameter estimation is done via the maximum likelihood approach:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} L(\mathcal{D}, \mathbf{w}), \qquad (1)$$

where $L(\mathcal{D}, \mathbf{w}) = \sum_{k=1}^{K} \log p_{\mathbf{w}}(\mathbf{y}^{(k)} \mid \mathbf{x}^{(k)})$ is log-likelihood.

Many AL methods are based on greedy query strategies that select instances for annotation, optimizing a certain criterion $\mathcal{A}(\mathbf{x} \mid \mathcal{D}, \hat{\mathbf{w}})$ called an *acquisition function*:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \mathcal{A}(\mathbf{x} \mid \mathcal{D}, \hat{\mathbf{w}}). \qquad (2)$$

The selected instance $\mathbf{x}^*$ is then annotated with a target value $\mathbf{y}^*$ (document summary) and added to the training dataset: $\mathcal{D} := \mathcal{D} \cup (\mathbf{x}^*, \mathbf{y}^*)$. Subsequently, the model parameters $\mathbf{w}$ are updated and the instance selection process continues until the desired model quality is achieved or the available annotation budget is depleted.

The right choice of an acquisition function is crucial for AL. A common heuristic for acquisition is selecting instances with high uncertainty. Below, we consider several measures of uncertainty used in text generation.

Normalized Sequence Probability (NSP) was originally proposed by Ueffing and Ney (2007) and has been used in many subsequent works (Haffari et al., 2009; Wang et al., 2019; Xiao et al., 2020; Lyu et al., 2020). This measure is given by

$$\mathrm{NSP}(\mathbf{x}) = 1 - \bar{p}_{\hat{\mathbf{w}}}(\mathbf{y} \mid \mathbf{x}), \qquad (3)$$

where we define the geometric mean of probabilities of tokens predicted by the model as: $\bar{p}_{\hat{\mathbf{w}}}(\mathbf{y} \mid \mathbf{x}) = \exp\bigl\{\frac{1}{n} \log p_{\hat{\mathbf{w}}}(\mathbf{y} \mid \mathbf{x})\bigr\}$.

A wide family of uncertainty measures can be derived using the Bayesian approach to modeling. Under the Bayesian approach, it is assumed that model parameters have a prior distribution $\pi(\mathbf{w})$. Optimization of the log-likelihood $L(\mathcal{D}, \mathbf{w})$ in this case leads to the optimization of the posterior distribution of the model parameters:

$$\pi(\mathbf{w} \mid \mathcal{D}) \propto \exp\{L(\mathcal{D}, \mathbf{w})\} \cdot \pi(\mathbf{w}). \qquad (4)$$

Usually, the exact computation of the posterior is intractable, and to perform training and inference, a family of distributions $q_\theta(\mathbf{w})$ parameterized by θ is introduced. The parameter estimate $\hat{\theta}$ minimizes the KL-divergence between the true posterior $\pi(\mathbf{w} \mid \mathcal{D})$ and the approximation $q_{\hat{\theta}}(\mathbf{w})$. Given such an approximation, several uncertainty measures can be constructed.

Expected Normalized Sequence Probability (ENSP) is proposed by Wang et al. (2019) and is also used in (Xiao et al., 2020; Lyu et al., 2020):

$$\mathrm{ENSP}(\mathbf{x}) = 1 - \mathbb{E}_{\mathbf{w} \sim q_{\hat{\theta}}} \bar{p}_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}). \qquad (5)$$

In practice, the expectation is approximated via Monte Carlo dropout (Gal and Ghahramani, 2016), i.e. averaging multiple predictions obtained with activated dropout layers in the network.

Expected Normalized Sequence Variance (ENSV; Wang et al. (2019)) measures the variance of the sequence probabilities obtained via Monte Carlo dropout:

$$\mathrm{ENSV}(\mathbf{x}) = \mathrm{Var}_{\mathbf{w} \sim q_{\hat{\theta}}} \bar{p}_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}). \qquad (6)$$

BLEU Variance (BLEUVar) is proposed by Xiao et al. (2020). It treats documents as points in some high dimensional space and uses the BLEU metric (Papineni et al., 2002) for measuring the difference between them. In such a setting, it is possible to calculate the variance of generated texts in the following way:

$$\mathrm{BLEUVar}(\mathbf{x}) = \qquad (7)$$
$$= \mathbb{E}_{\mathbf{w} \sim q_{\hat{\theta}}} \mathbb{E}_{\mathbf{y}, \mathbf{y}' \sim p_{\mathbf{w}}(\cdot \mid \mathbf{x})} \bigl(1 - \mathrm{BLEU}(\mathbf{y}, \mathbf{y}')\bigr)^2.$$

The BLEU metric is calculated as a geometric mean of n-grams overlap up to 4-grams. Consequently, when summaries consist of less than 4 tokens, the metric is equal to zero since there will be no common 4-grams. This problem can be mitigated with the SacreBLEU metric (Post, 2018), which smoothes the n-grams with zero counts. When we use this query strategy with the SacreBLUE metric, we refer to it as **SacreBLEUVar**.

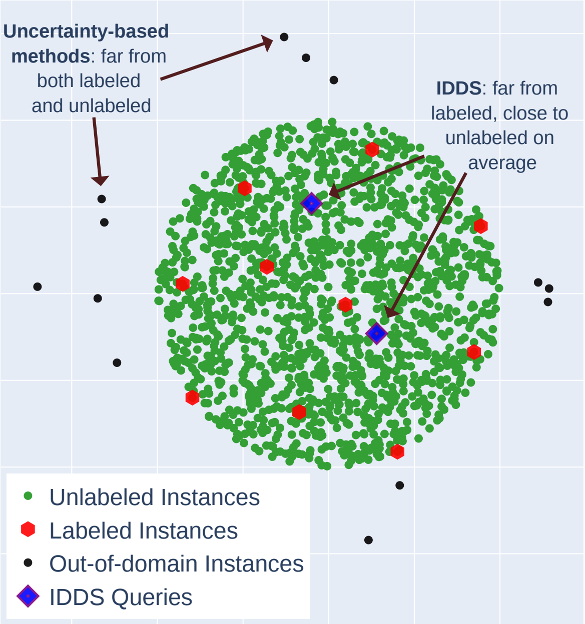


Figure 1: The visualization of the idea behind the IDDS alogrithm on the synthetic data: select instances located far from labeled data while close on average to unlabeled data.

4 Proposed Methods

4.1 In-Domain Diversity Sampling

We argue that uncertainty-based query strategies tend to select noisy instances that have little value for training ATS models. To alleviate this issue, we propose a novel query strategy named in-domain diversity sampling (IDDS). It aims to maximize the diversity of the annotated instances by selecting instances that are dissimilar from the already annotated ones. At the same time, it avoids selecting noisy outliers. These noisy documents that are harmful to training an ATS model are usually semantically dissimilar from the core documents of the domain represented by the unlabeled pool. Therefore, IDDS queries instances that are dissimilar to the annotated instances but at the same time are similar to unannotated ones (Figure 1).

We propose the following acquisition function that implements the aforementioned idea (the higher the value – the higher the priority for the annotation):

$$\mathrm{IDDS}(\mathbf{x}) = \lambda \frac{\sum\limits_{j=1}^{|U|} s(\mathbf{x}, \mathbf{x}_j)}{|U|} - (1-\lambda) \frac{\sum\limits_{i=1}^{|L|} s(\mathbf{x}, \mathbf{x}_i)}{|L|}, \tag{8}$$

where $s(\mathbf{x}, \mathbf{x}')$ is a similarity function between texts, U is the unlabeled set, L is the labeled set, and $\lambda \in [0; 1]$ is a hyperparameter.

Below, we formalize the resulting algorithm of the IDDS query strategy.

1. For each document in the unlabeled pool $\mathbf{x}$, we obtain an embedding vector $\mathbf{e}(\mathbf{x})$. For this purpose, we suggest using the [CLS] pooled sequence embeddings from BERT. We note that using a pre-trained checkpoint straightforwardly may lead to unreasonably high similarity scores between instances since they all belong to the same domain, which can be quite specific. We mitigate this problem by using the task-adaptive pre-training (TAPT; Gururangan et al. (2020)) on the unlabeled pool. TAPT performs several epochs of self-supervised training of the pre-trained model on the target dataset to acquaint it with the peculiarities of the data.
2. Deduplicate the unlabeled pool. Instances with duplicates will have an overrated similarity score with the unlabeled pool.
3. Calculate the informativeness scores using the IDDS acquisition function (8). As a similarity function, we suggest using a scalar product between document representations: $s(\mathbf{x}, \mathbf{x}') = \langle \mathbf{e}(\mathbf{x}), \mathbf{e}(\mathbf{x}') \rangle$.

The idea of IDDS is close to the MMR-based strategy proposed in (Kim et al., 2006). Yet, despite the resemblance, IDDS differs from it in several crucial aspects. The MMR-based strategy focuses on the uncertainty and diversity components. However, as shown in Section 6.1, selecting instances by uncertainty leads to worse results compared to random sampling. Consequently, instead of using uncertainty, IDDS leverages the unlabeled pool to capture the *representativeness* of the instances. Furthermore, IDDS differs from the MMR-based strategy in how they calculate the diversity component. MMR directly specifies the usage of the "max" aggregation function for calculating the similarity with the already annotated data, while IDDS uses "average" similarity instead and achieves better results as shown in Section 6.2.

We note that IDDS does not require retraining an acquisition model in contrast to uncertainty-based strategies since document vector representations and document similarities can be calculated before starting the AL annotation process. This results in the fact that no heavy computations during AL are required. Consequently, IDDS does not harm the interactiveness of the annotation process, which is a common bottleneck (Tsvigun et al., 2022).

4.2 Self-learning

Pool-based AL assumes that there is a large unlabeled pool of data. We propose to use this data source during AL to improve text summarization models with the help of self-learning. We train the model on the labeled data and generate summaries for the whole unlabeled pool. Then, we concatenate the generated summaries with the labeled set and use this data to fine-tune the final model. We note that generated summaries can be noisy: irrelevant, grammatically incorrect, contain factual inconsistency, and can harm the model performance. We detect such instances using the uncertainty estimates obtained via NSP scores and exclude k_l% instances with the lowest scores and k_h% of instances with the highest scores. We choose this uncertainty metric because according to our experiments in Section 6.1, high NSP scores correspond to the noisiest instances. We note that adding the filtration step does not introduce additional computational overhead, since the NSP scores are calculated simultaneously with the summary generation for self-learning.

5 Experimental Setup

5.1 Active Learning Setting

We evaluate IDDS and other query strategies using the conventional scheme of AL annotation emulation applied in many previous works (Settles and Craven, 2008b; Shen et al., 2017; Siddhant and Lipton, 2018; Shelmanov et al., 2021; Dor et al., 2020). For uncertainty-based query strategies and random sampling, we start from a small annotated seeding set selected randomly. This set is used for fine-tuning the summarization model on the first iteration. For IDDS, the seeding set is not used, since this query strategy does not require fine-tuning the model to make a query. On each AL iteration, we select top-k instances from the unlabeled pool according to the informativeness score obtained with a query strategy. The selected instances with their gold-standard summaries are added to the so-far annotated set and are excluded from the unlabeled pool. On each iteration, we fine-tune a summarization model from scratch and evaluate it on a held-out test set. We report the performance of the model on each iteration to demonstrate the dynamics of the model performance depending on the invested annotation effort.

The query size (the number of instances selected for annotation on each iteration) is set to 10 documents. We repeat each experiment 9 times with different random seeds and report the mean and the standard deviation of the obtained scores. For the WikiHow and PubMed datasets, on each iteration, we use a random subset from the unlabeled pool since generating predictions for the whole unlabeled dataset is too computationally expensive. In the experiments, the subset size is set to 10,000 for WikiHow and 1,000 for PubMed.

5.2 Baselines

We use random sampling as the main baseline. To our knowledge, in the ATS task, this baseline has not been outperformed by any other query strategy yet. In this baseline, an annotator is given randomly selected instances from the unlabeled pool, which means that AL is not used at all. We also report results of uncertainty-based query strategies and an MMR-based query strategy (Kim et al., 2006) that is shown to be useful for named entity recognition.

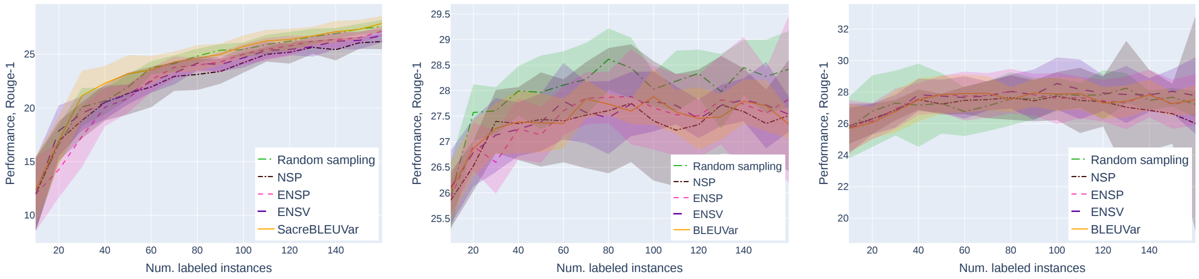
5.3 Metrics

Quality of Text Summarization. To measure the quality of the text summarization model, we use the commonly adopted ROUGE metric (Lin, 2004). Following previous works (See et al., 2017; Nallapati et al., 2017; Chen and Bansal, 2018; Lewis et al., 2020; Zhang et al., 2020), we report ROUGE-1, ROUGE-2, and ROUGE-L. Since we found the dynamics of these metrics coinciding, for brevity, in the main part of the paper, we keep only the results with the ROUGE-1 metric. The results with other metrics are presented in the appendix.

Factual Consistency. Inconsistency (hallucination) of the generated summaries is one of the most crucial problems in summarization (Kryscinski et al., 2020; Huang et al., 2021; Nan et al., 2021; Goyal et al., 2022). Therefore, in addition to the ROUGE metrics, we measure the factual consistency of the generated summaries with the original documents. We use the SummaC-ZS (Laban et al., 2022) – a state-of-the-art model for inconsistency detection. We set *granularity = "sentence"* and *model_name = "vitc"*.

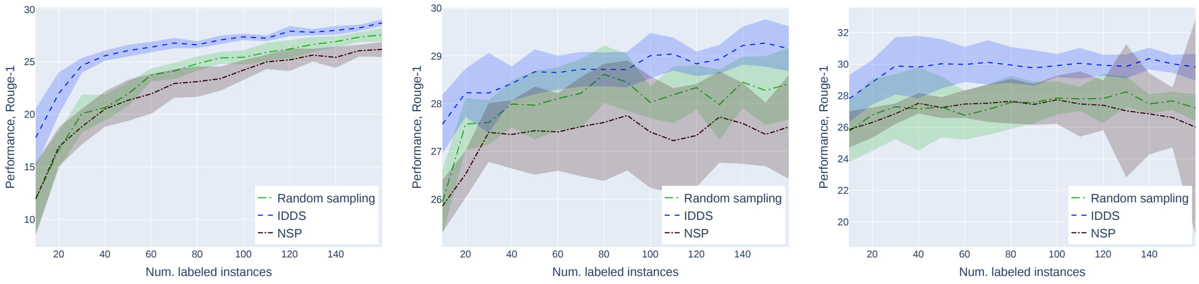
5.4 Datasets

We experiment with three datasets widely-used for evaluation of ATS models: AESLC (Zhang and Tetreault, 2019), PubMed (Cohan et al., 2018), and WikiHow (Koupaee and Wang, 2018). AESLC consists of emails with their subject lines as summaries. WikiHow contains articles from WikiHow pages



a) AESLC dataset b) WikiHow dataset c) PubMed dataset

Figure 2: ROUGE-1 scores of BART-base with various uncertainty-based strategies compared with random sampling (baseline) on various datasets. Full results are provided in Figures 6, 8, 9, respectively.



a) AESLC dataset b) WikiHow dataset c) PubMed dataset

Figure 3: ROUGE-1 scores of BART-base with the IDDS strategy compared with random sampling (baseline) and NSP (uncertainty-based strategy) on various datasets. Full results are provided in Figures 10, 12 and 13, respectively.

with their headlines as summaries. PubMed (Cohan et al., 2018) is a collection of scientific articles from the PubMed archive with their abstracts. The choice of datasets is stipulated by the fact that AESLC contains short documents and summaries, WikiHow contains medium-sized documents and summaries, and PubMed contains long documents and summaries. We also use two non-intersecting subsets of the Gigaword dataset (Graff et al., 2003; Rush et al., 2015) of sizes 2,000 and 10,000 for hyperparameter optimization of ATS models and additional experiments with self-learning, respectively. Gigaword consists of news articles and their headlines representing summaries. The dataset statistics is presented in Table 2 in Appendix A.

5.5 Models and Hyperparameters

We conduct experiments using the state-of-the-art text summarization models: BART (Lewis et al., 2020) and PEGASUS (Zhang et al., 2020). In all experiments, we use the "base" pre-trained version of BART and the "large" version of PEGASUS. Most of the experiments are conducted with the BART model, while PEGASUS is only used for the AESLC dataset (results are presented in Appendices B, C) since running it on two other datasets in AL introduces a computational bottleneck.

We tune hyperparameter values of ATS models using the ROUGE-L score on the subset of the Gigaword dataset. The hyperparameter values are provided in Table 3 in Appendix A.

For the IDDS query strategy, we use $\lambda = 0.67$. We analyze the effect of different values of this parameter in Section 6.2.

6 Results and Discussion

6.1 Uncertainty-based Query Strategies

In this series of experiments, we demonstrate that selected uncertainty-based query strategies are not suitable for AL in ATS. Figure 2a and Figures 6, 7 in Appendix B present the results on the AESLC dataset. As we can see, none of the uncertainty-based query strategies outperform the random sampling baseline for both BART and PEGASUS models. NSP and ENSP strategies demonstrate the worst results with the former having the lowest performance for both ATS models. Similar results are obtained for the WikiHow and PubMed datasets (Figures 2b and 2c).

In some previous work on NMT, uncertainty-based query strategies outperform the random sampling baseline (Haffari et al., 2009; Ambati, 2012; Ananthakrishnan et al., 2013). Their low results for ATS compared to NMT might stem from the differences between these tasks. Both NMT and ATS are seq2seq tasks and can be solved via similar models. However, NMT is somewhat easier, since the

AL Strat.	Document	Golden Summary	Gener. Summ.
NSP	Aquarius - Horoscope Friday, September 8, 2000 by Astronet.com. Powerful forces are at work to challenge you (...) Don't let hurt feelings prevent you from (...)	**These things are beginning to scare me...**	Invitation – Aquarius
NSP	Prod Area and Long Haul k# **Volume** Rec Del 3.6746 5000 St 62 (...) #6563 PPL (Non NY) should have this contract tomorrow. (...) 3.5318 6500 Leidy PSE&G	**TRCO capacity for Sep**	Prod Area
IDDS	Greg, I wanted to forward this letter to you that I received from a **good friend of mine** who is interested in discussing (...) **with Enron**. (...) set up a **meeting** (...) Sincerely,	**Meeting with Enron Networks**	n/a
IDDS	Larry, Could I have the **price** for a 2 day swing **peaker** option at **NGI Chicago**, that can be exercised on any day in **February** 2002. Strike is FOM **February**, (...)	**Peaker price for NGI Chicago Feb**	n/a

Table 1: Examples of instances selected with the NSP and IDDS strategies. Tokens overlapping with the source document are highlighted with **green**. Tokens that refer to paraphrasing a part of the document and the corresponding part are highlighted with **blue**. Tokens that cannot be derived from the document are highlighted with **red**.

output is usually of similar length as the input and its variability is smaller. It is much easier to train a model to reproduce an exact translation rather than make it generate an exact summary. Therefore, uncertainty estimates of ATS models are way less reliable than estimates of translation models. These estimates often select for annotation noisy documents that are useless or even harmful for training summarization models. Table 1 reveals several documents selected by the worst-performing strategy NSP on AESLC. We can see that NSP selects domain-irrelevant documents or very specific ones. Their summaries can hardly be restored from the source documents, which means that they most likely have little positive impact on the generalization ability of the model. More examples of instances selected by different query strategies are presented in Table 5 in Appendix E.

6.2 In-Domain Diversity Sampling

In this series of experiments, we analyze the proposed IDDS query strategy. Figure 3a and Figures 10, 11 in Appendix C show the performance of the models with various query strategies on AESLC. We can see that the proposed strategy outperforms random sampling on all iterations for both ATS models and subsequently outperforms the uncertainty-based strategy NSP. IDDS demonstrates similar results on the WikiHow and PubMed datasets, outperforming the baseline with a large margin as depicted in Figures 3b and 3c. We also report the improvement of IDDS over random sampling in percentage on several AL iterations in Table 4. We can see that IDDS provides an especially large improvement in the cold-start AL scenario when the amount of labeled data is very small.

We carry out several ablation studies for the proposed query strategy. First, we investigate the effect of various models for document embeddings construction and the necessity of performing TAPT. Figures 17 and 18 in Appendix F illustrate that TAPT substantially enhances the performance of IDDS. Figure 17 also shows that the BERT-base encoder appears to be better than SentenceBERT (Reimers and Gurevych, 2019) and LongFormer (Beltagy et al., 2020).

Second, we try various functions for calculating the similarity between instances. Figures 19, 20 in Appendix F compare the originally used dot product with Mahalanobis and Euclidean distances on AESLC and WikiHow. On AESLC, IDDS with Mahalanobis distance persistently demonstrates lower performance. IDDS with the Euclidean distance shows a performance drop on the initial AL iterations compared to the dot product. On WikiHow, however, all the variants perform roughly the same. Therefore, we suggest keeping the dot product for computing the document similarity in IDDS since it provides the most robust results across the datasets.

We also compare the dot product with its normalized version – cosine similarity on AESLC and PubMed, see Figures 21 and 22 in Appendix F. On both datasets, adding normalization leads to substantially worse results on the initial AL iterations. We deem that this happens because normalization may damage the representativeness component since the norm of the embedding can be treated as a measure of the representativeness of the corresponding document.

Third, we investigate how different values for the lambda coefficient influence the performance of IDDS. Table 7 and Figure 23 in Appendix F shows that smaller values of $\lambda \in \{0, 0.33, 0.5\}$ substantially deteriorate the performance. Smaller values correspond to selecting instances that are highly dissimilar from the documents in the unlabeled pool, which leads to picking many outliers. Higher values lead to the selection of instances from the core of the unlabeled dataset, but also very similar to the annotated part. This also results in a lower quality on the initial AL iterations. The best and most stable results are obtained with $\lambda = 0.67$.

Fourth, we compare IDDS with the MMR-based strategy suggested in (Kim et al., 2006). Since it uses uncertainty, it requires a trained model to calculate the scores. Consequently, the initial query is taken randomly as no trained model is available on the initial AL iteration. Therefore, we use the modification, when the initial query is done with IDDS because it provides substantially better results on the initial iteration. We also experiment with different values of the λ hyperparameter of the MMR-based strategy. Figure 24 illustrates a large gap in performance of IDDS and the MMR-based strategy regardless of the initialization / λ on AESLC. We believe that this is attributed to the fact that strategies incorporating uncertainty are harmful to AL in ATS as shown in Section 6.1.

Finally, we compare "aggregation" functions for estimating the similarity between a document and a collection of documents (labeled and unlabeled pools). Following the MMR-based strategy (Kim et al., 2006), instead of calculating the *average* similarity between the embedding of the target document and the embeddings of documents from the labeled set, we calculate the *maximum* similarity. We also try replacing the "average" aggregation function with "maximum" in both IDDS components in (8). Figures 25 and 26 in Appendix F show that *average* leads to better performance on both AESLC and WikiHow datasets.

The importance of diversity sampling is illustrated in Table 6 in Appendix E. We can see that NSP-based query batches contain a large number of overlapping instances. This may partly stipulate the poor performance of the NSP strategy since almost 9% of labeled instances are redundant. IDDS, on the contrary, does not have instances with overlapping summaries inside batches at all.

6.3 Self-learning

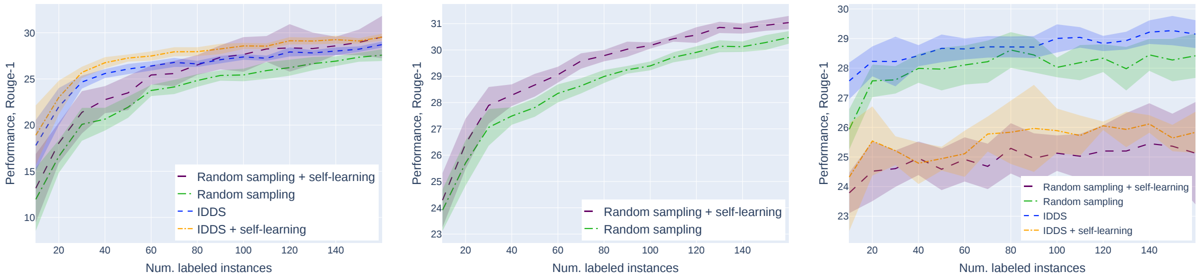
In this section, we investigate the effect of self-learning in the AL setting. Figures 4a, 4b illustrate the effect of self-learning on the AESLC and Gigaword datasets. For this experiment, we use $k_l = 10, k_h = 1$, filtering out 11% of automatically generated summaries. In both cases: with AL and without, adding automatically generated summaries of documents from the unlabeled pool to the training set improves the performance of the summarization model. On AESLC, the best results are obtained with both AL and self-learning: their combination achieves up to 58% improvement in all ROUGE metrics compared to using passive annotation without self-learning.

The same experiment on the WikiHow dataset is presented in Figure 4c. To make sure that the quality is not deteriorated due to the addition of noisy uncertain instances, we use $k_l = 38, k_h = 2$ for this experiment, filtering out 40% of automatically generated summaries. On this dataset, self-learning reduces the performance for both cases (with AL and without). We deem that the benefit of self-learning depends on the length of the summaries in the dataset. AESLC and Gigaword contain very short summaries (less than 13 tokens on average, see Table 2). Since the model is capable of generating short texts that are grammatically correct and logically consistent, such data augmentation does not introduce much noise into the dataset, resulting in performance improvement. WikiHow, on the contrary, contains *long* summaries (77 tokens on average). Generation of long, logically consistent, and grammatically correct summaries is still a challenging task even for the state-of-the-art ATS models. Therefore, the generated summaries are of low quality, and using them as an additional training signal deteriorates the model performance. Consequently, we suggest using self-learning only if the dataset consists of relatively short texts. We leave a more detailed investigation of this topic for future research.

6.4 Consistency

We analyze how various AL strategies and self-learning affect the consistency of model output in two ways. We measure the consistency of the generated summaries with the original documents on the test set on each AL iteration. Figure 5 shows that the model trained on instances queried by IDDS generates the most consistent summaries across all considered AL query strategies on AESLC. On the contrary, the model trained on the instances selected by the uncertainty-based NSP query strategy generates summaries with the lowest consistency.

Figure 28 in Appendix G demonstrates that on AESLC, self-learning also improves consistency



a) AESLC dataset
$k_l = 10, k_h = 1.$

b) Gigaword dataset
$k_l = 10, k_h = 1$

c) WikiHow dataset
$k_l = 38, k_h = 2$

Figure 4: ROUGE-1 scores of the BART-base model with IDDS and random sampling strategies with and without self-learning on AESLC, Gigaword, and WikiHow. Full results are provided in Figures 14, 15, and 16, respectively.

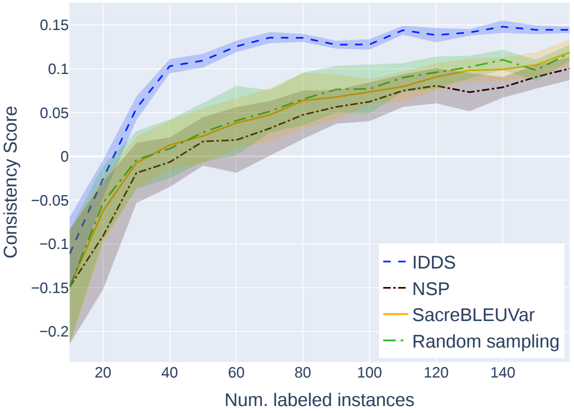


Figure 5: The consistency score calculated via SummaC with BART-base on AESLC with various AL strategies.

regardless of the AL strategy. The same trend is observed on Gigaword (Figure 27 in Appendix G).

However, for WikiHow, there is no clear trend. Figure 29 in Appendix G shows that all query strategies lead to similar consistency results, with NSP producing slightly higher consistency, and BLEUVar – slightly lower. We deem that this may be due to the fact that summaries generated by the model on WikiHow are of lower quality than the golden summaries regardless of the strategy. Therefore, this leads to biased scores of the SummaC model with similar results on average.

6.5 Query Duration

We compare the average duration of AL iterations for various query strategies. Figure 30 in the Appendix H presents the average training time and the average duration of making a query. We can see that training a model takes considerably less time than selecting the instances from the unlabeled pool for annotation. Therefore, the duration of AL iterations is mostly determined by the efficiency of the query strategy. The IDDS query strategy does not require any heavy computations during AL, which makes it also the best option for keeping the AL process interactive.

7 Conclusion

In this work, we convey the first study of AL in ATS and propose the first active learning query strategy that outperforms the baseline random sampling. The query strategy aims at selecting for annotation the instances with high similarity with the documents in the unlabeled pool and low similarity with the already annotated documents. It outperforms the random sampling in terms of ROUGE metrics on all considered datasets. It also outperforms random sampling in terms of the consistency score calculated via the SummaC model on the AESLC dataset. We also demonstrate that uncertainty-based query strategies fail to outperform random sampling, resulting in the same or even worse performance. Finally, we show that self-learning can improve the performance of an ATS model in terms of both the ROUGE metrics and consistency. This is especially favorable in AL since there is always a large unlabeled pool of data. We show that combining AL and self-learning can give an improvement of up to 58% in terms of ROUGE metrics.

In future work, we look forward to investigating IDDS in other sequence generation tasks. This query strategy might be beneficial for tasks with the highly variable output when uncertainty estimates of model predictions are unreliable and cannot outperform the random sampling baseline. IDDS facilitates the representativeness of instances in the training dataset without leveraging uncertainty scores.

Limitations

Despite the benefits, the proposed methods require some conditions to be met to be successfully applied in practice. IDDS strategy requires making TAPT of the embeddings-generated model, which may be computationally consuming for a large dataset. Self-learning, in turn, may harm the performance when the summaries are too long, as shown in Section 6.3. Consequently, its application requires a detailed analysis of the properties of the target domain summaries.

Ethical Considerations

It is important to note that active learning is a method of biased sampling, which can lead to biased annotated corpora. Therefore, active learning can be used to deliberately increase the bias in the datasets. Our research improves the active learning performance; hence, our contribution would also make it more efficient for introducing more bias as well. We also note that our method uses the pre-trained language models, which usually contain different types of biases by themselves. Since bias affects all applications of pre-trained models, this can also unintentionally facilitate the biased selection of instances for annotation during active learning.

Acknowledgements

We thank anonymous reviewers for their insightful suggestions to improve this paper. The work was supported by a grant for research centers in the field of artificial intelligence (agreement identifier 000000D730321P5Q0002 dated November 2, 2021 No. 70-2021-00142 with ISP RAS). This research was supported in part by computational resources of the HPC facilities at the HSE University (Kostenetskiy et al., 2021).

References

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. 2021. Better fine-tuning by reducing representational collapse. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Vamshi Ambati. 2012. *Active Learning and Crowdsourcing for Machine Translation in Low Resource Scenarios*. Ph.D. thesis, Language Technologies Institute School of Computer Science Carnegie Mellon University, USA. AAI3528171.

Sankaranarayanan Ananthakrishnan, Rohit Prasad, David Stallard, and Prem Natarajan. 2013. Batch-mode semi-supervised active learning for statistical machine translation. *Computer Speech & Language*, 27(2):397–406.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2019. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*.

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.

Jaime G. Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 24-28 1998, Melbourne, Australia*, pages 335–336. ACM.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 675–686. Association for Computational Linguistics.

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

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.

David A Cohn, Zoubin Ghahramani, and Michael I Jordan. 1996. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145.

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.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org.

Alexios Gidiotis and Grigorios Tsoumakas. 2021. Bayesian active summarization. *CoRR*, abs/2110.04480.

Alexios Gidiotis and Grigorios Tsoumakas. 2022. Should we trust this summary? bayesian abstractive summarization to the rescue. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4119–4131. Association for Computational Linguistics.

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.

Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8342–8360. Association for Computational Linguistics.

Gholamreza Haffari, Maxim Roy, and Anoop Sarkar. 2009. Active learning for statistical phrase-based machine translation. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA*, pages 415–423. The Association for Computational Linguistics.

Yichong Huang, Xiachong Feng, Xiaocheng Feng, and Bing Qin. 2021. The factual inconsistency problem in abstractive text summarization: A survey.

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.

P. S. Kostenetskiy, R. A. Chulkevich, and V. I. Kozyrev. 2021. HPC Resources of the Higher School of Economics. *Journal of Physics: Conference Series*, 1740(1):012050.

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

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9332–9346. Association for Computational Linguistics.

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.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 3–12. ACM/Springer.

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.

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.

Ming Liu, Wray L. Buntine, and Gholamreza Haffari. 2018. Learning to actively learn neural machine translation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 334–344. Association for Computational Linguistics.

Yixin Liu and Pengfei Liu. 2021. Simcls: A simple framework for contrastive learning of abstractive summarization. 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 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 1065–1072. Association for Computational Linguistics.

Yixin Liu, Pengfei Liu, Dragomir R. Radev, and Graham Neubig. 2022. BRIO: bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2890–2903. 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*.

Andrey Malinin and Mark J. F. Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3075–3081. AAAI Press.

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.

Álvaro Peris and Francisco Casacuberta. 2018. Active learning for interactive neural machine translation of data streams. In *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 151–160. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. 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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Tobias Scheffer, Stefan Wrobel, Borislav Popov, Damyan Ognianov, Christian Decomain, and Susanne Hoche. 2002. Lerning hidden markov models for information extraction actively from partially labeled text. *Künstliche Intell.*, 16(2):17–22.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics.

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

Burr Settles and Mark Craven. 2008a. An analysis of active learning strategies for sequence labeling tasks. In *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1070–1079. Association for Natural Language Processing.

Burr Settles and Mark Craven. 2008b. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1070–1079, Honolulu, Hawaii. Association for Computational Linguistics.

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.

Aditya Siddhant and Zachary C. Lipton. 2018. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2904–2909. 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.

Akim Tsvigun, Artem Shelmanov, Gleb Kuzmin, Leonid Sanochkin, Daniil Larionov, Gleb Gusev, Manvel Avetisian, and Leonid Zhukov. 2022. Towards computationally feasible deep active learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1198–1218, Seattle, United States. Association for Computational Linguistics.

Nicola Ueffing and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Comput. Linguistics*, 33(1):9–40.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

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.

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.

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.

Mike Zhang and Barbara Plank. 2021. Cartography active learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 395–406. 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.

A Dataset Statistics and Model Hyperparameters

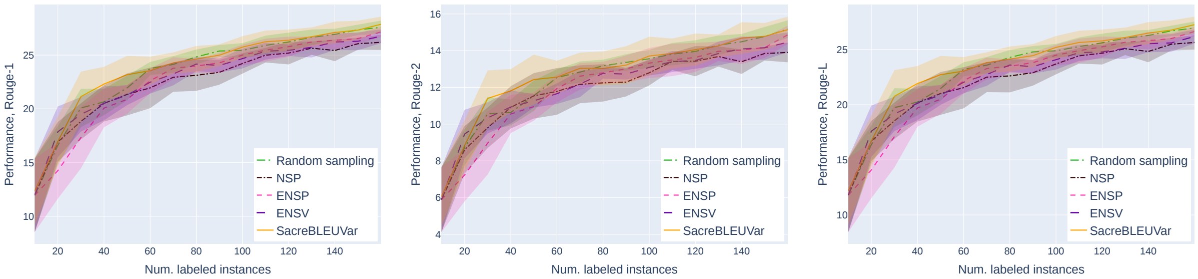
Table 2: Dataset statistics. We provide a number of instances for the training and test sets and average lengths of documents / summaries in terms of tokens. All the datasets are English-language. We filter the WikiHow dataset since it contains many noisy instances: we exclude instances with documents that have 10 or less tokens and instances with summaries that have 3 or less tokens.

Dataset	Subset	Num. instances	Av. document len.	Av. summary len.
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
Gigaword (self-learning)	Train	10K	38.9	11.9
	Test	2K	37.1	12.8
Gigaword (hyperparam. optimiz.)	Train	200	40.8	13.3
	Test	2K	38.6	12.5

Table 3: Hyperparameter values and checkpoints from the HuggingFace repository (Wolf et al., 2019) of the models. We imitate the low-resource case by randomly selecting 200 instances from Gigaword train dataset as a train sample, and 2,000 instances from the validation set as a test sample for evaluation consistency. For each model, we find the optimal hyperparameters according to evaluation scores on the sampled subset. Generation maximum length is set to the maximum summary length from the available labeled set.
For WikiHow and PubMed datasets, we reduce the batch size and increase gradient accumulation steps by the same amount due to computational bottleneck.
Hardware configuration: 2 Intel Xeon Platinum 8168, 2.7 GHz, 24 cores CPU; NVIDIA Tesla v100 GPU, 32 Gb of VRAM.

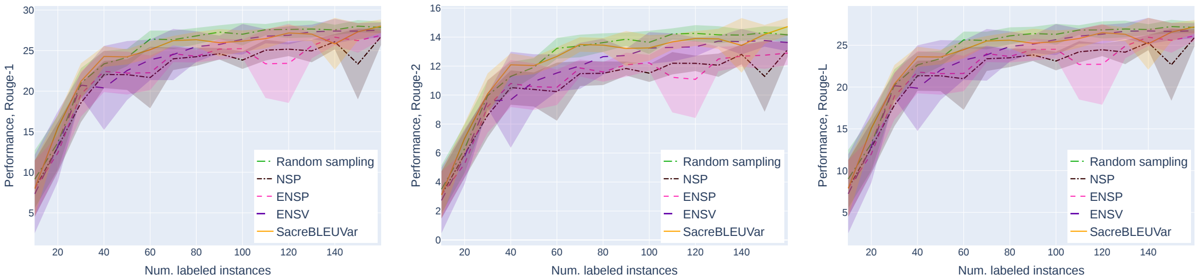
Hparam	BART	PEGASUS
Checkpoint	facebook/bart-base	google/pegasus-large
# Param.	139M	570M
Number of epochs	6	4
Batch size	16	2
Gradient accumulation steps	1	8
Min. number of training steps	350	200
Max. sequence length	1024	1024
Optimizer	AdamW	AdamW
Learning rate	2e-5	5e-4
Weight decay	0.028	0.03
Gradient clipping	0.28	0.3
Sheduler	STLR	STLR
% warm-up steps	10	10
Num. beams at evaluation	4	4
Generation max. length	Adapt.	Adapt.

B Full Results for Uncertainty-based Methods



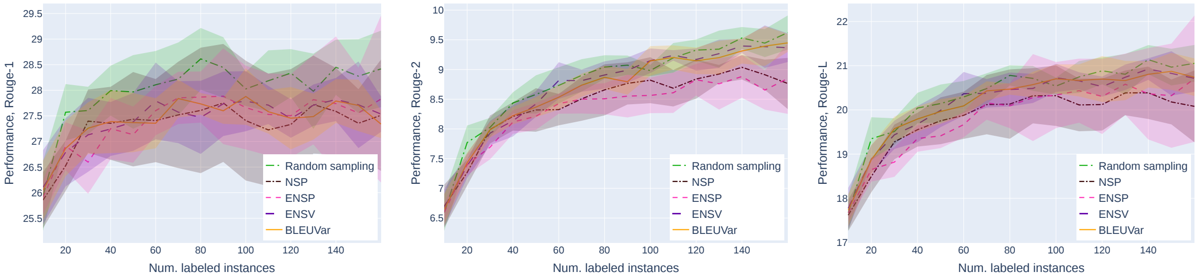
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 6: The performance of the BART-base model with various uncertainty-based strategies compared with random sampling (baseline) on AESLC.



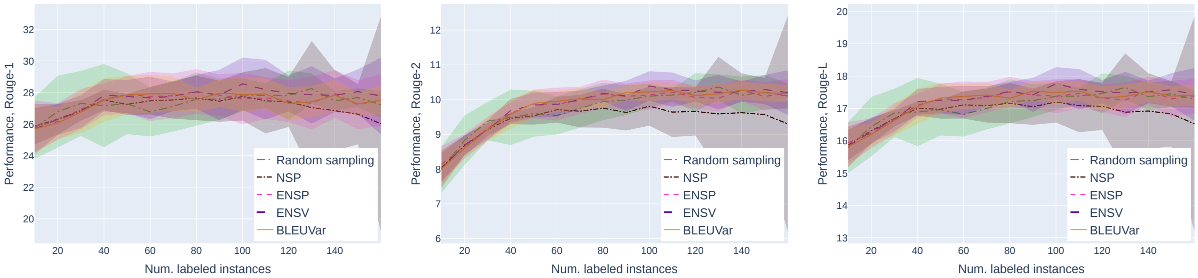
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 7: The performance of the PEGASUS-large model with various uncertainty-based strategies compared with random sampling (baseline) on AESLC.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 8: The performance of the BART-base model with various uncertainty-based strategies compared with random sampling (baseline) on WikiHow.



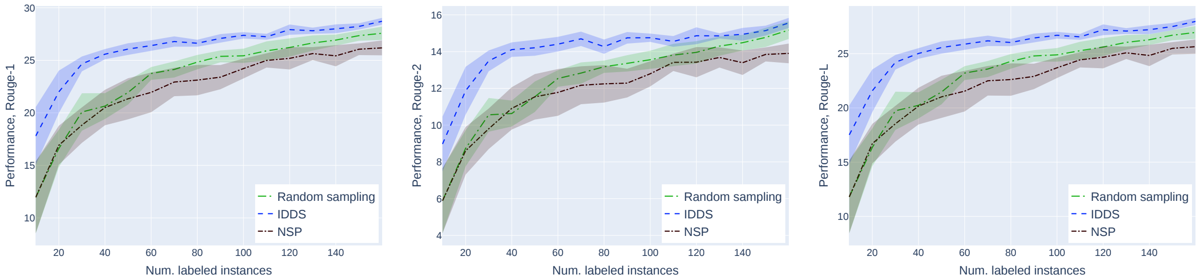
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 9: The performance of the BART-base model with various uncertainty-based strategies compared with random sampling (baseline) on PubMed.

C Full Results for IDDS

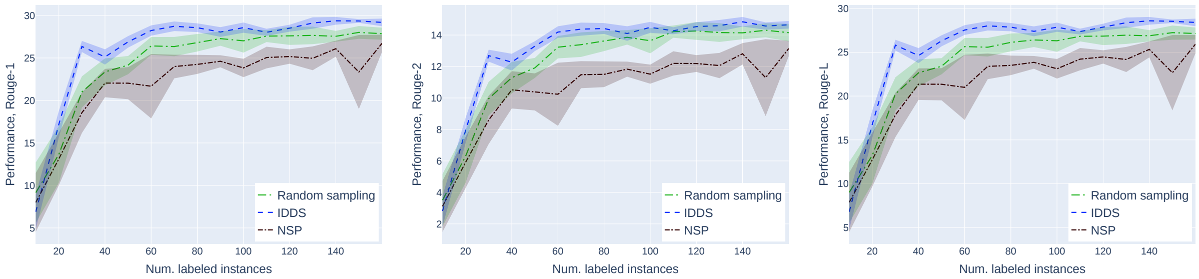
Iter. 0 R-1/R-2/R-L	Iter. 5 R-1/R-2/R-L	Iter. 10 R-1/R-2/R-L	Iter. 15 R-1/R-2/R-L	Average R-1/R-2/R-L
AESLC + BART-base				
48.8 / 52.5 / 48.4	11.2 / 14.9 / 11.4	5.2 / 5.4 / 5.0	4.1 / 2.6 / 3.8	10.2 / 11.9 / 10.0
AESLC + PEGASUS-large				
-24.8 / -19.7 / -24.5	6.9 / 7.3 / 7.4	1.6 / 0.4 / 2.0	4.8 / 3.5 / 4.7	7.6 / 6.7 / 8.0
WikiHow + BART-base				
6.3 / 12.5 / 5.4	1.9 / 2.7 / 1.3	3.0 / 4.2 / 2.5	2.6 / 2.9 / 1.8	2.3 / 3.2 / 1.5
PubMed + BART-base				
8.0 / 10.4 / 5.8	12.0 / 11.7 / 8.0	8.1 / 6.4 / 4.9	9.5 / 6.7 / 5.1	8.9 / 7.7 / 5.5

Table 4: Percentage increase in ROUGE F-scores of IDDS over the baseline on different AL iterations. **Average** refers to the average increase throughout the whole AL cycle.



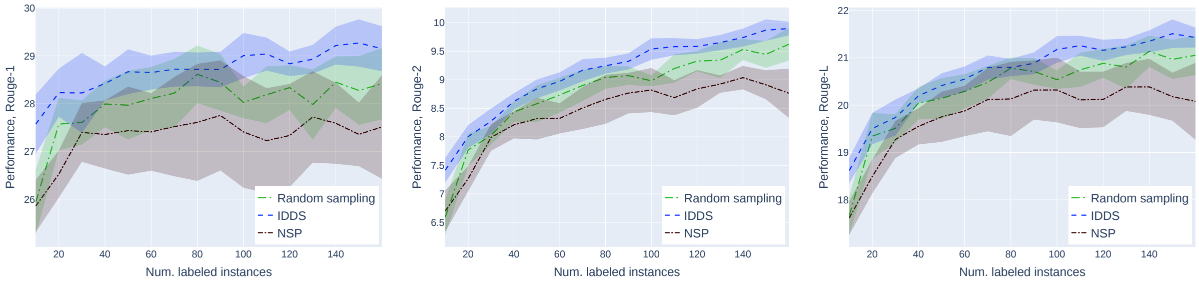
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 10: The performance of the BART-base model with the IDDS strategy compared with random sampling (baseline) and NSP (uncertainty-based strategy) on AESLC.



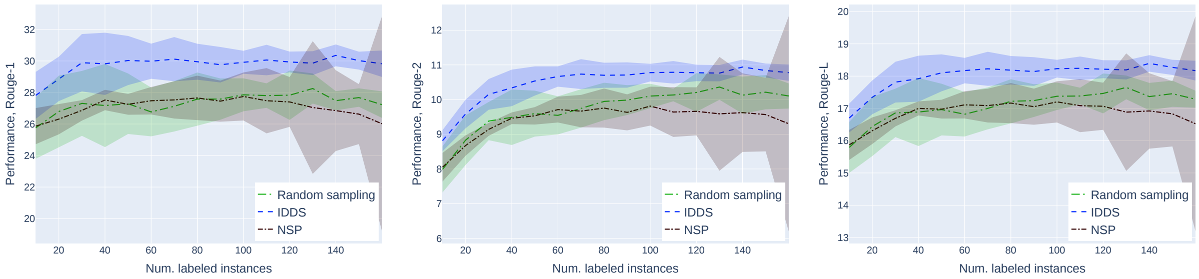
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 11: The performance of the PEGASUS-large model with the IDDS strategy compared with random sampling (baseline) and NSP (uncertainty-based strategy) on AESLC.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

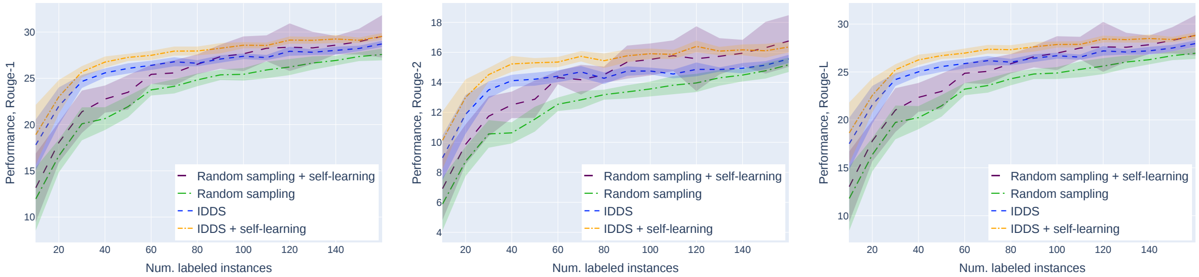
Figure 12: The performance of the BART-base model with the IDDS strategy compared with random sampling (baseline) and NSP (uncertainty-based strategy) and NSP (uncertainty-based strategy) on WikiHow.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 13: The performance of the BART-base model with the IDDS strategy compared with random sampling (baseline) and NSP (uncertainty-based strategy) on PubMed.

D Full Results for Self-learning

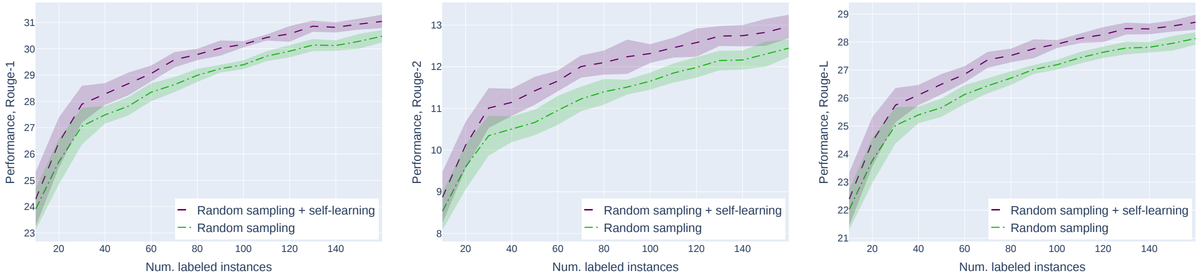


a) ROUGE-1

b) ROUGE-2

c) ROUGE-L

Figure 14: The performance of the BART-base model with the IDDS and random sampling strategies with and without pseudo-labeling of the unlabeled data on AESLC ($k_l = 0.1, k_h = 0.01$).

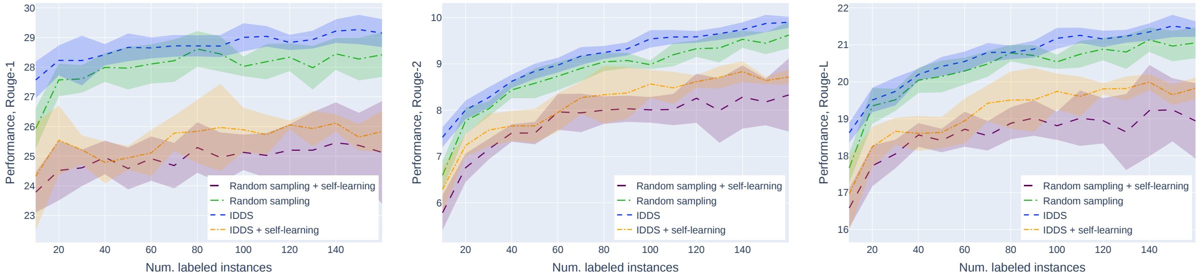


a) ROUGE-1

b) ROUGE-2

c) ROUGE-L

Figure 15: The performance of the BART-base model without AL (random sampling) with and without pseudo-labeling of the unlabeled data on the randomly sampled subset of Gigaword ($k_l = 0.1, k_h = 0.01$).



a) ROUGE-1

b) ROUGE-2

c) ROUGE-L

Figure 16: The performance of the BART-base model with the IDDS and random sampling strategies with and without self-supervised learning on WikiHow ($k_l = 0.38, k_h = 0.02$).

E Diversity Statistics and Query Examples

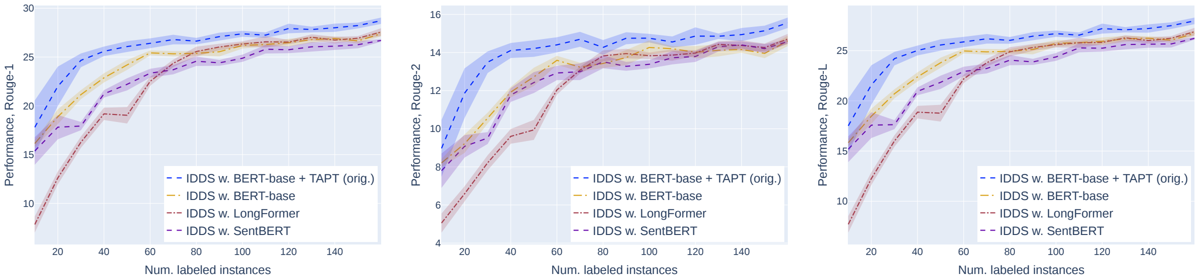
AL Strat.	Document	Golden summary	Gen. summary
IDDS	"Here's the latest info. regarding **Bloomberg's** ability to accept **deals** with **Pinnacle** West (formerly **Arizona Public Service** Co.) (...)	**Bloomberg-Pinnacle/APS deals**	n/a
IDDS	Hi. Nice to see you in Houston. I'm giving a **presentation** on **gas** issues on Tuesday. I've got a draft of (...)	**Gas Presentation**	n/a
IDDS	Kelley, I am writing to you to (...) Can you give me the name and contact information for the person within your company that would work with us to put a **Confidentiality Agreement** in place (...)	**confidentiality agreement**	n/a
NSP	**tantivy** (tan-TIV-ee) adverb At full gallop; at full speed. noun A fast gallop; rush.adjective Swift.interjection A hunting cry by a hunter riding a horse at full speed(...)	**A.Word.A.Day–tantivy**	tricky (tan-TIV-ee) adjective
NSP	Prod Area and Long Haul k# **Volume** Rec Del 3.6746 5000 St 62 Con Ed 3.4358 15000 St 65 Con Ed 3.5049 10000 St (...)	**TRCO capacity for Sep**	Prod Area and Long Haul k# Volume
NSP	This is a list of RisktRAC book-ids corresponding to what has been created in ERMS. Let me know if the book-id naming is ok with you. Regards	**Book2.xls**	RisktRAC
ENSP	Fred, I suggest a phone call among the team today to make sure we are all on the same wave length. What is your schedule? Thanks	**PSEG**	Firm schedule
ENSP	Stephanie - When you get a chance, could you finalize the attached (also found in Tana's O drive). I am not sure where the originals need to go after signed by Enron, but I have a request for that information currently out to **Hess**. Thanks.	**Hess NDA**	Enron O Drive
ENSP	Current Notes User: To ensure that you experience a successful migration from Notes to Outlook, it is necessary to gather individual user information prior to your date of migration. Please take a few minutes to completely fill out the following survey (...)	**2- SURVEY /INFORMATION EMAIL 5-17-01**	Office 2000 Migration Survey
Sacre-BleuVAR	Sheri, We are going to NO for JazzFest at the end of April. April 27th-29th to be exact. Let me know if you're going. DG	**southwest.com** weekly specials	JazzFest
Sacre-BleuVAR	This warning is sent automatically to inform you that your mailbox is approaching the maximum size limit. Your mailbox size is currently 78515 KB. Mailbox size limits (...)	**WARNING: Your mailbox is approaching the size limit**	Mailbox size limit

Table 5: Examples of the instances queried with different AL strategies. Tokens overlapping with the source document are highlighted with **green**. Tokens that refer to paraphrasing the part of the document and the corresponding part are highlighted with **blue**. Tokens that cannot be derived from the document are highlighted with **red**. Tokens, the usage of which depends on the peculiarities of the dataset, are not highlighted. Summaries for IDDS are not presented, because IDDS does not require model inference.

AL Iter.	SP	ESP	SacreBleuVAR	Random	IDDS
1	33.3% / 0%	30.0% / 4.4%	0% / 0%	0% / 0%	0% / 0%
6	15.6% / 0%	0% / 1.1%	0% / 2.2%	0% / 0%	0% / 0%
15	3.3% / 0%	0% / 0%	0% / 0%	0% / 2.2%	0% / 0%
Mean	7.8% / 1.0%	2.1% / 0.8%	0.1% / 0.3%	0% / 0.7%	**0% / 0%**

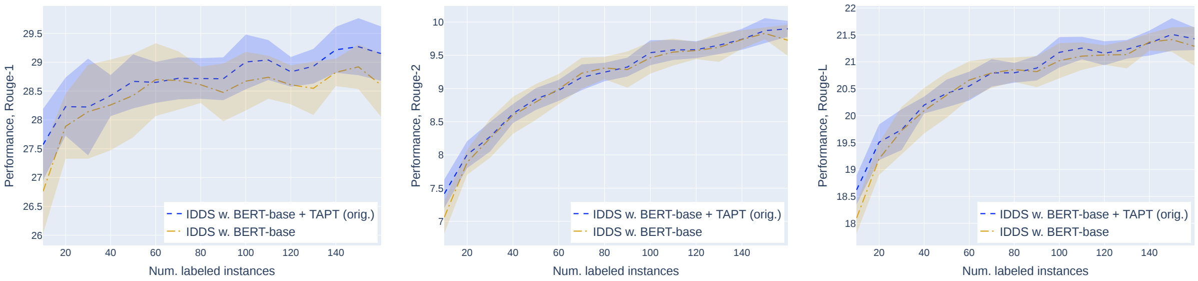
Table 6: Share of fully / partly overlapping summaries among batches of instances, queried with various AL strategies during AL using BART-base model on AESLC. We consider two summaries to be partly overlapping if their ROUGE-1 score > 0.66. The results are averaged across 9 seeds for all the strategies except for IDDS, which has constant seed-independent queries.

F Ablation Studies of IDDS



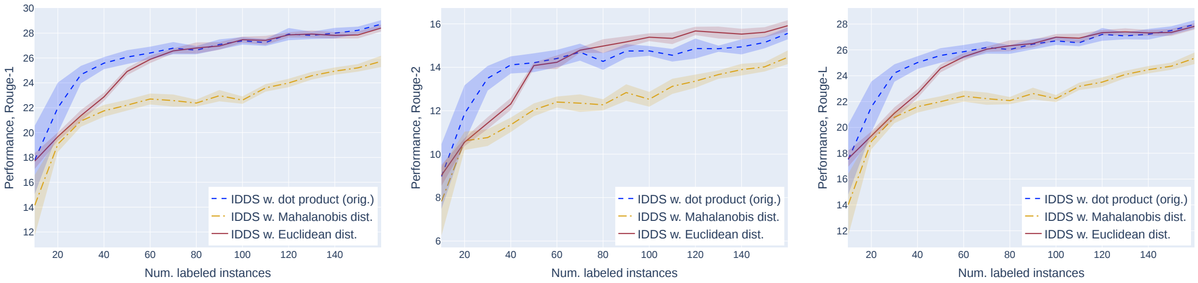
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 17: Ablation study of the document embeddings model & the necessity of performing TAPT for it in the IDDS strategy with BART-base on AESLC.



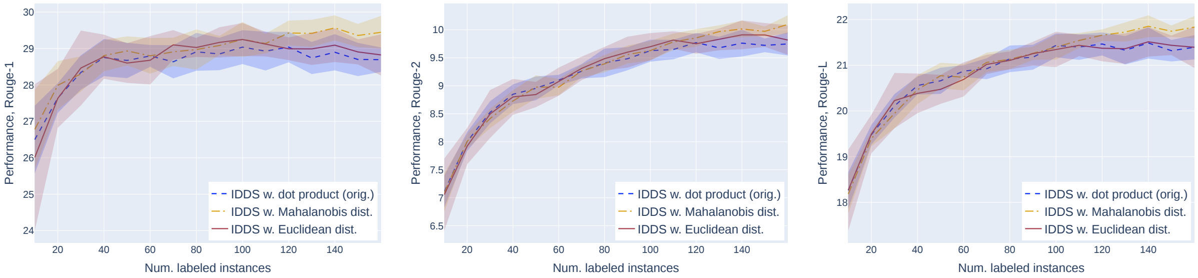
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 18: Ablation study of the necessity of performing TAPT for the model, which generates embeddings in the IDDS strategy with BART-base on WikiHow.



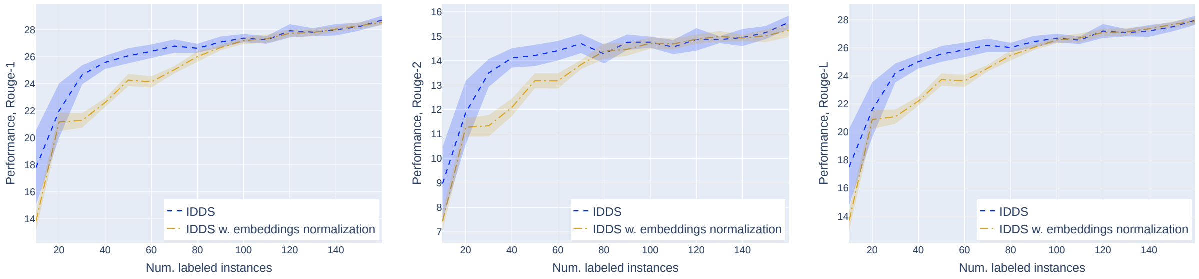
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 19: The performance of IDDS with different similarity functions with BART-base on AESLC.



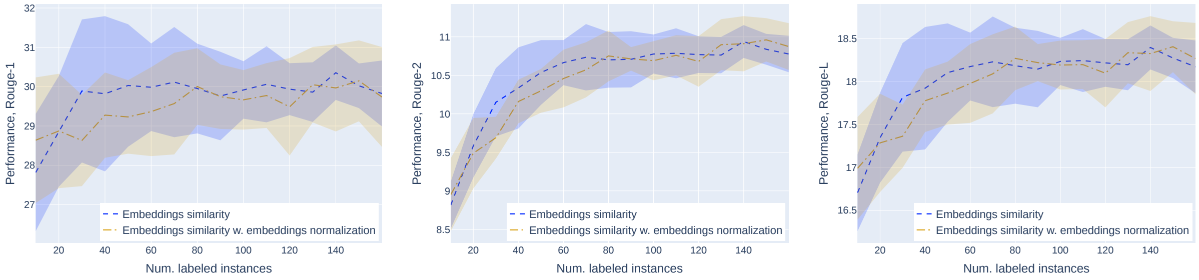
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 20: The performance of IDDS with different similarity functions with BART-base on WikiHow.



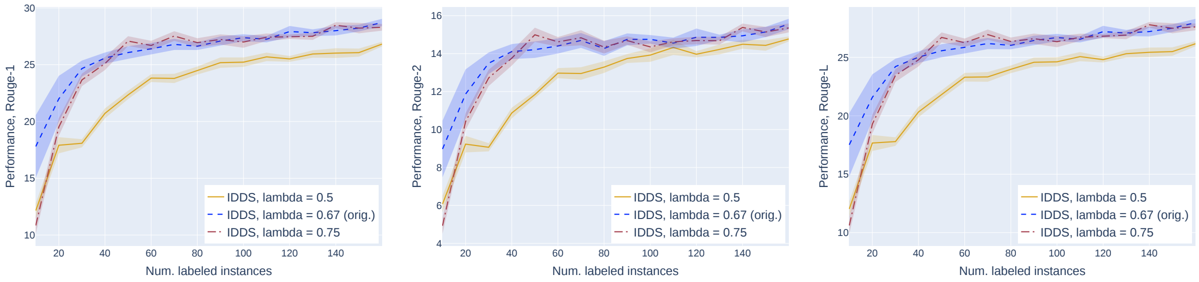
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 21: The performance of the BART-base model with the standard IDDS strategy compared with its modification when embeddings are normalized on AESLC.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 22: The performance of the BART-base model with the standard IDDS strategy compared with its modification when embeddings are normalized on PubMed.

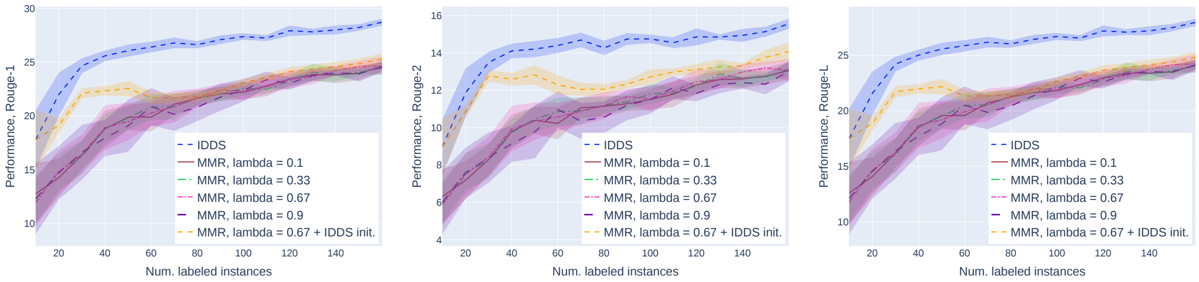


a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 23: Ablation study for the hyperparameter λ in the IDDS strategy with BART-base on AESLC.

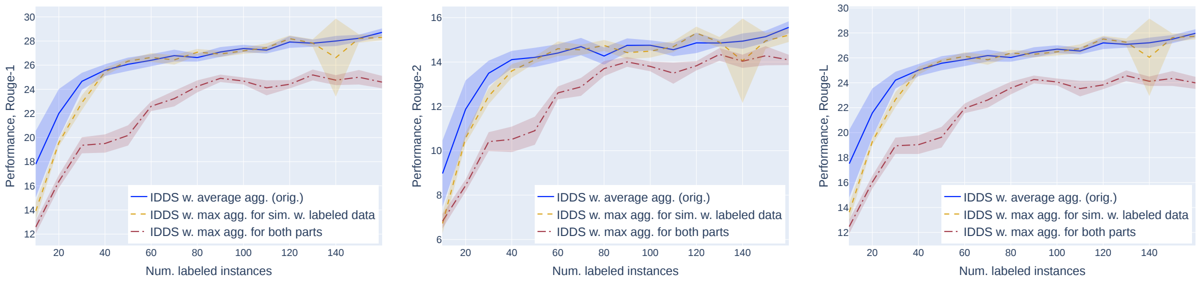
AL Strategy	**Iter. 0**	**Iter. 5**	**Iter. 10**	**Iter. 15**	**Average**
λ = 0.	9.08 / 4.8 / 8.79	19.6 / 10.87 / 19.29	22.6 / 12.58 / 22.1	23.68 / 13.32 / 23.23	21.25 / 11.72 / 20.88
λ = 0.33	15.77 / 7.67 / 15.46	22.47 / 12.18 / 22.07	23.98 / 13.54 / 23.51	24.68 / 13.81 / 24.21	23.19 / 12.88 / 22.78
λ = 0.5	12.15 / 6.07 / 12.03	23.82 / 12.97 / 23.3	25.69 / 14.33 / 25.06	26.81 / 14.77 / 26.17	23.84 / 12.94 / 23.31
λ = 0.67 (orig.)	**17.8 / 8.97 / 17.52**	**26.4 / 14.4 / 25.86**	**27.25 / 14.55 / 26.55**	**28.72 / 15.56 / 27.97**	**26.7 / 14.43 / 26.07**
λ = 0.75	10.84 / 4.93 / 10.61	**26.7 / 14.62 / 26.26**	**27.42 / 14.62 / 26.72**	28.29 / **15.36** / 27.61	**26.54 / 14.31 / 26.0**
λ = 0.83	16.47 / 7.84 / 16.03	26.06 / **14.42** / 25.59	26.57 / 14.12 / 25.92	**28.7** / 15.22 / **28.0**	26.0 / 13.93 / 25.46
λ = 1.	16.41 / **8.66** / 16.23	25.2 / 13.66 / 24.72	26.74 / 14.4 / 26.04	27.44 / 14.66 / 26.67	25.73 / 13.81 / 25.16

Table 7: ROUGE scores on AL iterations for different values of the $lambda$ hyperparameter in IDDS. We select with **bold** the largest values w.r.t. the confidence intervals.



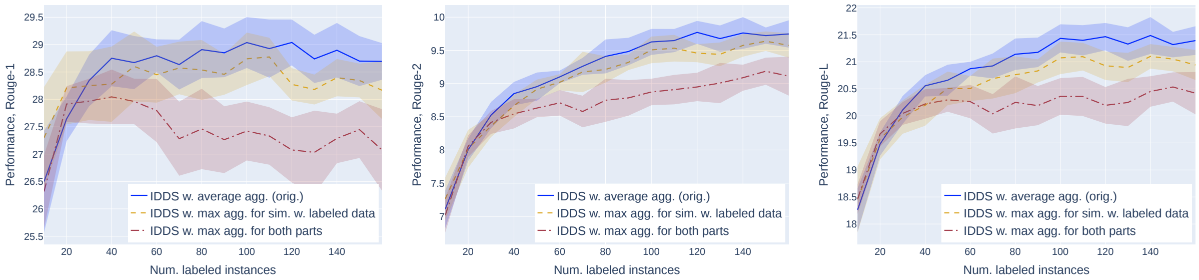
a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 24: Comparison of IDDS with the MMR-based strategy suggested in (Kim et al., 2006) with BART-base on AESLC. We experiment with different λ values in MMR and the initialization schemes.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 25: Comparison of the *average* and *maximum* aggregation functions in IDDS with BART-base on AESLC.



a) ROUGE-1 b) ROUGE-2 c) ROUGE-L

Figure 26: Comparison of the *average* and *maximum* aggregation functions in IDDS with BART-base on WikiHow.

G Additional Experiments with Consistency Analysis

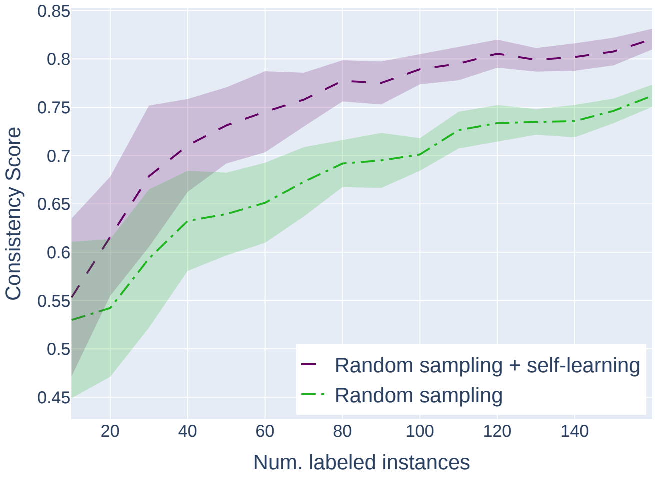


Figure 27: The consistency score calculated via SummaC with BART-base on Gigaword without AL (random sampling) with and without self-learning.

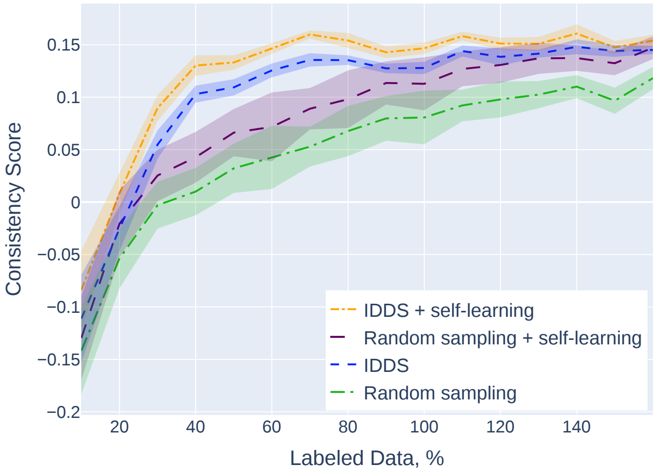


Figure 28: The consistency score calculated via SummaC on the test sample on AESLC for BART-base with the IDDS and random sampling strategies with and without self-learning.

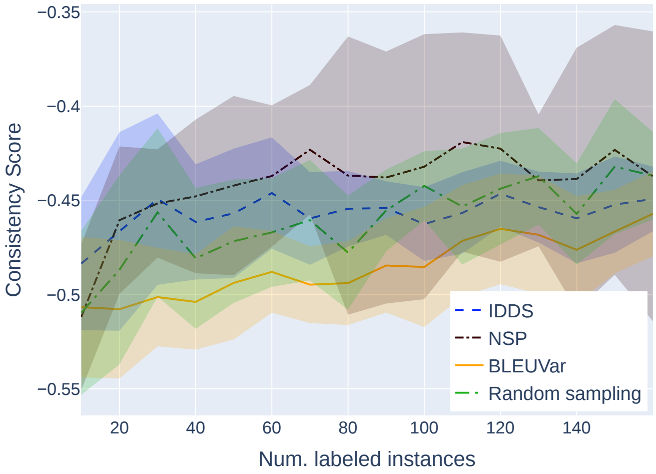


Figure 29: The consistency score calculated via SummaC on the test subset of WikiHow for the BART-base model with various AL strategies.

H Query Duration

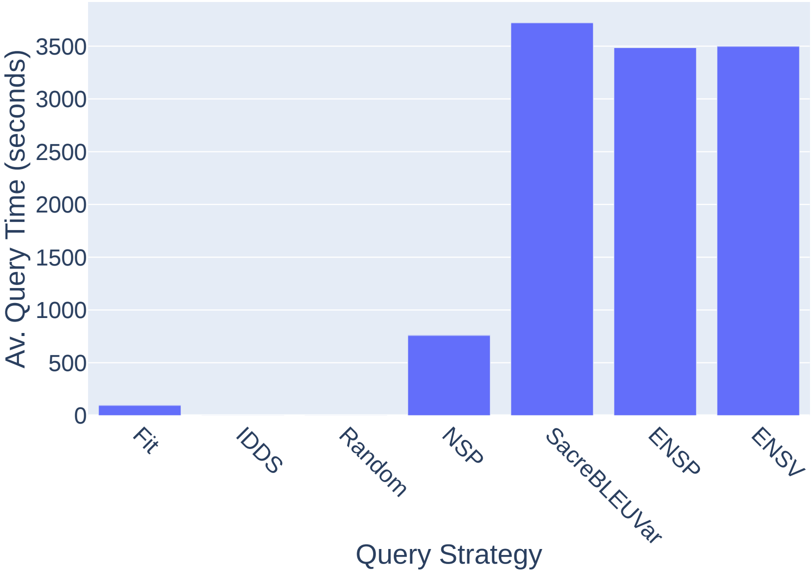


Figure 30: Average duration in seconds of one AL query of 10 instances with different strategies on the AESLC dataset with BART-base as an acquisition model. *Train* refers to the average time required for training the model throughout the AL cycle. Hardware configuration: 2 Intel Xeon Platinum 8168, 2.7 GHz, 24 cores CPU; NVIDIA Tesla v100 GPU, 32 Gb of VRAM.