Active Learning Efficiency Benchmark for Coreference Resolution including Advanced Uncertainty Representations

1st Marko Sahan
Dept. of Computer Science, FEE, CTU in Prague
Prague, Czechia
sahanmar@fel.cvut.cz

2nd Václav Šmídl
Dept. of Computer Science, FEE, CTU in Prague
Prague, Czechia
smidlva1@fel.cvut.cz

3rd Taro Watanabe
NLP Laboratory, Div. of Inf. Science, NAIST
Nara, Japan
taro@is.naist.jp

4th Radek Mařík
Dept. of Telecom. Engineering, FEE, CTU in Prague
Prague, Czechia
radek.marik@fel.cvut.cz

***Abstract*—Active learning is a powerful technique that accelerates model learning by iteratively expanding training data based on the model's feedback. This approach has proven particularly relevant in natural language processing and other machine learning domains. While active learning has been extensively studied for conventional classification tasks, its application to more specialized tasks like neural coreference resolution has the potential for improvement. In our research, we present a significant advancement by applying active learning to the neural coreference problem, and setting a benchmark of 39% reduction in required annotations for training data. Simultaneously, it preserves performance compared to the original model trained on the full data. We compare various uncertainty sampling techniques along with Bayesian modifications of coreference resolution models, conducting a comprehensive analysis of annotation efforts. The results demonstrate that the best-performing techniques seek to maximize label annotation in previously chosen documents, showcasing their effectiveness and preserving performance.**

***Index Terms*—active learning, Bayesian neural networks, neural coreference resolution, RoBERTa**

I. Introduction

The active learning (AL) technique attempts to solve the problem of iterative smart unlabeled data selection to maximize annotation efficiency. The selection of documents to annotate at random may not bring enough discriminability to the model [1], which can skew model decisions. The basis of active learning lies in the uncertainty measurement of the machine learning (ML) model's prediction. The vast amount of works in AL, especially in natural language processing (NLP) [2], [3] show undeniable improvement in performance over random selection of unlabeled data. Active learning in NLP is a generally well-studied field with contributions in text classification [4], fake news detection [5], NER [6], etc. However, the field of active learning for coreference resolution (CR) still has potential for improvement.

Coreference resolution is the task of determining if linguistic expressions in text refer to the same entity. Major contributions in AL for CR, such as [7]–[9], show model performance scores for different sampling techniques given a measurement of annotation efforts time or number of sampled instances. Therefore, previous works lack large-scale evaluation of AL methods on full benchmark data to CR models trained on a complete training set. Consequently, the extent of annotations required and the comparative performance of active learning approaches versus models trained on fully annotated dataset remains uncertain.

Some works propose improvement in AL sampling based on Bayesian uncertainty representation. Recent results in [10] show that the Bayesian approach of estimating model parameter distribution for text classification can improve model training by better instance sampling. Nevertheless, the improvement was dataset-specific, meaning that the approach worked well on particular data only. The aforementioned setup had not been previously tested for CR problems. Thus, the hypothesis of better uncertainty capturing led us to experiment with the Bayesian model parameters estimation integrated into the instance sampling process.

Our contribution sets the baseline of a minimized training dataset for AL sampling by achieving equivalent model performance trained on the full OntoNotes 5.0 corpus and CR labels from CoNLL-2012 Shared Task [11]. Specifically:

- We conducted close to real-life annotation experiments where a model was limited to the number of documents from which it was allowed to sample tokens. The output evidence demonstrated a 39% reduction of the original training data, which preserved the model's performance as if trained on full OntoNotes 5.0 corpus.
- We modified a close to state-of-the-art CR Dobrovolskii [12] model and introduced uncertainty-based instance sampling methods: i) random token ii) random mention [9] iii) entropy mention [9], and iv) HAC entropy mention. The last method was generalized to a CR problem for the first time from the image recognition task [13] and proved to work well with higher levels of uncertainty.
- We implemented and tested a Bayesian approach on a CR Dobrovolskii [12] model such as MC Dropout [14] for a more granular model parameter distribution estimation, making us the first to test Bayesian uncertainty representation on neural coreference models.

II. Related Works

Some experiments in active learning date back several decades [1] before neural networks took over the field. Since that time, interest in the field increased from different (ML) domains like image recognition [14]–[16], natural language processing [5], [16], etc.

The model's uncertainty for unlabeled data helps to choose annotation candidates that will be labeled and added to the training dataset. Iteratively repeating the querying and annotating processes creates a training dataset that maximizes the model's understanding of the data. Multiple works compare the performance of uncertainty measurement methods under different conditions. A broad sensitivity study based on the type of dataset from [17] reports different winning methods for different data. Throughout multiple studies such as [2], [3], entropy-based uncertainty measurement along with its modifications like HAC min-margin introduced in [13] and modified for NLP problems in [10] have a pattern of performing better compared to the rest of methods, especially random training data selection.

The epistemic uncertainty is the uncertainty that is measured for unlabeled data querying in the AL step. It is caused by a lack of training data. Therefore, based on [18] it can be minimized with more input samples. Aleatoric uncertainty is the inherent unpredictability affected by uncontrollable randomness and can be minimized with an appropriately chosen model. As a consequence, an uncertainty representation has a significant role in AL. The Bayesian approach of uncertainty representation from [14], [19] aims to estimate model parameters distribution instead of point-wise estimate. The experiments for text classification with single instance sampling from [5] and batch active learning from [10] showed that Bayesian methods like MC Dropout defined in [14] and deep ensembles showed in [19] can outperform models with only point-wise parameters distribution. This occurred for both new initializations in every AL step and continuation training of the classifier with reusing weight from previous iterations.

Neural CR models are relatively new, with the first end-to-end-like Feed Forward neural network-based model outperforming rule-based approaches shown in [20] in 2016. At that time, the only non-neural network sub-models were syntactic parsers. The first fully end-to-end CR model that uses only gold mention clusters is based on LSTM layers and was presented the following year in [21]. The current state-of-the-art CR model from [22] is significantly more advanced and represented as a seq2seq large language model with modified prompting and output to detect antecedents. In this work, we use one of the last pre-prompting based CR models from [12] that uses sub-models and will be subsequently explained in detail.

The ratio of coreferenced (positive) tokens in a document to noncoreferenced (negative) tokens is very unbalanced. Thus, classic AL methods require modifications to more class unbalanced problems with a solution in between instances vs. document choice. The comparison of document selection, instances selection, and combined approaches from [23] do not show a significant improvement for random document choice. A better compromise between documents vs. instance sampling was shown in [9]. It involves limitations for document selection a more sophisticated annotation process and proposes using models' sub-models for unlabeled data by querying mentions classifier. The output from [9] became a seed idea for our work. A more sophisticated annotation method from [8] follows annotating the first antecedent instead of sampling pairs of tokens.

III. Methods

A. Neural Coreference Resolution

We follow the definition of CR problem from [24] with a modification in [12]. The goal is to learn a distribution $P(y_i|d)$ over antecedents for each token i in a document d:

$$P(y_i|d) = \frac{e^{s(i,y_i)}}{\sum_{y' \in \mathbf{Y}_d(i)} e^{s(i,y')}}, \tag{1}$$

where $s(i,j)$ is a pairwise score for a coreference link between token i and token j, and $\mathbf{Y}_d(i) = \{\epsilon, 1, 2, ..., i-1|d\}$ is a set of possible antecedents for token y_i in document d. Symbol ϵ is a dummy antecedent that covers two cases: i) the span is not an entity mention or ii) the span is an entity mention, but it is not coreferent with any previous span. Compared to [24], we define antecedents, mentions, and coreferences through tokens but not spans. Coreferent spans are retrieved in a post-processing step by inference through a span detector model.

The CR model from [12] can be decomposed into an encoding model and a CR superstructure that consists of three sub-models. The text encoding model throughout the article is RoBERTa [25]. The CR superstructure on the top of RoBERTa consists of

- **Token representation** is an attention layer that creates mention embedding for each token $i \in d$ by applying attention weights on encoded tokens.
- **Mention detector** is a sub-model that returns a list of k antecedents for every input token where $k = 50$. This step is useful to further reduce the computational complexity of the model.
- **Coreference scorer** performs pair-wise coreference detection on antecedents from the mention detector.
- **Span detector** is only applied on tokens that are found to be coreferent to some other tokens. The module reconstructs the span for each token by predicting the most probable start and end tokens in the same sentence.

B. Active Learning

Classical AL text classification approaches as shown in [10] perform document sampling and reach significant improvement compared to the random selection. Earlier attempts of AL for CR presented in [23] demonstrate that document selection strategies do not outperform random selection and show that a sampling decision must be made on the instance level. Moreover, the previous approach in [26] follows pairwise

annotations and labels if a sampled pair is coreferent. The downside to pairwise annotations is that it requires many labels. A newer annotating approach from [8], [9] is called a discrete annotation and involves annotation of the closest antecedent of a sampled instance. In the later work of [9], instance sampling is limited to freedom of document selection by introducing a constraint for the number of documents that instances can be selected from. The selection of antecedents to annotate throughout different documents may result in sparse annotations and encasement of annotation efforts. We will refer to documents that we can sample from in each AL step as *documents of interest*. More specifically *documents of interest* is a term that represents the number of documents the model is allowed to choose from in each AL iteration.

The output of CR model is a coreference matrix of words and antecedents containing 1 at position (i, j) if $i-$th and $j-$th words corefer

$$\hat{\mathbf{C}} = \mathbf{C}(\mathbf{X}_d, \theta) \tag{2}$$

where $\mathbf{X}_d$ is a set of encoded instances $\mathbf{x}_d$ (tokens in terms of the paper experiments) from document d, and a set of CR model parameters θ. CR clusters are derived from a combination of a coreference scorer, which is an estimate of equation 1, and a span detector. In a supervised setting, the classifier parameters are found on a training set $[\mathbf{X}_d, \mathbf{C}_d]_{d=1}^{D}$ by matching the prediction $\hat{\mathbf{C}}_d$ with provided label $\mathbf{C}_d$ for each document $d \in \{1, ..., D\}$ and all instances in every document.

In AL approach, all instance labels in the document are rarely available. Hence, we introduce a notation of set of token indices $\mathcal{I}_d^t$ for document $d \in \{1, \ldots, D\}$ and iteration $t = 0, \ldots, T$. In further parts, the notation helps us to define a subset of document instances that are and are not annotated.

For $t-$th active learning iteration, only a subset of tokens that form clusters is available. The full set $\cup_{d=1}^{D}\mathbf{X}_d$ is thus split into the labeled

$$\mathbf{X}_{\mathcal{I}^t} = \cup_{d=1}^{D}\mathbf{X}_{\mathcal{I}_d^t},\ \mathbf{C}_{\mathcal{I}^t} = \cup_{d=1}^{D}\mathbf{C}_{\mathcal{I}_d^t}, \tag{3}$$

and unlabeled parts

$$\begin{aligned}\mathbf{X}_{\mathcal{I}^t}^u &= \cup_{d=1}^{D}(\mathbf{X}_d \backslash \mathbf{X}_{\mathcal{I}_d^t}),\\ \mathbf{C}_{\mathcal{I}^t}^u &= \cup_{d=1}^{D}(\mathbf{C}_d \backslash \mathbf{C}_{\mathcal{I}_d^t}).\end{aligned} \tag{4}$$

Active learning for coreference resolution is illustrated in Algorithm 1.

C. Uncertainty Measurement and Sampling

In the paper, four different uncertainty-based sampling options and three different types of uncertainty measurement methods are tested. Authors from [9] published great results in favor of using the mention detector sub-model, for instance sampling. We applied and modified this approach on a more advanced CR model from [12]. The mention detector model returns k most probable mentions (antecedent candidates) for each input token. Antecedent candidates are extracted with the argmax function applied to the softmax output of neural network prediction. Uncertainty sampling strategies are:

Algorithm 1 Active Learning for Coreference Resolution

Initialize: $t = 0$ iterations counter, m number of documents of interest, $\mathcal{I}^0 = \{\}$, initial value of parameter $\theta^{(0)}$ is random, token scoring strategy $A = A(\mathbf{X}, \theta, m)$, CR classifier $\mathbf{C} = \mathbf{C}(\mathbf{X}, \theta)$, and s tokens to sample.

for cycles $t = 1, ..., T$ **do:**

- $\mathcal{A} = A(\mathbf{X}_{\mathcal{I}^{t-1}}^u, \theta^{(t-1)}, m)$ - set of all unlabeled token scores. Token scores from m documents of interest are prioritized for token selection by having higher scores. Selection of m documents of interest is also done by A.
- $\tilde{\mathbf{X}}$ is created by iteratively selecting tokens with the highest score from $\mathcal{A}$ and annotating it along with the closest antecedent (if exists) until s tokens are selected.
- $\tilde{\mathbf{C}}$ denotes s annotated tokens for $\tilde{\mathbf{X}}$.
- $[\mathbf{X}_{\mathcal{I}^t}, \mathbf{C}_{\mathcal{I}^t}] = [\mathbf{X}_{\mathcal{I}^{t-1}} \cup \tilde{\mathbf{X}}, \mathbf{C}_{\mathcal{I}^{t-1}} \cup \tilde{\mathbf{C}}]$ is labeled data update, and $\mathbf{X}_{\mathcal{I}^t}^u = \cup_{d=1}^{D}(\mathbf{X}_d \backslash \mathbf{X}_{\mathcal{I}_d^t})$ is unlabeled data update.
- Train a classifier parameter $\theta^{(t)}$ with $[\mathbf{X}_{\mathcal{I}^t}, \mathbf{C}_{\mathcal{I}^t}]$.

return: $\theta^{(T)}$

- **random token sampling** does purely random selection of s token from m randomly selected documents of interest without any model knowledge.
- **random mention sampling** performs a random selection of s tokens from a mention detector sub-model and m randomly selected documents of interest.
- **entropy mention sampling** calculates the entropy value for every token given antecedent candidates from a softmax mention detector output. Next, m documents of interest based on the highest tokens' mention entropy are selected. Afterward, s tokens with the highest mention entropy scores from m documents of interest are chosen.
- **HAC entropy mention sampling** is a modification of HAC min-margin introduced in [13] for binary classification problem and then generalized to the entropy-based version and text classification problems from [10]. The method performs hierarchical agglomerative clustering on document encoding. The document encoding is calculated as a mean value of encoded tokens. Next, document entropy is calculated as an average mention entropy. Afterwards, m documents of interest that belong to the smallest nonsingleton clusters and have the highest document mention entropy are selected. In the end, s tokens with the highest mention entropy are chosen from m documents of interest. The aim of the method is to diversify selected documents.

Throughout the active learning process, the same documents can be selected multiple times in different AL steps if they have unlabeled data.

D. Uncertainty Representation

The uncertainty can be represented by a maximum likelihood estimate, which is represented by a single network, or

a Bayesian probabilistic estimate. The Bayesian estimate is typically denoted by an ensemble of networks. The output from ensembles for the mention detector is calculated as a normalized expected value of softmax output for every ensemble result. Next, same as for the non-Bayesian approach, softmax values are passed further to select top k mentions. The softmax probability for every ensemble is conditioned on the neural network parameters θ and thus captures only aleatoric uncertainty, i.e. inherent unpredictability affected by uncontrollable randomness. Ensemble of networks represents uncertainty in parameters by different parameter values in each ensemble, thus capturing both aleatoric and epistemic uncertainty (i.e. model uncertainty due to lack of data). We consider the MC dropout method from [14]. The method is an extension of the ordinary dropout that samples binary mask multiplying the output of a layer, hence stopping propagation through all neurons where zeros are sampled through the network. The extension applies the sampled mask even for predictions generating samples from the predictive distribution. Experiments in [5] and [10] showed that this method performs the best out of all other Bayesian methods on text classification problems in an active learning setup.

E. Evaluation

There is a variety of methods for measuring active learning performance such as the area under ROC [27], presented in [10] or a combination of accuracy and precision from [13]. However, the metrics for coreference resolution are well-defined through the years. The most common metric of measuring coreference models performance is a mean value of three $F1$ scores such as B^3 [28], MUC [29], and $\mathrm{CEAF}_{\phi 4}$ [30]. The metrics can also be compared separately. The discriminability of these metrics is argued, and a new metric LEA metric is proposed in [31]. The metric represents how well a specific entity is resolved by capturing the entity size as a measure of importance. We chose LEA as a primary metric for AL simulations. Besides previously named metrics, we also introduce the document annotation ratio metric. The metric shows the average annotations ratio over the documents that have at least one annotated token and is defined as follows,

$$\text{annot. ratio} = \frac{1}{\tilde{D}} \sum_{i=1}^{D} \frac{\#\text{ annotated tokens }_i}{\#\text{ all tokens }_i} \tag{5}$$

where $\tilde{D}$ is the number of all documents with at least one annotated token.

IV. Experiment Setup

Data: active learning experiments are run on OntoNotes 5.0 data, which consists of 2802 documents and around 1.3 million tokens of training data, 348 documents of test data, and 343 documents of validation records. **Model and model parameters:** experiments use base coreference Dobrovolskii [12] model with RoBERTa [25] text encoding. The original article [12] suggests 20 epochs of training. However, for the purposes of AL, we use two configurations of 10 and 18 epochs for different experiments. Simulations helped us to conclude that the best performance of the model can be estimated for a maximum of 18 epochs. The results are obtained based on different variations of documents of interest for $m = 50$ and $m = 200$. For the Bayesian network and MC Dropout model, the number of sampled ensembles is 5. **Active learning:** All experiments perform 50 active learning steps. The number of tokens sampled in every active learning step is $s = 20000$. Hence, starting with 0 available tokens, active learning simulations end with 1 million sampled tokens. We provide results for random tokens selection, random mention selection, entropy mention selection, and HAC entropy mention selection. These experiments are run both for the base CR model and the MC dropout version of the CR model. All simulations are repeated 4 times for statistical validation. **Hardware:** the training is performed on 4 x NVIDIA Tesla $A100$ with 40GB graphic memory and NVLink3 interconnection. **Evaluation metrics:** LEA is used for demonstration and comparison of active learning simulation results. In addition, we also provide results for annotation ratio metric, the number of documents to read for annotation in every AL step, and classic CR metrics such as MUC, $\mathrm{CEAF}_{\phi 4}$, and B^3. The latter three metrics are provided for the winning strategy and a comparison to the original Dobrovolskii [12] model trained on all data.

V. Results and Discussions

A. Base Model and a Freedom of Choice

The first set of experiments was performed on a base model with four different sampling strategies: i) random tokens sampling, ii) random mentions sampling, iii) entropy mentions sampling, and iv) HAC mentions sampling. LEA results aggregated over four simulations for four active learning scoring strategies and two different sets of documents of interest are presented in Table Ia.

The data show better results for entropy-based strategies and 200 documents of interest. The results for 50 documents of interest show a better start for the random strategy with a further overtake by entropy-based sampling. Random token sampling with 50 documents of interest follows the strategy of document exhaustion in each active learning iteration because the average tokens count for an article in the training data is around 400 tokens. Hence, bringing more complete information about the dataset in the early stage of simulations. In Figure 1a we can see that the plot of annotation ratio metric has a relatively high value for every strategy. Therefore, a lack of freedom of document choice for algorithms bounds them for weaker performance. Whereas results for 200 documents of interest and entropy-based methods in Figure 1b act differently. Higher values of documents of interest let these methods skim the space better and find the most valuable documents across the unlabeled dataset. Thus, the difference between random strategies and entropy-based strategies became even more significant.

Entropy sampling: in Figure 1b, we can see that even despite higher freedom of document selection, the mention

TABLE I: Active learning average LEA metric aggregated over 4 AL simulations for 5 AL iterations choice, 4 different AL sampling strategies, two variations of documents of interest, 20000 tokens sampling, and **two** coreference models with RoBERTa text encoding trained on 10 epochs and fresh initialization in every AL step. The best result across all methods and documents of interest is denoted with a **bold** font. The best result across all methods for a specific variation of document of interest setup is denoted by a star. Results within one standard deviation of the winner (tagged by a star) are displayed in red for 50 documents of interest and in blue for 200 documents of interest

(a) Base coreference model

AL iterations	5	20	30	40	50	doc. of int.
random token	0.607*	0.723*	0.742*	0.753	0.752	50
	0.316	0.705	0.727	0.741	0.753	200
random mention	0.598	0.722	0.737	0.752	0.761	50
	0.651	0.725	0.737	0.748	0.757	200
HAC entropy mention	0.573	0.711	0.741	0.756	0.762	50
	0.702	0.729	**0.756***	0.761	0.766	200
entropy mention	0.541	0.731	0.740	0.757*	0.768*	50
	0.714*	**0.743***	0.756	**0.763***	**0.767***	200

(b) MC Dropout coreference model

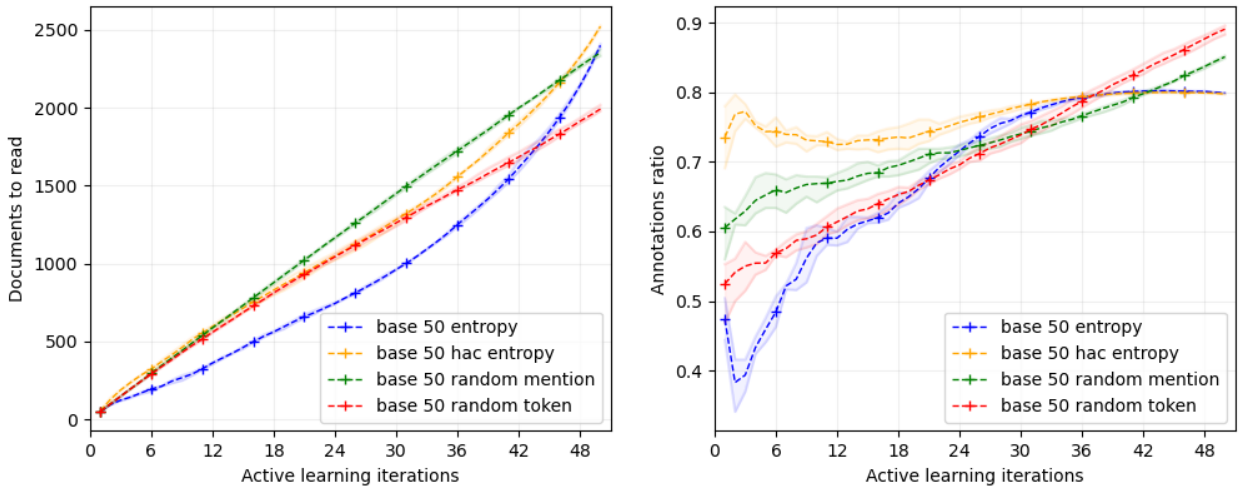
AL iterations	5	20	30	40	50	doc. of int.
random token	0.565	0.706	0.724	0.742	0.745	50
	0.587	0.698	0.726	0.743	0.750	200
random mention	0.573*	0.714*	0.734	0.739	0.754	50
	0.591	0.708	0.728	0.744	0.755	200
HAC entropy mention	0.563	0.702	0.731	0.749*	0.760*	50
	0.671	**0.744***	**0.752***	**0.758***	**0.763***	200
entropy mention	0.380	0.696	0.735*	0.744	0.760	50
	0.674*	0.742	0.750	0.755	0.762	200

TABLE II: Active learning simulation average LEA, documents to read, and annotation ratio metrics through 4 runs. The results are displayed for 6 AL iterations choice, 20000 tokens sampling, 2 different AL sampling strategies such as **random token with 50 documents of interest**, and **entropy mentions with 200 documents of interest**. The CR model is a **base** model with RoBERTa text encoding trained on **18 epochs** and fresh initialization in every AL step. The best result across all methods and specific metrics is denoted with a **bold** font. The best results for LEA and annotation ratio are the highest values and the lowest for documents to read

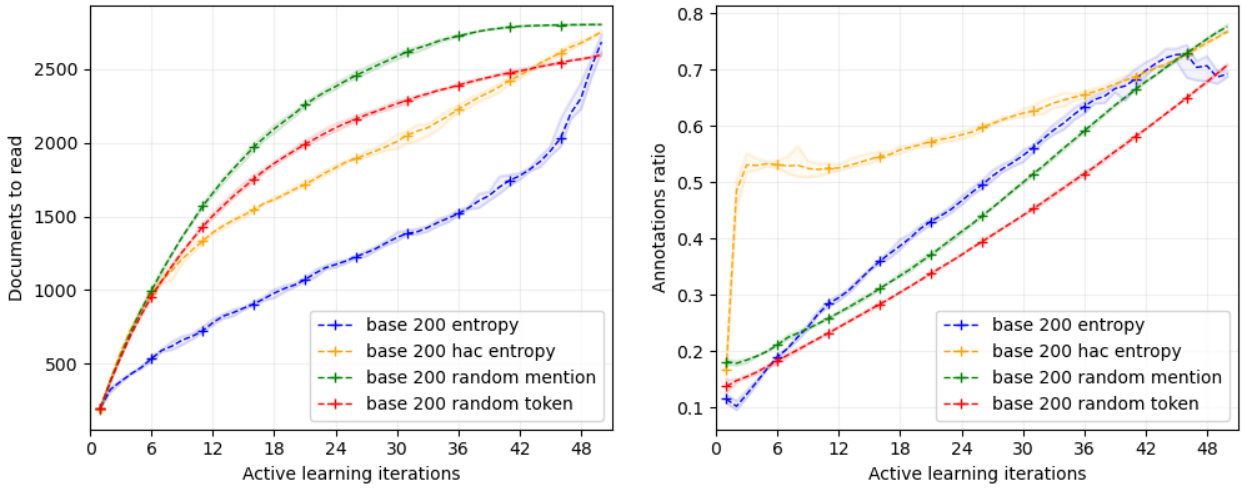
AL iterations	40	42	44	46	48	50	metric
random token	0.747	0.754	0.750	0.754	0.755	0.757	LEA
entropy mention	**0.768**	**0.765**	**0.769**	**0.771**	**0.772**	**0.770**	
random token	2039	2259	2511	2652	2728	2770	doc. to read
entropy mention	**1691**	**1769**	**1944**	**2052**	**2403**	**2712**	
random token	0.650	0.625	0.593	0.597	0.619	0.650	annot. ratio
entropy mention	**0.680**	**0.708**	**0.701**	**0.718**	**0.684**	**0.692**	

TABLE III: MUC, $\text{CEAF}_{\phi 4}$, B^3, and $F1$ mean metrics for the original model from Dobrobolskii [12] model trained on 2802 documents and 1.3 million tokens, the best performing random token model out of four AL simulation for 50 documents of interest with 18 training epochs from 50th AL iteration with LEA $= 0.757$ trained on 2791 documents and 1 million tokens, and the best entropy mention model out of four AL simulation for 200 documents of interest with 18 training epochs from 42nd AL iteration with LEA $= 0.775$ trained on 1715 documents and 840000 tokens

	MUC			$\text{CEAF}_{\phi 4}$			B^3			
	P	R	F1	P	R	F1	P	R	F1	Mean F1
Dobrovolskii [12]	84.9	87.9	86.3	76.1	77.1	76.6	77.4	82.6	79.9	81.0
random token	82.2	87.9	84.9	73.0	76.7	74.8	73.5	82.6	77.8	79.1
entropy mention	84.5	87.6	86.3	76.1	76.3	76.2	77.0	82.24	79.5	80.7



(a) 50 documents of interest



(b) 200 documents of interest

Fig. 1: Mean values over four AL simulations for documents to read and annotations ratio metrics for 50 active learning steps, 4 different AL sampling strategies, **two variations of documents of interest**, and base coreference model with RoBERTa text encoding trained on 10 epochs in every AL step. Uncertainty bounds are shown as maximal and minimal values through simulations

entropy sampling is very conservative in new document selection and prefers to query annotations from previously selected documents. The best LEA metric for the experiment setup described previously was reached for 200 documents of interest, and the entropy mention sampling with **LEA** $= 0.764 \pm 0.004$. The best run out of 4 simulations has **LEA** $= 0.772$ reached on **44** iteration with **Sampled documents** $= 1872 \pm 10$.[1] We can conclude that we reached a 33% documents reduction in annotation efforts with entropy mention sampling and only 10 training epochs for the base model with almost identical performance as in originally submitted article [12] trained on all data and 20 epochs.

HAC entropy sampling: in Figure 1a and 1b the annotation ratio for HAC entropy strategy is the highest from the beginning almost until the end. In combination with results from documents to read graphs, it can be observed that the algorithm prefers to sample smaller documents. The scores for HAC entropy sampling and entropy sampling are comparable. Thus, if the dataset has smaller documents, the HAC strategy will perform even better thanks to the clustering diversification of the input data.

B. MC Dropout Model

MC Dropout model is a modification of a base model where the dropout layer in the mentions detector is on during the prediction step. The mentions detector makes five predictions when called. Samples represent an empirical distribution estimate. The normalized average value of the mention detector outputs is a further input for subsequent submodels. LEA

[1]As a comparison, the LEA metric for the submitted model from [12] is 0.775 with 20 training epochs and 2802 documents

metrics aggregated over four simulations for MC Dropout are displayed in table Ib. The output did not show better results and overall performed worse compared to the base model in Table Ia. However, we can see that HAC entropy sampling in tables Ia and Ib is able to perform almost as well as without added noise to the model. This can be justified with a more diverse data sampling thanks to the clustering approach. Hence, the model is more robust to the added noise. Documents to read and annotation ratio patterns for the MC Dropout model act the same as in figures 1a and 1b. Thus, additional plot visualization is omitted due to a lack of new information.

The experiments followed the work from [10]. Our attempts include: i) different calibration of dropout parameter, ii) turning on the dropout layer during the prediction in RoBERTa encoding, iii) reusing model weights from the previous active learning step (hot-start), and iv) experimenting with a combination of fresh initialization (cold-start) retraining and hot-start training. The latter was tried in a setup of 5 and 10 hot-start AL steps between fresh retraining. None of the listed experiments showed better or at least the same performance as the base model. Overall, the results were $2-4$ LEA scores lower compared to the base model. We justify these results with a high information density inside the model across all neurons. Thus, turning off even some of them during the prediction step by means of dropout resulted in information loss.

C. Base Model with a Deeper Analysis

Based on results from previous sections, we chose the best-performing method, such as entropy mention sampling for 200 documents of interest, and trained it vs. random token sampling for 50 documents on interest with the increased number of training epochs in every AL step. Both simulations were run for 4 times with 18 training epochs. The rest of the simulation setup remained unchanged. The simulation results are visualized in Table II. It is seen that even for more training epochs, the results do not show a significant difference to results in Table Ia with only 10 training epochs. Nevertheless, starting from the 44th iteration, average LEA scores for the entropy model show a pattern of almost top model performance from [12] with a difference in less than three decimal places and significant reduction of documents to read with a high level of annotation ratio. The best-performing result out of all 4 simulations for entropy mention with 200 documents of interest and 18 training epochs is reached on the 42nd AL iteration and is **LEA** $= 0.775$ with **Documents to read** $= 1715$ what is 39% of training data reduction in annotation efforts.

The comparison of MUC, $\text{CEAF}_{\phi 4}$, and B^3 for the best-performing entropy mention AL step (reached on 42nd iteration with **LEA** $= 0.775$), the best performing random mention AL step (reached on 50th iteration with **LEA** $= 0.757$) and the results from [12] are shown in table III. As seen from the results, mean $F1$ for entropy mention is less by only 0.3 from the leader, given a 39% reduction of training data in annotation efforts.

VI. Conclusion

We introduce an active learning approach for the model from Dobrovolskii [12] and set a benchmark result for annotations efforts reduction of 39%. The output results were reached by experimenting with 4 different sampling approaches: random token, random mention, entropy mention, and HAC entropy mention sampling. The additional key factor is a detailed study of the document sampling freedom. Despite broad freedom in document selection (200 documents of interest), the method showed that it prefers conservative document selection in each active learning iteration. This resulted in the same LEA performance as the original Dobrovolskii model trained on the full dataset while using only 61% of labeled data. The random token selection model with 50 documents of interest that mimed the document exhaustion strategy resulted in sampling even more documents with lower documents selection freedom.

Finally, we experimented with a more complex uncertainty representation by applying the MC Dropout approach to a more complex type of model, such as coreference resolution. Unfortunately, the hypothesis of better uncertainty representation that would lead to better results did not prove itself to work. The output of MC Dropout leads to a slight performance loss due to lower discriminability. Nevertheless, the HAC entropy mentions sampling method showed that if the model is enriched with additional uncertainty that leads to a discriminability loss, more diverse data provided by the described method allows it to perform the best as compared to other techniques.

In conclusion, we would like to say that the output of the work shows the strength of AL techniques in a selection of instances to annotate compared to the model that was trained on full data. This data can be a baseline for future comparisons and design of annotation tasks.

Acknowledgment

This research was supported by the project TL05000057, The Technology Agency of the Czech Republic www.tacr.cz, within the ETA Programme. In addition, the authors acknowledge the support of the OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 “Research Center for Informatics”. Last but not least, we would like to express our gratitude to Sadie Abraham for her expert language review.

References

[1] S. Tong and D. Koller, “Support vector machine active learning with applications to text classification,” *Journal of machine learning research*, vol. 2, no. Nov, pp. 45–66, 2001.

[2] P. F. Jacobs, G. Maillette de Buy Wenniger, M. Wiering, and L. Schomaker, “Active learning for reducing labeling effort in text classification tasks,” in *Benelux Conference on Artificial Intelligence*. Springer, 2021, pp. 3–29.

[3] U. Ahmed and J. C.-W. Lin, “Deep explainable hate speech active learning on social-media data,” *IEEE Transactions on Computational Social Systems*, 2022.

[4] C. Schröder and A. Niekler, “A survey of active learning for text classification using deep neural networks,” *arXiv preprint arXiv:2008.07267*, 2020.

[5] M. Sahan, V. Smidl, and R. Marik, "Active learning for text classification and fake news detection," in *2021 International Symposium on Computer Science and Intelligent Controls (ISCSIC)*. IEEE, 2021, pp. 87–94.

[6] Y. Shen, H. Yun, Z. C. Lipton, Y. Kronrod, and A. Anandkumar, "Deep active learning for named entity recognition," *arXiv preprint arXiv:1707.05928*, 2017.

[7] B. Z. Li, G. Stanovsky, and L. Zettlemoyer, "Active learning for coreference resolution using discrete annotation," *arXiv preprint arXiv:2004.13671*, 2020.

[8] ——, "Active learning for coreference resolution using discrete annotation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 8320–8331. [Online]. Available: https://aclanthology.org/2020.acl-main.738

[9] M. Yuan, P. Xia, C. May, B. Van Durme, and J. Boyd-Graber, "Adapting coreference resolution models through active learning," *arXiv preprint arXiv:2104.07611*, 2021.

[10] M. Sahan, V. Smidl, and R. Marik, "Batch active learning for text classification and sentiment analysis," in *Proceedings of the 2022 3rd International Conference on Control, Robotics and Intelligent System*, 2022, pp. 111–116.

[11] S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Y. Zhang, "Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes," in *Joint conference on EMNLP and CoNLL-shared task*, 2012, pp. 1–40.

[12] V. Dobrovolskii, "Word-level coreference resolution," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 7670–7675. [Online]. Available: https://aclanthology.org/2021.emnlp-main.605

[13] G. Citovsky, G. DeSalvo, C. Gentile, L. Karydas, A. Rajagopalan, A. Rostamizadeh, and S. Kumar, "Batch active learning at scale," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 933–11 944, 2021.

[14] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," in *International conference on machine learning*. PMLR, 2017, pp. 1183–1192.

[15] G. Hacohen, A. Dekel, and D. Weinshall, "Active learning on a budget: Opposite strategies suit high and low budgets," *arXiv preprint arXiv:2202.02794*, 2022.

[16] A. Tsvigun, I. Lysenko, D. Sedashov, I. Lazichny, E. Damirov, V. Karlov, A. Belousov, L. Sanochkin, M. Panov, A. Panchenko *et al.*, "Active learning for abstractive text summarization," *arXiv preprint arXiv:2301.03252*, 2023.

[17] S. Prabhu, M. Mohamed, and H. Misra, "Multi-class text classification using bert-based active learning," *arXiv preprint arXiv:2104.14289*, 2021.

[18] A. Der Kiureghian and O. Ditlevsen, "Aleatory or epistemic? does it matter?" *Structural safety*, vol. 31, no. 2, pp. 105–112, 2009.

[19] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[20] K. Clark and C. D. Manning, "Improving coreference resolution by learning entity-level distributed representations," *arXiv preprint arXiv:1606.01323*, 2016.

[21] K. Lee, L. He, M. Lewis, and L. Zettlemoyer, "End-to-end neural coreference resolution," *arXiv preprint arXiv:1707.07045*, 2017.

[22] B. Bohnet, C. Alberti, and M. Collins, "Coreference resolution through a seq2seq transition-based system," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 212–226, 2023.

[23] T. Miller, D. Dligach, and G. Savova, "Active learning for coreference resolution," in *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, 2012, pp. 73–81.

[24] K. Lee, L. He, and L. Zettlemoyer, "Higher-order coreference resolution with coarse-to-fine inference," *arXiv preprint arXiv:1804.05392*, 2018.

[25] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[26] M. Sachan, E. Hovy, and E. P. Xing, "An active learning approach to coreference resolution," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[27] C. E. Metz, "Basic principles of roc analysis," in *Seminars in nuclear medicine*, vol. 8, no. 4. Elsevier, 1978, pp. 283–298.

[28] A. Bagga and B. Baldwin, "Algorithms for scoring coreference chains," in *The first international conference on language resources and evaluation workshop on linguistics coreference*, vol. 1. Citeseer, 1998, pp. 563–566.

[29] M. Vilain, J. D. Burger, J. Aberdeen, D. Connolly, and L. Hirschman, "A model-theoretic coreference scoring scheme," in *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995*, 1995.

[30] X. Luo, "On coreference resolution performance metrics," in *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, 2005, pp. 25–32.

[31] N. S. Moosavi and M. Strube, "Which coreference evaluation metric do you trust? a proposal for a link-based entity aware metric," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 632–642.