Deep Bayesian Active Learning for Preference Modeling in Large Language Models

Luckeciano C. Melo[*1,2] **Panagiotis Tigas**[1] **Alessandro Abate**[†2] **Yarin Gal**[†1]
[1] OATML, University of Oxford [2] OXCAV, University of Oxford

Abstract

Leveraging human preferences for steering the behavior of Large Language Models (LLMs) has demonstrated notable success in recent years. Nonetheless, data selection and labeling are still a bottleneck for these systems, particularly at large scale. Hence, selecting the most informative points for acquiring human feedback may considerably reduce the cost of preference labeling and unleash the further development of LLMs. Bayesian Active Learning provides a principled framework for addressing this challenge and has demonstrated remarkable success in diverse settings. However, previous attempts to employ it for Preference Modeling did not meet such expectations. In this work, we identify that naive epistemic uncertainty estimation leads to the acquisition of redundant samples. We address this by proposing the **B**ayesian **A**ctive **L**earner for **P**reference **M**odeling (BAL-PM), a novel stochastic acquisition policy that not only targets points of high epistemic uncertainty according to the preference model but also seeks to maximize the entropy of the acquired prompt distribution in the feature space spanned by the employed LLM. Notably, our experiments demonstrate that BAL-PM requires 33% to 68% fewer preference labels in two popular human preference datasets and exceeds previous stochastic Bayesian acquisition policies.

1 Introduction

Preference Modeling is a key component to aligning unsupervised pre-trained Large Language Models (LLMs) towards human preferences [1–4]. It is often performed by collecting human feedback for a set of prompt-completion pairs and then leveraging the data to steer the behavior of such models, either directly [5] or via reward models [6]. Nevertheless, human feedback generation is laborious [7], especially when it requires specialized knowledge [8, 9]. Furthermore, the quality of the prompts has a crucial impact on the performance of fine-tuned models [10]. Hence, selecting the most informative points to gather feedback is essential to reduce costs and enable better LLMs.

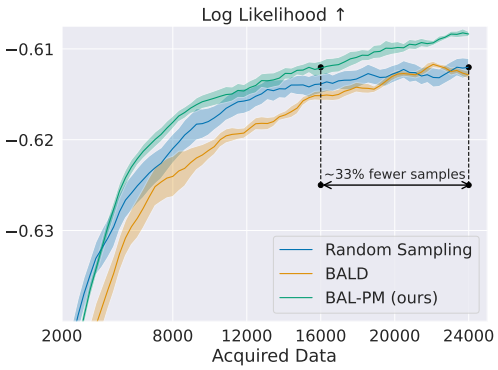


Figure 1: **Log-Likelihood of learned preference models in the Reddit TL;DR dataset** [1]. Our method, BAL-PM, reduces the volume of required human feedback by 33% over random acquisition.

Despite its substantial impact, data selection for Preference Modeling poses a significant challenge. The prompt-completion pool is arbitrarily

[*]Correspondence to: luckeciano.carvalho.melo@cs.ox.ac.uk
[†]Denotes equal supervision.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-3749

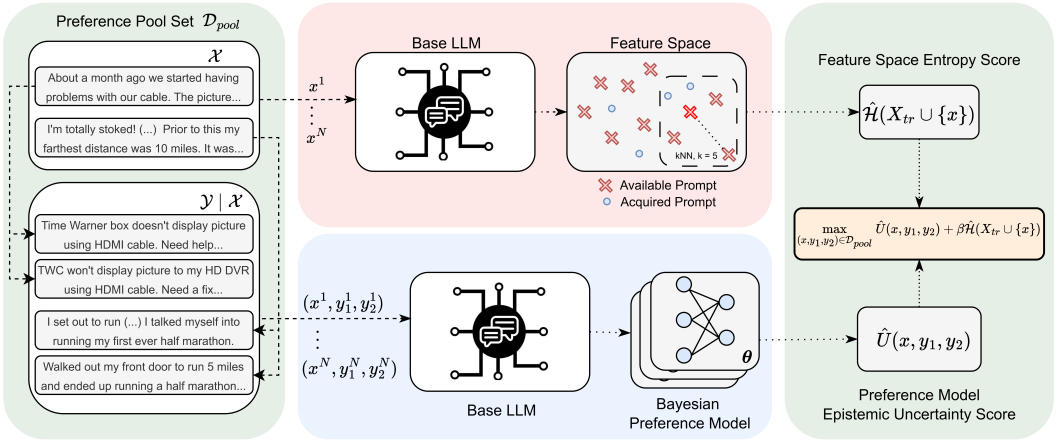


Figure 2: **An illustration of how BAL-PM works**. For each tuple $(x, y_1, y_2) \in \mathcal{D}_{pool}$, we obtain features for the prompt and prompt-completion pairs by computing the last layer embeddings of the base LLM. We leverage the prompt feature space to estimate the entropy score of the acquired prompt distribution, $\hat{\mathcal{H}}(X_{train} \cup \{x\})$. Similarly, we use the prompt-completion features as input for the Bayesian Preference Model, which is used to estimate task-dependent epistemic uncertainty scores, $\hat{U}(x, y_1, y_2)$. BAL-PM selects the tuple that maximizes the linear combination of both scores.

large and semantically rich. Additionally, human feedback is inherently noisy, with low agreement rates among labelers, typically observed between 60% – 75% for these settings [6, 1, 11, 12]. Lastly, the intrinsic scale of LLM development requires parallelized labeling and makes frequent model updates prohibitively expensive, limiting the applicability of many active learning schemes that rely on single-point acquisition [13].

Bayesian Active Learning provides a principled approach to data selection [14–16], which has demonstrated remarkable success across different fields [17–19]. However, its application in Active Preference Modeling is not straightforward. Past attempts of employing the framework in this setting reported no benefits over random selection [20], arguably due to poor uncertainty estimation in the context of LLMs, which is indeed an open challenge and active area of research [21].

We identify two reasons for this phenomenon. First, the inherent bias of approximate Bayesian inference in deep learning models, particularly for LLMs. Second, and more nuanced, the current intractability of epistemic uncertainty estimation methods in Preference Modeling for LLMs, a context that intrinsically requires batch acquisition. Proper estimators for this setting present combinatorial complexity, and even greedy approximations are still computationally demanding and impractical [13, 22]. This limitation leads to relying on simpler single-point acquisition schemes such as BALD [23] (as in Gleave and Irving [20]), designed to acquire individual points followed by model updates. However, these assumptions are far from realistic for the scale of Preference Modeling in LLMs, and naively applying such methods for batch acquisition leads to the selection of redundant samples.

In this work, we argue that leveraging the information available from the feature space spanned by the LLM – a task-agnostic[3] source of epistemic uncertainty – alleviates these problems. We propose **B**ayesian **A**ctive **L**earner for **P**reference **M**odeling (BAL-PM), a novel stochastic acquisition policy that not only targets points of high epistemic uncertainty according to the preference model but also seeks to maximize the entropy of the acquired prompt distribution in the feature space. This entropy score encourages the active learner to select prompts from low-density regions, effectively reducing the feature space epistemic uncertainty [24]. As a result, it promotes diversity in the acquired training set, preventing the selection of redundant samples and also helping in learning a better Bayesian preference model and its task-dependent epistemic uncertainty estimates for subsequent acquisitions. Figure 2 illustrates how BAL-PM works.

We conduct active learning experiments in the Reddit and CNN/DM preference datasets [25, 26, 1] to validate our method. BAL-PM demonstrates strong gains over random sampling, reducing by approximately 33% (as shown in Figure 1) and 68% the volume of feedback required to learn the

[3]"task" refers to Preference Modeling. A task-agnostic estimation is independent of preference labels.

preference model in the considered datasets. It also consistently surpasses other strong stochastic Bayesian acquisition policies [22]. Finally, we further analyze the acquired prompt distribution to show that BAE-PM prevents redundant exploration and effectively balances the contribution of the two sources of epistemic uncertainty.

2 Related Work

Bayesian Active Learning is an established form of active learning that leverages the uncertainty in model parameters to select the most informative points [14, 27, 16], demonstrating relevant impact in several applications [19, 18, 17, 28, 29]. In this work, we apply this technique for Preference Modeling [30, 31] in LLMs. Given the requirements of such a problem setting, we focus on batch acquisition [14, 13], particularly in the design of stochastic acquisition policies, similarly to Kirsch et al. [22]. However, our work fundamentally differs from theirs in the strategy of incorporating stochasticity. The policies introduced by Kirsch et al. [22] directly sample from the distribution determined by the single-point, task-dependent epistemic uncertainty scores. In contrast, our method maximizes the entropy of the acquired data distribution, which allows leveraging a task-agnostic source of epistemic uncertainty, alleviating the effect of biased task-dependent uncertainty scores.

Active Preference Modeling leverages active learning techniques to reduce the feedback needed for Preference Modeling [27]. There has been a recent surge of interest in the area [32–35, 20, 36] given the impact of Preference Optimization in fine-tuning LLMs [2, 5, 10, 6]. A portion of this line of work focuses on query generation to directly optimize preferences. Mehta et al. [32] theoretically formalizes the problem and proposes a method that generates one completion to maximize the uncertainty of the triple in a kernelized setting. Das et al. [35] proposes a method based on confidence bands, accounting for both completions in the triple and relaxing linearity assumptions on the reward function. Ji et al. [33] constructs an optimistic estimator for the reward gap between completions and selects those with the least gap, using an uncertainty estimator to reduce query complexity. Lastly, Dwaracherla et al. [36] generate completions using double Thompson Sampling, representing epistemic uncertainty with an Epistemic Neural Network [37], similar to our Bayesian model. Overall, while having the shared goal of reducing the volume of human feedback, query generation is orthogonal to our problem setting. Instead, we focus on the pool-based setting, as in Gleave and Irving [20], which allows us to leverage real human feedback in experiments rather than relying on synthetic preference simulators. Gleave and Irving [20] was the first attempt of Bayesian Active Learning in this setting, and it directly applies BALD acquisition [23] for Preference Modeling, using a fully fine-tuned deep ensemble for epistemic uncertainty estimation. In contrast, our work proposes a new objective that extends BALD acquisition to account for the entropy of the acquired prompt distribution to encourage the acquisition of more diversified samples, and formulates the Bayesian model as an ensemble of adapters.

Task-Agnostic Uncertainty Estimation refers to a set of techniques that quantifies uncertainties based on density estimation of the input in a learned latent feature space [38, 39]. In this context, distant points from the training set offer more information about the input space, which is useful for out-of-distribution detection [39] and unsupervised active learning [40]. Similarly, leveraging information from the feature space via entropy maximization is a common approach in Reinforcement Learning for state exploration [41–44]. While our method relies on the same principles – acquiring more information about the feature space – our problem setting and methodology differ substantially, as we focus on Active Preference Modeling in the context of LLMs.

3 Preliminaries

Problem Statement. We formulate our setup as an *active inverse* variant of the contextual dueling bandit problem [45, 46]. We assume a prompt space $\mathcal{X}$, an action space $\mathcal{Y}$, and a language policy $\tau : \mathcal{X} \times \mathcal{Y} \to [0, \infty)$. Given $x \sim \mathcal{X}$, this language policy τ (e.g., an LLM) selects actions $y_1, y_2 \sim \tau(\cdot \mid x)$ (also referred to as completions), generating a dataset of tuples $\mathcal{D}_{pool} = \{x^i, y_1^i, y_2^i\}^N$. Crucially, x is sampled with replacement, i.e., we may generate multiple completions for the same prompt. Then, we define a policy $\pi : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$, which select tuples $(x, y_1, y_2) \in \mathcal{D}_{pool}$ to query for human binary preference over completions $y_1 \succ y_2$, forming a preference dataset $\mathcal{D}_{train} = \{x^i, y_1^i, y_2^i, y_1 \succ y_2{}^i\}^B$. Finally, D_{train} is used to learn a preference model $p_{\boldsymbol{\theta}}(y_1 \succ y_2 \mid x, y_1, y_2)$, parameterized by $\boldsymbol{\theta}$, which aims to recover the human preference function. The goal is to find π that minimizes the amount of samples B required to learn $p_{\boldsymbol{\theta}}(y_1 \succ y_2 \mid x, y_1, y_2)$.

 https://doi.org/10.52202/079017-3749

Preference Modeling. In this work, we assume that the preferences $y_1 \succ y_2$ are generated by an unknown latent reward model $r(x, y)$. We model $y_1 \succ y_2$ following the Bradley-Terry (BT) model [47]:

$$p(y_1 \succ y_2 \mid x, y_1, y_2) = \frac{\exp r(x, y_1)}{\exp r(x, y_1) + \exp r(x, y_2)}. \tag{1}$$

The BT model is often implemented by learning a parameterized latent reward model $r_{\boldsymbol{\theta}(x,y)}$ and optimizing $\boldsymbol{\theta}$ via maximum likelihood estimation. This means minimizing the negative Log-Likelihood with respect to the human preference labels.

Bayesian Active Learning. We adopt a Bayesian Model, which assumes a probability distribution over the parameters $\boldsymbol{\theta}$, such that, given a classification setting with inputs $x \sim X$ and labels $y \sim Y$ the predictive preference distribution is given by:

$$p(y \mid x) = \mathbb{E}_{p(\boldsymbol{\theta})}[p(y \mid x, \boldsymbol{\theta})]. \tag{2}$$

For active learning, we follow the methodology introduced by Lindley [23], namely BALD (Bayesian Active Learning by Disagreement), which proposes that the utility of a data point $x \sim X$ is given by the expected information gain about the parameters $\boldsymbol{\theta}$ with respect to the predictive distribution, a proxy of epistemic uncertainty:

$$U(x) \coloneqq \mathcal{I}(\boldsymbol{\theta}, y \mid \mathcal{D}_{train}, x) \quad = \mathcal{H}(p(y \mid \boldsymbol{x}, \mathcal{D}_{train})) - \mathbb{E}_{p(\boldsymbol{\theta} \mid \mathcal{D}_{train})}[\mathcal{H}(p(y \mid \boldsymbol{x}, \boldsymbol{\theta})]. \tag{3}$$

Kozachenko–Leonenko Entropy. The KL entropy estimator [48] is a non-parametric, particle-based estimator that leverages the k-nearest neighbors distance. Given a random variable X and a set of N i.i.d particles $\{x_i\}^N$, $x_i \sim X$, the KL entropy estimation for X is defined as:

$$\hat{\mathcal{H}}_{KL}(X) = \frac{d_X}{N} \sum_{i=0}^{N} \log D_x(i) + \log v_{d_X} + \psi(N) - \psi(k), \tag{4}$$

where d_X is the dimension of X, v_{d_X} is the volume of the d_X-dimensional unit ball, ψ is the digamma function, and $D_x(i)$ is twice the distance between the particle x_i to its k-nearest neighbor.

4 Bayesian Active Learner for Preference Modeling

We now introduce our method for Active Preference Modeling, BAL-PM, illustrated in Figure 2. Our desiderata is to design an acquisition policy that addresses the shortcomings of naive epistemic uncertainty estimation – such as the acquisition of redundant samples – by leveraging an unsupervised source of epistemic uncertainty that encourages diversity in the acquired training distribution.

Objective. Based on the above, we propose the following objective:

$$\pi = \underset{(x,y_1,y_2) \in \mathcal{D}_{pool}}{\arg\max} \hat{U}(x, y_1, y_2) + \beta \hat{\mathcal{H}}(X_{tr} \cup \{x\}), \tag{5}$$

where $\hat{U}(x, y_1, y_2)$ is the preference model epistemic uncertainty estimate for the tuple (x, y_1, y_2) and $\hat{\mathcal{H}}(X_{tr} \cup \{x\})$ is the entropy estimate *for the acquired prompt distribution*, assuming the policy selects x. X_{tr} is a slight abuse of the notation that refers to the set of prompts in the previously acquired training set. Lastly, β is a hyperparameter to balance the contribution of each term. Crucially, the first term represents a *task-dependent source* of epistemic uncertainty, since it refers to the learned preference Model. In contrast, the second term represents a *task-agnostic source*, as it solely relies on the information available in the feature space spanned by the base LLM.

Preference Model Epistemic Uncertainty Estimation. We first describe our Bayesian Preference Model. Assuming a prior distribution over parameters $p(\boldsymbol{\theta})$, the posterior predictive distribution over preferences after observing $\mathcal{D}_{train}$ is given by:

$$p(y_1 \succ y_2 \mid x, y_1, y_2, \mathcal{D}_{train}) = \int p(y_1 \succ y_2 \mid x, y_1, y_2, \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathcal{D}_{train}) d\boldsymbol{\theta}, \tag{6}$$

where the likelihood term $p(y_1 \succ y_2 \mid x, y_1, y_2, \boldsymbol{\theta})$ follows the BT model in Equation 1. Considering deep models, solving this inference problem is intractable, given the large parameter space. Nonetheless, we may assume a simple yet effective posterior approximation via deep ensembles [49–51]:

$$p(\boldsymbol{\theta} \mid \mathcal{D}_{train}) \approx \sum_{k=0}^{K} \delta(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_k). \tag{7}$$

Equation 8 approximates the posterior distribution over parameters $p(\boldsymbol{\theta} \mid \mathcal{D}_{train})$ as a mixture of delta functions, where K is the number of ensemble models and $\hat{\boldsymbol{\theta}}_k$ is the MAP estimate of model k. The posterior predictive distribution is then computed via the following approximation:

$$p(y_1 \succ y_2 \mid x, y_1, y_2, \mathcal{D}_{train}) \approx \frac{1}{K} \sum_{k=0}^{K} p(y_1 \succ y_2 \mid x, y_1, y_2, \boldsymbol{\theta}_k), \boldsymbol{\theta}_k \sim p(\boldsymbol{\theta} \mid \mathcal{D}_{train}). \tag{8}$$

Equations 7 and 8 allow approximate inference by training multiple preference models separately. However, this is challenging in the context of LLMs, as fine-tuning billions of parameters several times is computationally expensive and impractical in many settings. Alternatively, we employ an ensemble of adapters [52, 53], which consists of multiple lightweight networks (with a few million parameters each) on top of the frozen LLM that works as a feature extractor. This allows us to generate the LLM features offline and use them as a dataset, considerably reducing the resources required for training and Bayesian inference. This also enables using very large base models, with dozens or hundreds of billions of parameters in a single GPU setting. Finally, based on the previous modeling assumptions, we can estimate the epistemic uncertainty term employing Equation 3:

$$\hat{U}(x, y_1, y_2) = \mathcal{H}(\frac{1}{K} \sum_{k=0}^{K} p(y_1 \succ y_2 \mid x, y_1, y_2, \boldsymbol{\theta}_k)) - \frac{1}{K} \sum_{k=0}^{K} \mathcal{H}(p(y_1 \succ y_2 \mid x, y_1, y_2, \boldsymbol{\theta}_k)). \tag{9}$$

Feature Space Entropy Estimation. Equation 5 requires estimating the entropy of the acquired prompt distribution, $\mathcal{H}(X_{train})$. For this matter, we employ a kNN-based entropy estimator. We represent each prompt in the pool as the last-layer embedding vector generated by the base LLM, leveraging the semantic representations learned during unsupervised pre-training.

However, naively applying the KL estimator from Equation 4 has a major drawback: the training set $\mathcal{D}_{train}$ initially contains very few data points and does not provide support to represent the probability density, introducing bias to the estimates and affecting the data selection.

For illustration, we show the scenario of Figure 3a. In this case, we estimate the entropy using Equation 4, with $k = 3$. Since it does not account for the available points in the pool, it underestimates the density around the top cluster and ends up selecting the green point as the one that maximizes the entropy of the feature space, while the point that does so is in the bottom cluster. In an extreme case where all the points in

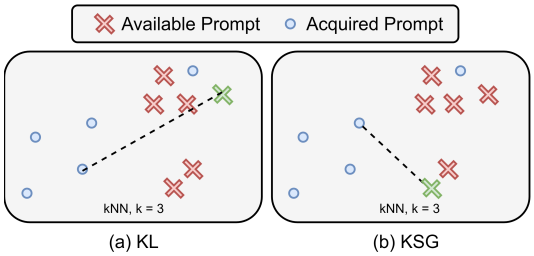


Figure 3: **Illustration of entropy estimators.** The green point maximizes the entropy estimation of the prompt distribution (according to the employed estimator). Dashed lines represent its k-NN distance. In (a), the KL estimator (Equation 4) does not account for the available prompts in the pool (in red) and underestimates the density in regions not covered by the acquired set (in blue). In (b), the KSG estimator (Equation 10) uses all data points, leading to better estimation and effectively selecting the point that maximizes the true entropy.

https://doi.org/10.52202/079017-3749

Algorithm 1 BAL-PM

Require: Pool set $\mathcal{D}_{pool} = \{x^i, y_1^i, y_2^i\}^N$, training set $\mathcal{D}_{train} = \{x^i, y_1^i, y_2^i, y_1 \succ y_2{}^i\}^B$
Require: Base LLM τ, Bayesian Preference Model $p(y_1 \succ y_2 \mid x, y_1, y_2)$
  Compute feature sets for $\mathcal{D}_{pool}$ and $\mathcal{D}_{train}$ by performing forward passes on τ
  Compute kNN distances for points in $\mathcal{D}_{pool} \cup \mathcal{D}_{train}$
  while true **do**
    Train Bayesian Preference Model (ensemble) in $\mathcal{D}_{train}$ assuming Equations 7 and 8
    Compute Epistemic Uncertainty Estimates $\hat{U}(x, y_1, y_2)$ via Equation 9
    Initialize $n_{X_{tr}}(x)$ by counting $\{u \mid u \in \mathcal{D}_{train} \wedge (\|x - u\| \leq D(x)/2\}, \forall x \in \mathcal{D}_{pool}$
    Initialize Batch: $\mathcal{B} = \{\}$
    while batch not full **do**
      Compute entropy term: $e(x) = \log D(x) - \frac{1}{d_X}\psi(n_{X_{tr}}(x) + 1)$
      Select tuple (x^*, y_1^*, y_2^*) following $\pi = \underset{(x,y_1,y_2)\in\mathcal{D}_{pool}}{\arg\max} \hat{U}(x, y_1, y_2) + \beta e(x)$
      Update Pool and Batch: $\mathcal{D}_{pool} = \mathcal{D}_{pool} \backslash (x^*, y_1^*, y_2^*)$, $\mathcal{B} = \mathcal{B} \cup (x^*, y_1^*, y_2^*)$
      Update counts: $\forall x \in \mathcal{D}_{pool}$, $n_{X_{tr}}(x) = n_{X_{tr}}(x) + 1$ if $\|x - x^*\| \leq D(x)/2$
    end while
    Collect human feedback for $\mathcal{B}$ and update training set $\mathcal{D}_{train} = \mathcal{D}_{train} \cup \mathcal{B}$
  end while

the top cluster are the same, this bias leads to acquiring duplicated points. In Appendix E we formally derive the KL entropy estimator and show how the low-data regime challenges its main assumptions.

Alternatively, we may use the available unlabeled pool, often much larger than the acquired set. Following the argument introduced by Kraskov et al. [54], the key insight is to notice that Equation 4 holds for *any* value of k and it does not require a fixed k over different particles for entropy estimation (we provide more details in Appendix E). Therefore, we can find the distance to the k-th nearest neighbor in the joint space spanned by the pool and the acquired set and map it to the corresponding neighbor (denoted as $n_{X_{tr}}$) in X_{train} to estimate the marginal entropy. This results in the KSG marginal entropy estimator [54], but repurposed to our setting:

$$\hat{\mathcal{H}}_{KSG}(X) = \frac{d_X}{N}\sum_{i=0}^{N} \log D_x(i) + \log v_{d_X} + \psi(N) - \frac{1}{N}\sum_{i=0}^{N} \psi(n_{X_{tr}}(i) + 1), \tag{10}$$

where $D_x(i)$ is now computed in the joint space and $n_{X_{tr}}(i)$ is the number of points in $\mathcal{D}_{train}$ whose distance to x_i is less than $D_x(i)/2$. Figure 3 (b) illustrates the data selection by following this alternative estimation, leading to more diversity in the feature space.

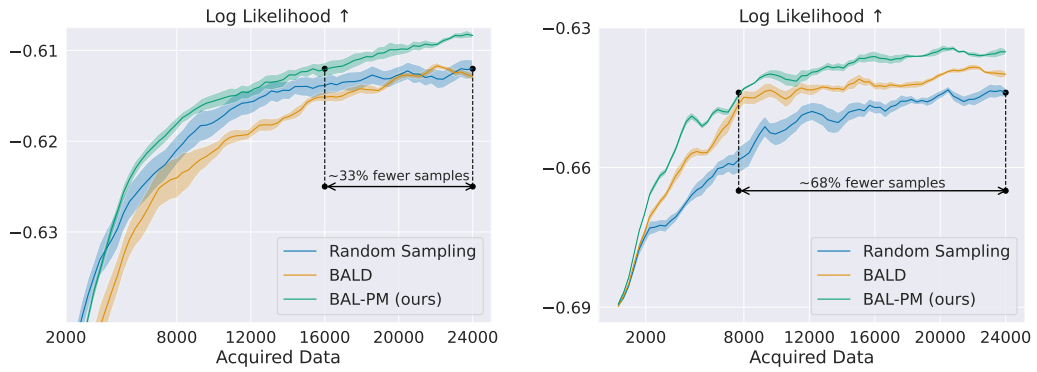
Implementation. Firstly, we simplify the entropy term by dropping the constant terms with respect to x:

$$\underset{(x,y_1,y_2)\in\mathcal{D}_{pool}}{\arg\max} \hat{\mathcal{H}}(X_t \cup \{x\}) = \underset{(x,y_1,y_2)\in\mathcal{D}_{pool}}{\arg\max} \log D(x) - \frac{1}{d_X}\psi(n_{X_{tr}}(x) + 1). \tag{11}$$

Equation 11 acquire points by computing $D(x)$ (based on the kNN distance) and the counter $n_{X_{tr}}$ related to prompt x only. Furthermore, as $D(x)$ accounts for the full dataset in the KSG estimator, it does not change over training. Hence, we may compute it offline once, and potentially scale to very large datasets [55]. Lastly, BAL-PM acquisition scheme builds a batch of data by successively selecting points following Equation 5. Crucially, while BAL-PM keeps the preference model uncertainty estimates the same over the batch, it updates the entropy term after in-batch iteration. This operation boils down to updating the counter $n_{X_{tr}}$, a lightweight operation. In Algorithm 1, we present the pseudocode for BAL-PM. Appendix H further describes its computational cost.

5 Experiments and Discussion

In this Section, we aim to evaluate how BAL-PM performs in Active Preference Modeling. Our central hypothesis is that leveraging the information available on the feature space spanned by the



(a) Reddit TL;DR (Test)

(b) CNN/DM Dataset (OOD)

Figure 4: **Comparison with baseline methods in Active Preference Modeling**. BAL-PM considerably reduces the number of samples required for preference modeling, achieving **33%** and **68%** of reduction in the Reddit TL;DR test split and CNN/DM News datasets, respectively. The shaded area corresponds to the standard error computed over five seeds.

base LLM — a task-agnostic source of epistemic uncertainty – addresses the problem of acquiring redundant samples, a natural pathology of relying on task-dependent epistemic uncertainty estimators designed for single-point acquisition schemes. BAL-PM, our proposed stochastic acquisition policy, promotes this diversity by maximizing the entropy of the acquired prompt distribution, besides selecting points for which the preference model presents high epistemic uncertainty.

Experimental Setup. We consider a pool-based active learning setup. Each experiment consists of several acquisition cycles, where each iteration performs a batch acquisition in the currently available pool. The training set starts with the size of one acquired batch and leaves the remaining data for the pool set. Following previous works [13, 14], we reinitialize the model after each acquisition to decorrelate subsequent acquisitions. We train the ensemble of adapters on previously acquired data and employ early stopping based on the Log-Likelihood of a held-out validation set. We evaluate the preference model after each acquisition loop and report the average Log-Likelihood of the test sets. Appendix G discusses why the test average Log-Likelihood is a proper metric for Preference Modeling. Finally, Appendix C details hyperparameters and tuning methodology used in this work[4].

Model Architecture. As described in Section 4, we employ an ensemble of adapters on top of a base LLM. Each adapter is a multi-layer perceptron with non-linear activations. In most experiments, the base LLM is a 7-billion parameter model, although we also employed 70-billion and 140-billion parameter ones when analyzing the effect of scaling the base LLM. All considered models are only unsupervised pre-trained and have not undergone any preference fine-tuning.

Datasets. Following previous work [20], we considered prompts from the Reddit TL;DR dataset of Reddit posts [25] and the CNN/DM News dataset [26]. We leverage the generated completions and human feedback collected by Stiennon et al. [1]. The Reddit dataset contains train/eval/test splits, and we adopt the train split (92,858 points) for the pool and training sets, the eval split (33,083 points) for validation, and report results in the test set (50,719 points). The CNN/DM dataset contains a single split (2,284 points), and we use it for the Out-Of-Distribution (OOD) evaluation.

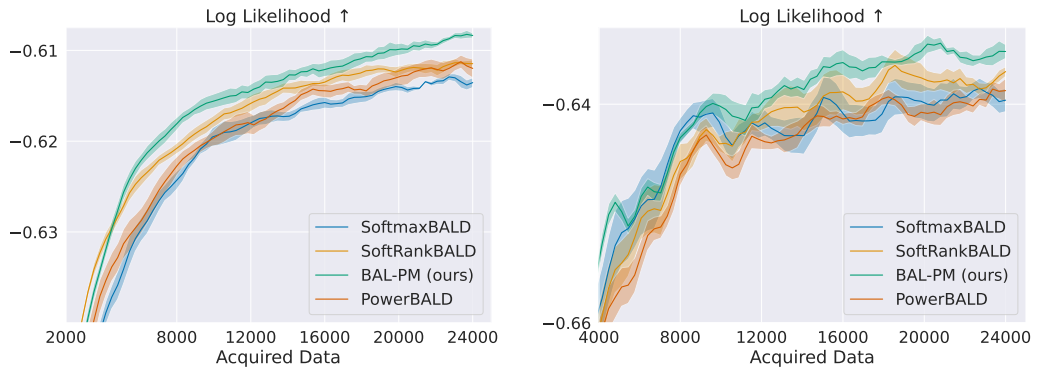
Comparison Methods. We considered **Random Sampling** and **BALD** [23] as baselines. BALD selects points based on the utility function of Equation 3 and is equivalent to the acquisition function used by Gleave and Irving [20]. We also compared BAL-PM with other stochastic acquisition policies [22], namely **SoftmaxBALD**, **SoftRankBALD**, and **PowerBALD**. We refer to Kirsch et al. [22] for a detailed description of these methods.

5.1 Experiments

We highlight and analyze the following questions to evaluate our hypothesis and proposed method.

[4]We release our code at `https://github.com/luckeciano/BAL-PM`.

https://doi.org/10.52202/079017-3749



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset (OOD)

Figure 5: **Comparison with Bayesian stochastic acquisition policies for Active Preference Modeling**. BAL-PM consistently outperforms other policies in Test and OOD settings.

Does BAL-PM reduce the volume of feedback required for Preference Modeling? We start evaluating how BAL-PM performs against standard random acquisition and BALD, as presented in Figure 4. BAL-PM considerably reduces the volume of data required to learn the preference model. Particularly compared with random sampling, it reduces the number of required samples in **33%** for the Reddit TL;DR dataset and **68%** for the out-of-distribution setting of CNN/DM News dataset, representing a substantial reduction in the human feedback needed. BALD does not present any benefits over random sampling in the TL;DR dataset, which aligns with previous work [20]. Interestingly, BALD also presents an interesting improvement over random sampling in the OOD setting, but BAL-PM consistently outperforms BALD with more data.

How does BAL-PM compare with other stochastic acquisition policies? Next, we analyze BAL-PM in comparison with other Bayesian stochastic acquisition policies. These policies address the acquisition of redundant samples by relying on sampling points from the distribution determined by the task-dependent epistemic uncertainty scores. BAL-PM consistently surpasses all variations in both datasets, suggesting that leveraging the information available in the prompt feature space – as a task-agnostic source of epistemic uncertainty – is more effective in encouraging diversity for batch acquisition in the considered setting.

Does BAL-PM encourage diversity and prevent the acquisition of redundant samples? We evaluate the exploration approach of the considered methods by analyzing the statistics of the acquired prompt distribution, particularly the number of unique prompts over the course of training.

Figure 6 presents three different perspectives on the acquired distribution. On the left, it presents the number of unique acquired prompts over learning, which indicates diversity in the training set. BAL-PM selects new prompts at a much faster rate than random sampling and BALD. Naturally, this rate saturates when the selection exhausts the number of distinct prompts available in the pool (approximately 14,000). The rate is also not equivalent to the data acquisition rate since BAL-PM does not simply select different prompts but also prioritizes points with high epistemic uncertainty.

The middle plot shows the ratio of unique prompts in each active learning loop, and BAL-PM acquires batches with all distinct prompts during almost the whole training. BALD only maintains a rate of 70%, which means a substantial number of duplicated prompts. In Appendix K, we present the first batch sampled by BALD and BAL-PM for a qualitative analysis. Lastly, the plot on the right shows the ratio of unique prompts across all training. While random sampling presents a high unique prompt ratio in each separate batch, it consistently samples duplicated prompts throughout learning. In contrast, BAL-PM maintains a high ratio of unique prompts during most of the training. Again, this rate progressively decays as BAL-PM exhausts the pool of different prompts and due to the influence of the epistemic uncertainty prioritizing particular prompt-completion pairs.

How does BAL-PM scale to larger LLMs? As highlighted in Section 4 our design choices allow us to scale our experiment for very large base LLMs in a single GPU setting. We investigate the effect of scaling the base LLM in BAL-PM performance, considering 70-billion and 140-billion parameter models in their 4-bit quantized versions. Naturally, the preference model performance improves substantially against the 7-billion parameter model. More interestingly, BAL-PM presents similar

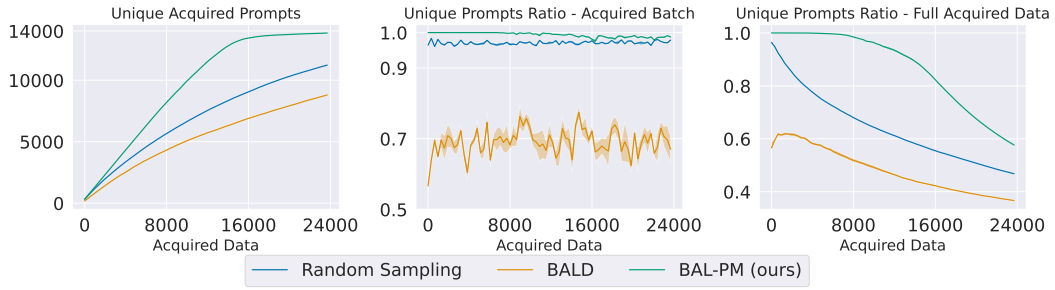
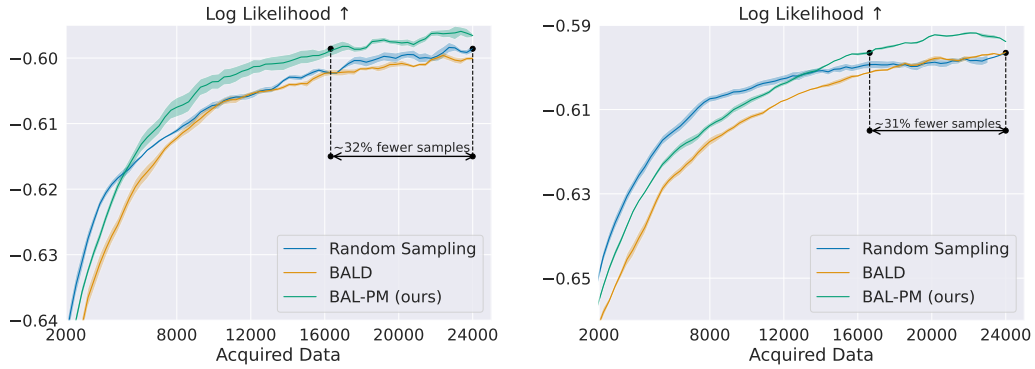


Figure 6: **Statistics of acquired prompt distribution**. We present the total number of unique acquired prompts (left), the ratio of unique acquired prompts per active learning loop (middle), and the ratio of unique acquired prompts over training. BAL-PM consistently acquires novel prompts and encourages diversity in each acquired batch and the full training set.



(a) 70b Parameter Model (b) 140b Parameter Model

Figure 7: **The effect of scaling the base LLM**. We analyzed how increasing the size of the base LLM affects BAL-PM performance in the Reddit TL;DR dataset. We considered (a) a 70-billion parameter model and (b) a 140-billion parameter model. Interestingly, we find approximately the same gains (31%–33% reduction of required samples) across all models.

gains across all scales, with around 31%–33% reduction of required samples compared to random sampling. In contrast, BALD still does not present benefits over random sampling, suggesting that the scale of the base LLM is not the prevailing factor for its negative result.

Ablations and Further Analysis. We conduct ablation studies in the key components of the proposed method in Appendix D. More concretely, we ablate the components of the objective to show that both preference model epistemic uncertainty and entropy scores play a relevant role in BAL-PM. We also ablate the type of uncertainty and the employed entropy estimator. Furthermore, we conduct further empirical analysis in Appendix F to investigate how each component of Equation 5 contributes to the data selection, and conduct a robustness analysis for the β hyperparameter in Appendix I. Lastly, we provide comparison with additional data selection baselines in Appendix J.

6 Closing Remarks

In this work, we present BAL-PM, a Bayesian Active Learning method for Preference Modeling in Language Models. BAL-PM is a stochastic acquisition policy that selects points for which the preference model presents high epistemic uncertainty and also maximizes the entropy of the acquired prompt distribution. We show that leveraging the information available on the feature space spanned by the base LLM via this entropy term has a crucial role in preventing the acquisition of redundant samples. BAL-PM substantially reduces the volume of feedback required for Preference Modeling and outperforms existing Bayesian stochastic acquisition policies. It also scales for very large LLMs and effectively balances the contribution of both considered sources of uncertainty.

https://doi.org/10.52202/079017-3749

Limitations. Despite its encouraging results, BAL-PM presents some limitations. For instance, it heavily relies on the quality of the feature representations provided by the base LLM. Particularly, it might be subject to the Noisy-TV problem [56] and provide high-entropy scores to nonsensical prompts if those are spread in the representation space rather than collapsed into a single region. Fortunately, we expect this limitation to be progressively addressed by better LLMs.

Future Work may evaluate BAL-PM in larger preference datasets with millions or billions of data points. Another direction analyzes how the learned models perform in the Preference Optimization setting. Lastly, future work may extend BAL-PM to consider recent prediction-oriented methods of epistemic uncertainty estimation [57] in contrast to parameter-based methods such as BALD.

Acknowledgments and Disclosure of Funding

We thank Jannik Kossen for the insightful discussions in the early stages of this project. We also thank the reviewers for providing insightful feedback. Luckeciano C. Melo acknowledges funding from the Air Force Office of Scientific Research (AFOSR) European Office of Aerospace Research & Development (EOARD) under grant number FA8655-21-1-7017.

References

[1] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1f89885d556929e98d3ef9b86448f951-Paper.pdf.

[2] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

[3] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL https://arxiv.org/abs/2204.05862.

[4] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[5] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

[6] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019. URL https://arxiv.org/abs/1909.08593.

[7] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphael Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=bx24KpJ4Eb. Survey Certification.

[8] Michael Bailey, Saeed Moayedpour, Ruijiang Li, Alejandro Corrochano-Navarro, Alexander Kötter, Lorenzo Kogler-Anele, Saleh Riahi, Christoph Grebner, Gerhard Hessler, Hans Matter, Marc Bianciotto, Pablo Mas, Ziv Bar-Joseph, and Sven Jager. Deep batch active learning for drug discovery. July 2023. doi: 10.1101/2023.07.26.550653. URL http://dx.doi.org/10.1101/2023.07.26.550653.

[9] Steven C. H. Hoi, Rong Jin, Jianke Zhu, and Michael R. Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 417–424, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143897. URL https://doi.org/10.1145/1143844.1143897.

[10] Meta Llama Team. Introducing Meta Llama 3: The most capable openly available LLM to date — ai.meta.com. https://ai.meta.com/blog/meta-llama-3/. [Accessed 13-05-2024].

[11] Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 30039–30069. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/5fc47800ee5b30b8777fdd30abcaaf3b-Paper-Conference.pdf.

[12] Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dcjtMYkpXx.

[13] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/95323660ed2124450caaac2c46b5ed90-Paper.pdf.

[14] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1183–1192. JMLR.org, 2017.

[15] Andreas Kirsch. *Advanced deep active learning and data subset selection: unifying principles with information-theory intuitions*. PhD thesis, 2023. URL https://ora.ox.ac.uk/objects/uuid:3799959f-1f39-4ae5-8254-9d7e54810099.

[16] David J. C. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992. doi: 10.1162/neco.1992.4.4.590.

[17] Aditya Siddhant and Zachary C. Lipton. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2904–2909, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1318. URL https://aclanthology.org/D18-1318.

[18] Biraja Ghoshal, Stephen Swift, and Allan Tucker. Bayesian deep active learning for medical image analysis. In Allan Tucker, Pedro Henriques Abreu, Jaime Cardoso, Pedro Pereira Rodrigues, and David Riaño, editors, *Artificial Intelligence in Medicine*, pages 36–42, Cham, 2021. Springer International Publishing. ISBN 978-3-030-77211-6.

[19] Jianxiang Feng, Jongseok Lee, Maximilian Durner, and Rudolph Triebel. Bayesian active learning for sim-to-real robotic perception. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10820–10827, 2022. doi: 10.1109/IROS47612.2022.9982175.

[20] Adam Gleave and Geoffrey Irving. Uncertainty estimation for language reward models, 2022. URL https://arxiv.org/abs/2203.07472.

 https://doi.org/10.52202/079017-3749

[21] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

[22] Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frédéric Branchaud-Charron, and Yarin Gal. Stochastic batch acquisition: A simple baseline for deep active learning. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=vcHwQyNBjW. Expert Certification.

[23] D. V. Lindley. On a Measure of the Information Provided by an Experiment. *The Annals of Mathematical Statistics*, 27(4):986 – 1005, 1956. doi: 10.1214/aoms/1177728069. URL https://doi.org/10.1214/aoms/1177728069.

[24] Jishnu Mukhoti, Andreas Kirsch, Joost R. van Amersfoort, Philip H. S. Torr, and Yarin Gal. Deep deterministic uncertainty: A new simple baseline. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24384–24394, 2021. URL https://api.semanticscholar.org/CorpusID:246411740.

[25] Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In Lu Wang, Jackie Chi Kit Cheung, Giuseppe Carenini, and Fei Liu, editors, *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4508. URL https://aclanthology.org/W17-4508.

[26] Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1693–1701, Cambridge, MA, USA, 2015. MIT Press.

[27] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning, 2011.

[28] A. Gilad Kusne, Heshan Yu, Changming Wu, Huairuo Zhang, Jason Hattrick-Simpers, Brian DeCost, Suchismita Sarker, Corey Oses, Cormac Toher, Stefano Curtarolo, Albert V. Davydov, Ritesh Agarwal, Leonid A. Bendersky, Mo Li, Apurva Mehta, and Ichiro Takeuchi. On-the-fly closed-loop autonomous materials discovery via bayesian active learning. *Nature Communications*, 11(5966), 2020.

[29] Christopher Tosh, Mauricio Tec, Jessica White, Jeffrey F. Quinn, Glorymar Ibanez Sanchez, Paul Calder, Andrew L. Kung, Filemon S. Dela Cruz, and Wesley Tansey. A bayesian active learning platform for scalable combination drug screens. *bioRxiv*, 2023. doi: 10.1101/2023.12.18.572245. URL https://www.biorxiv.org/content/early/2023/12/20/2023.12.18.572245.

[30] Johannes Fürnkranz and Eyke Hüllermeier. Pairwise preference learning and ranking. In *European Conference on Machine Learning*, 2003. URL https://api.semanticscholar.org/CorpusID:4735672.

[31] Wei Chu and Zoubin Ghahramani. Preference learning with gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 137–144, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102369. URL https://doi.org/10.1145/1102351.1102369.

[32] Viraj Mehta, Vikramjeet Das, Ojash Neopane, Yijia Dai, Ilija Bogunovic, Jeff Schneider, and Willie Neiswanger. Sample efficient reinforcement learning from human feedback via active exploration, 2024. URL https://openreview.net/forum?id=2RJAzSphy9.

[33] Kaixuan Ji, Jiafan He, and Quanquan Gu. Reinforcement learning from human feedback with active queries, 2024.

[34] Anonymous. DUO: Diverse, uncertainty-aware, on-policy query generation and selection for reinforcement learning from human feedback, 2024. URL https://openreview.net/forum?id=gsMtrVUF0q.

[35] Nirjhar Das, Souradip Chakraborty, Aldo Pacchiano, and Sayak Ray Chowdhury. Provably sample efficient rlhf via active preference optimization, 2024.

[36] Vikranth Reddy Dwaracherla, Seyed Mohammad Asghari, Botao Hao, and Benjamin Van Roy. Efficient exploration for llms. *ArXiv*, abs/2402.00396, 2024. URL https://api.semanticscholar.org/CorpusID:267364948.

[37] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, MORTEZA IBRAHIMI, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 2795–2823. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/07fbde96bee50f4e09303fd4f877c2f3-Paper-Conference.pdf.

[38] Janis Postels, Hermann Blum, César Cadena, Roland Y. Siegwart, Luc Van Gool, and Federico Tombari. Quantifying aleatoric and epistemic uncertainty using density estimation in latent space. *ArXiv*, abs/2012.03082, 2020. URL https://api.semanticscholar.org/CorpusID:227335360.

[39] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f5496252609c43eb8a3d147ab9b9c006-Paper.pdf.

[40] Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9690–9700. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/van-amersfoort20a.html.

[41] Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2681–2691. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/hazan19a.html.

[42] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

[43] Dongyoung Kim, Jinwoo Shin, Pieter Abbeel, and Younggyo Seo. Accelerating reinforcement learning with value-conditional state entropy exploration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=97E3YXvcFM.

[44] Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 18459–18473. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/99bf3d153d4bf67d640051a1af322505-Paper.pdf.

[45] Aadirupa Saha. Optimal algorithms for stochastic contextual preference bandits. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 30050–30062. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/fc3cf452d3da8402bebb765225ce8c0e-Paper.pdf.

[46] Siddartha Y. Ramamohan, Arun Rajkumar, Shivani Agarwal, and Shivani Agarwal. Dueling bandits: Beyond condorcet winners to general tournament solutions. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/fccb3cdc9acc14a6e70a12f74560c026-Paper.pdf.

[47] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444. URL http://www.jstor.org/stable/2334029.

[48] L. F. Kozachenko and N. Leonenko. Sample estimate of the entropy of a random vector. *Problemy Peredachi Inform*, 1987. URL ^1^.

[49] Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023. URL http://probml.github.io/book2.

 https://doi.org/10.52202/079017-3749

[50] Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4697–4708. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/322f62469c5e3c7dc3e58f5a4d1ea399-Paper.pdf.

[51] Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4629–4640. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/izmailov21a.html.

[52] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/e7b24b112a44fdd9ee93bdf998c6ca0e-Paper.pdf.

[53] Sylvestre-Alvise Rebuffi, Andrea Vedaldi, and Hakan Bilen. Efficient parametrization of multi-domain deep neural networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8119–8127, 2018. doi: 10.1109/CVPR.2018.00847.

[54] Alexander Kraskov, Harald Stoegbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.

[55] Qi Chen, Bing Zhao, Haidong Wang, Mingqin Li, Chuanjie Liu, Zengzhong Li, Mao Yang, and Jingdong Wang. Spann: Highly-efficient billion-scale approximate nearest neighborhood search. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 5199–5212. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/299dc35e747eb77177d9cea10a802da2-Paper.pdf.

[56] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1lJJnR5Ym.

[57] Freddie Bickford Smith, Andreas Kirsch, Sebastian Farquhar, Yarin Gal, Adam Foster, and Tom Rainforth. Prediction-oriented bayesian active learning. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 7331–7348. PMLR, 25–27 Apr 2023. URL https://proceedings.mlr.press/v206/bickfordsmith23a.html.

[58] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

[59] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020.

[60] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In Bruce W. Croft and C. J. van Rijsbergen, editors, *SIGIR '94*, pages 3–12, London, 1994. Springer London. ISBN 978-1-4471-2099-5.

[61] Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5753–5763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/kossen21a.html.

[62] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[63] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

[64] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 10835–10866. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

[65] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

[66] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In Eneko Agirre, Marianna Apidianaki, and Ivan Vulić, editors, *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.deelio-1.10. URL https://aclanthology.org/2022.deelio-1.10.

[67] Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. Demystifying prompts in language models via perplexity estimation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.679. URL https://aclanthology.org/2023.findings-emnlp.679.

 https://doi.org/10.52202/079017-3749

A Impact Statement

Preference fine-tuning has become a crucial step in aligning LLMs toward human preferences and demonstrated a real-world impact in many open-source and production systems [2, 10]. Nonetheless, collecting human feedback is very expensive and time-consuming, posing a substantial bottleneck for further development of LLMs. In this work, we approach the problem of Active Preference Modeling, which aims to reduce the volume of feedback required for learning preferences. We show that our proposed method, BAL-PM, requires 33% to 68% fewer labels in the human preference datasets considered. We believe that these results point out to **a strong impact in the process of acquiring labels**, and estimate an **economy of hundreds of thousands of dollars and months of labeling work in the current scale of LLMs**. This scenario represents faster cycles of preference optimization, potentially leading to better-aligned and safer models. Therefore, we believe our work poses a relevant positive societal impact for the upcoming years.

B Reproducibility Statement

Code Release. To ensure the reproducibility of our research findings, we release our code at `https://github.com/luckeciano/BAL-PM`. Our implementation is based on PyTorch [58] and HuggingFace [59]. All baselines are available in the released code.

Experiments Reproducibility. We detail our methodology in Section 4 and our experimental setup in Section 5. We provide all hyperparameters used in this work as well as the selection strategy in Appendix C. We plan to release all the raw experiment logs and feature datasets generated in this work. For all experiments in this paper, we report the results over five seeds with standard errors. For better visualization, we applied smoothing for the curves considering two past observations.

Datasets. All preference datasets are open-source and available online for academic use [1].

Compute Resources. We execute all active learning experiments in a single A100 GPU, and each experiment takes approximately one day. For the base LLM feature generation, we also use a single A100 GPU, taking a few hours for the 7-billion parameter model and approximately four days for the 70-billion and 140-billion parameter models.

C Hyperparameters

In Table 2, we share all hyperparameters used in this work. We specifically performed a hyperparameter search on the entropy term parameters and baselines. The search strategy was a simple linear search on the options in Table 1, considering each parameter in isolation. The selection followed the final performance on a held-out validation set. For baselines, we mostly considered the values presented in prior work [22]. For the proposed method, we also considered d_X as a hyperparameter and found that smaller values often work better than using the dimensionality of the base LLM embeddings.

Hyperparameter	Value
Acquisition Batch Size	320
Initial Training Set Size	320
Initial Pool Size	92,000
Active Learning Iterations	75
Adapter Network Hidden Layers	[2048, 256]
Adapter Network Activation Function	$\tanh$
7b Model Name	OpenHermes-2.5-Mistral-7B
70b Model Name	Llama3-70b
140b Model Name	Mixtral-8x22B-v0.1
Early Stopping Patience	3
Training Batch Size	32
Learning Rate	3e-5
Learning Rate Scheduler	Cosine
Optimizer	AdamW
Entropy Term β	0.01
Entropy Term k	13
Entropy Term d_X	1.0
SoftmaxBALD β	10,000
SoftRankBALD β	1.0
PowerBALD β	8.0

Table 1: **Training Hyperparameters**.

Hyperparameter	Search Space
Entropy Term β	[0.0001, 0.001, 0.01, 0.1, 1.0]
Entropy Term k	[1, 7, 13, 19, 25]
Entropy Term d_X	[4096, 2048, 1024, 256, 64, 32, 8, 4, 2, 1, 0.5]
SoftmaxBALD β	[0.25, 1.0, 2.0, 100.0, 1000.0, 5000.0, 10,000]
SoftRankBALD β	[0.25, 1.0, 2.0, 4.0, 8.0]
PowerBALD β	0.25, 1.0, 2.0, 4.0, 8.0, 10.0, 12.0

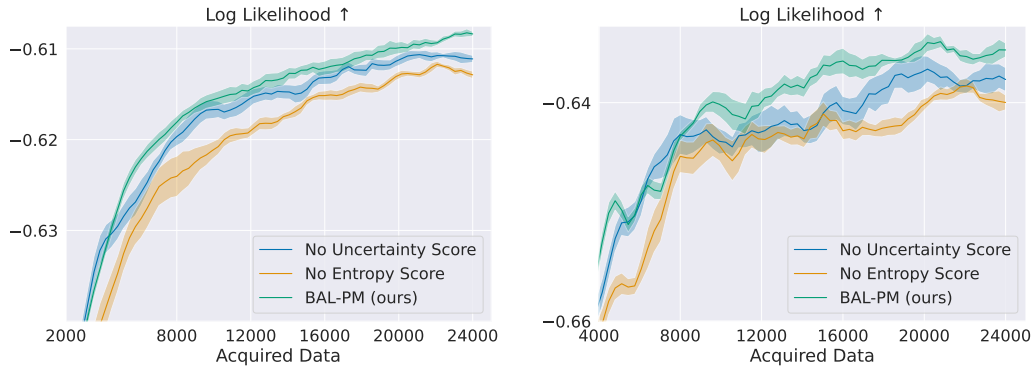
Table 2: **Hyperparameters search space**.

https://doi.org/10.52202/079017-3749

D Ablation Studies

This Section presents and discusses the results of the ablation studies. We focused on three different aspects: the components in the objective of Equation 5; the nature of the uncertainty considered; and the entropy estimator.

Objective Components. We considered three different versions for ablating components: BAL-PM (ours), which follows Equation 5 exactly; a version with **No Uncertainty Score** in the objective; and another version with **No Entropy Score**. Figure 8 shows the findings. In the datasets considered, both terms of the objective play a crucial role in the performance of BAL-PM. Disregarding the entropy score fundamentally means solely following BALD, which acquires several redundant samples. On the other side, disregarding the uncertainty score prevents the learner from acquiring points where the model lacks information.



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset

Figure 8: **The ablation study of the components in the BAL-PM objective.** We considered BAL-PM, a version without the uncertainty score, and a version without the entropy score.

Type of Uncertainty. In Machine Learning, we identify two different sources of uncertainty: *epistemic* and *aleatoric*. Epistemic Uncertainty refers to the uncertainty in the parameters of the model, often due to the lack of information from some regions of the parameter space. In contrast, aleatoric uncertainty refers to the uncertainty in the *data*, originating from the inherent noise of the data generation process. We reduce epistemic uncertainty by acquiring new data, while aleatoric is irreducible.

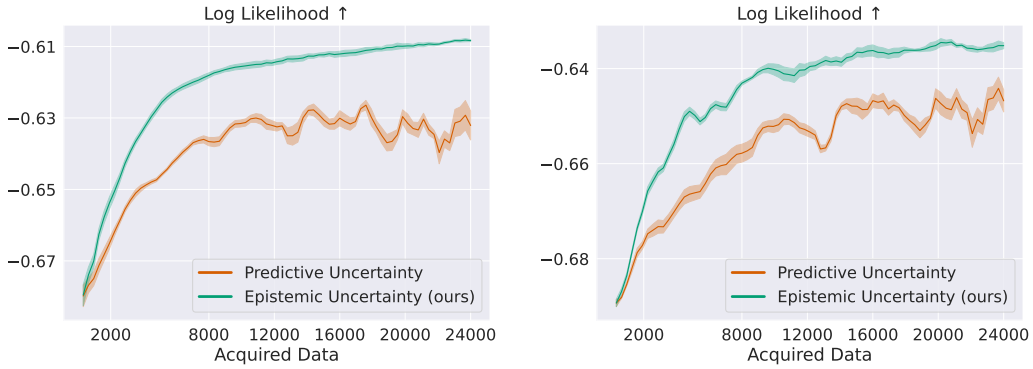
A common practice in Active Learning is to select points based on high *Predictive Uncertainty*, which is often referred to as "Uncertainty Sampling" [60]. This type represents the total uncertainty, i.e., it accounts for both epistemic and aleatoric sources. Therefore, we expect that following Predictive Uncertainty underperforms in datasets with high label noise, as the objective may favor points with high aleatoric uncertainty and low epistemic uncertainty.

Figure 9 compares using **Predictive** and **Epistemic** uncertainties in the objective of Equation 5. Selecting points based on epistemic uncertainty strongly outperforms the other variant, which aligns with the fact that preference datasets contain high levels of label noise – as mentioned in Section 1, the agreement rate among labelers is low, typically between 60% and 75%. This ablation also highlights the importance of a Bayesian preference model for epistemic uncertainty estimation.

Entropy Estimator. In Section 5, we argue for using the KSG entropy estimator (rather than the KL estimator) since it leverages the full dataset and better estimates the probability density in the feature space, leading to less biased entropy estimates. In this ablation, we compare both estimators to measure the impact of this design choice.

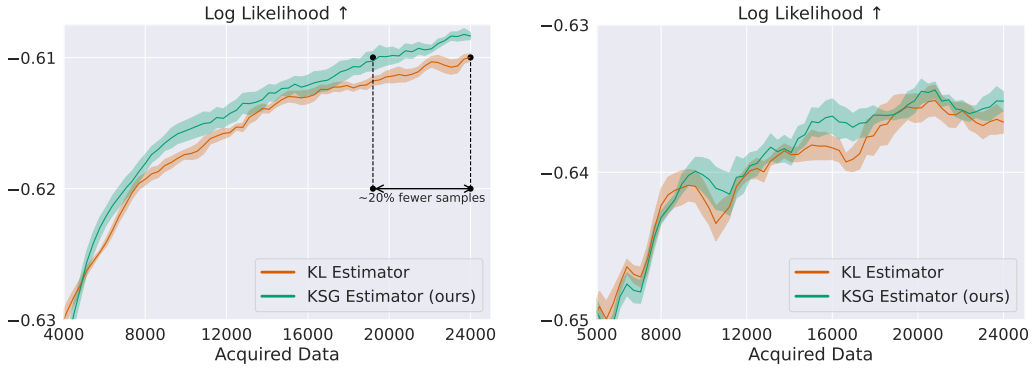
Figure 10 presents the results of this ablation. In the Reddit TL;DR dataset, the KSG estimator consistently outperforms the KL estimator, requiring approximately 20% fewer samples. In the OOD setting, both estimators performed equally. This is expected once that the available pool and training set does not provide support in the regions of the feature space with out-of-distribution samples.

https://doi.org/10.52202/079017-3749



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset

Figure 9: **The ablation study of the type of Uncertainty in the BAL-PM objective**. Leveraging Epistemic Uncertainty substantially surpasses Predictive Uncertainty since it disregards the effect of the high Aleatoric Uncertainty from preference datasets.



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset

Figure 10: **The ablation study of the type of entropy estimator in the BAL-PM objective**. Using the KSG estimator requires approximately 20% fewer samples than the KL estimator in the Reddit TL;DR dataset.

https://doi.org/10.52202/079017-3749

E KL Entropy Estimator: Review and Assumptions

In this Section, we review the derivation of the KL entropy estimator and highlight the main assumptions and how they impacted the design choices for BAL-PM. We mostly follow the derivation from Kraskov et al. [54].

We start defining X as a continuous random variable in a vector space where the Euclidean norm $\|x - x^*\|$ is well-defined (x and $x*$ are two realizations of X). Let $\mu(x)$ represent the density of X over this vector space. The Shannon entropy is defined as:

$$\mathcal{H}(X) \coloneqq -\mathbb{E}_{\mu(x)}[\log \mu(x)] = -\int \mu(x) \log \mu(x) dx. \tag{12}$$

To build an estimator, we can approximate Equation 12 via Monte-Carlo sampling:

$$\hat{\mathcal{H}}(X) = \frac{1}{N} \sum_{i=0}^{N} \log \hat{\mu}(x)), \tag{13}$$

where N is the number of samples for approximation and $\hat{\mu}(x)$ is an estimate of the density of X.

The goal of kNN-based entropy estimators is primarily to provide a good approximation for the density. For this matter, we first define a probability distribution $P_k(\epsilon)$ for the distance between any realization x_i and its k-nearest Neighbor. We start highlighting the first assumption:

Assumption 1. *The probability $P_k(\epsilon)d\epsilon$ is equal to the chance of existing one point such that $\|x - x^*\| < \epsilon/2$, $k-1$ other points with smaller distances, and $N-k-1$ points with larger distances.*

Following this assumption we can obtain the following trinomial distribution:

$$P_k(\epsilon) = k\binom{N-1}{k} \frac{dp_i(\epsilon)}{d\epsilon} p_i^{k-1} (1-p_i)^{N-k-1}, \tag{14}$$

where $p_i(\epsilon)$ is the probability mass of the ϵ-ball centered in x_i. The expectation of $\log p_i(\epsilon)$ is:

$$\begin{aligned} \mathbb{E}[\log p_i(\epsilon)] &= \int_0^\infty P_k(\epsilon) \log p_i(\epsilon) d\epsilon = k\binom{N-1}{k} \int_0^1 p_i^{k-1} (1-p_i)^{N-k-1} \log p_i \\ &= \psi(N) - \psi(k), \end{aligned} \tag{15}$$

where ψ denotes the digamma function. We now highlight the second assumption:

Assumption 2. *The density $\mu(x)$ is constant in the ϵ-ball.*

Assumption 2 allows us to approximate $p_i(\epsilon) \approx c_d \epsilon^d \mu(x_i)$, where d is the dimension of x and c_d is the volume of the d-dimensional unit ball. Using Equation 15 in this approximation and rearranging terms, we have:

$$\log \hat{\mu}(x_i) \approx \psi(k) - \psi(N) - d\mathbb{E}[\log \epsilon] - \log c_d. \tag{16}$$

Finally, using this estimator in Equation 13, we obtain the KL entropy estimator in Equation 4.

Remarks. Now, we analyze how this derivation and assumptions impact our entropy estimator. First, Assumption 1 models the probability based on the choice of k. For the low-data regime (i.e., N is small), this could lead to considerably large ϵ-balls where the Assumption 2 does not hold, and, therefore, it is not a good approximation. Thus, naively applying the KL estimator in the acquired training set may lead to strongly biased entropy estimates.

Secondly, in Section 4, we raise the key insight that Equation 4 holds for *any* value of k, and it does not require a fixed k over different particles for entropy estimation. Indeed, the density estimation $\hat{\mu}(x_i)$ is estimated for each particle x_i in isolation (Equation 16). Therefore, we may choose a different k for each particle to ensure that Assumptions 1 and 2 are valid.

F BAL-PM Objective – Empirical Analysis

Balancing Task-Dependent and Task-Agnostic Epistemic Uncertainty for Active Learning. Since considering the information in the feature space is crucial for Active Preference Modeling, a relevant question arises: how should an algorithm balance the contributions between the Bayesian preference model epistemic uncertainty and the prompt feature space uncertainty? Excessive reliance on the task-dependent term leads to acquiring redundant points. Similarly, the exacerbated contribution of the task-agnostic term prevents the acquisition of the points that reduce the uncertainty in the preference model. Thus, we investigate how BAL-PM balances these two terms over active learning. In Figure 11, we show the ratio of the entropy and preference model epistemic uncertainty scores in the first selected point of each acquired batch. Interestingly, BAL-PM automatically adjusts the contribution of each term. It progressively decays and converges the influence of the entropy score (task-agnostic source) as the novelty in the prompt feature space reduces due to the acquisition of new points. Similarly, it increases the relevance of the preference model uncertainty estimates (task-dependent source). A positive downstream effect is that BAL-PM switches to a more task-dependent selection as it improves the Bayesian model and, consequently, its epistemic uncertainty estimates.

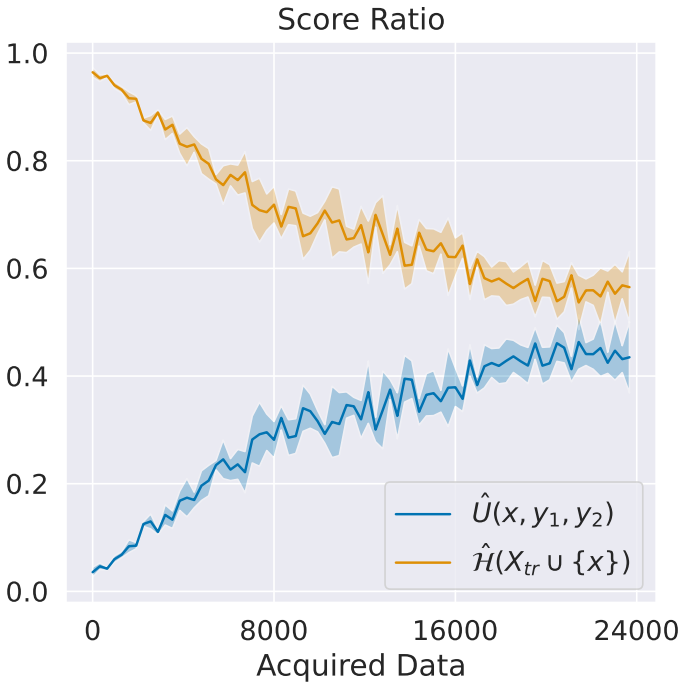


Figure 11: **Ratio of entropy and preference model uncertainty scores**. This plot represents the normalized contributions from the terms of Equation 5 on the first selected point of each batch. BAL-PM automatically adjusts the contribution based on the information available in the pool set.

https://doi.org/10.52202/079017-3749

G Is Log-Likelihood a proper performance measure for Preference Modeling?

In this Section, we argue why the Average Log-Likelihood on the test set is a good performance measure for Preference Modeling. Given a test set $\mathcal{D}_{test} = \{(x, y_1, y_2, y_1 \succ y_2)\}^N$ and the learned preference model $p_{\boldsymbol{\theta}}(y_1 \succ y_2 \mid x, y_1, y_2)$, the average Log-Likelihood is given by:

$$LL(\mathcal{D}_{test}, \boldsymbol{\theta}) = \mathbb{E}_{(x,y_1,y_2,y_1 \succ y_2) \sim \mathcal{D}_{test}} [\log p_{\boldsymbol{\theta}}(y_1 \succ y_2 \mid x, y_1, y_2)]. \tag{17}$$

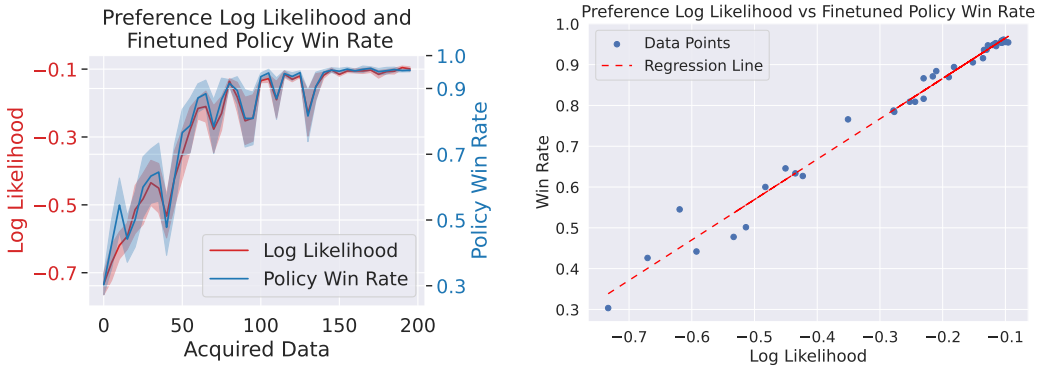
Equation 17 is exactly the objective maximized in standard binary classification (or, equally, the minimization of the negative Log-Likelihood loss) but computed over the test data. In other words, this is the negative "test loss".

Average LL is a typical metric in the Active Learning and Uncertainty Quantification literature [61–63]. For Preference Modeling, it is very relevant as **LL directly accounts for the *preference strength* to rank models**: given a triple (x, y_1, y_2) where all raters agree that y_1 is preferable over y_2, LL allows us to measure that a model A predicting $p_A(y_1 \succ y_2 \mid x, y_1, y_2) = 0.9$, $(LL \approx -0.1)$ is better (in that data point) than another model B predicting $p_B(y_1 \succ y_2 \mid x, y_1, y_2) = 0.6$ $(LL \approx -0.5)$. Accuracy would provide an equal score for both models since it only accounts for the binarized prediction. LL provides a more "fine-grained" measure.

Another crucial point is that **LL factors in the aleatoric uncertainty in the label-generating process**. For instance, in a scenario where only 70% of the raters agree that y_1 is preferable, LL better ranks models whose predictions are closer to $p = 0.7$, respecting the ground truth preference strength, which is not possible with accuracy.

G.1 Do the models better ranked by Average Log Likelihood (LL) lead to better fine-tuned policies?

In the context of Preference Modeling, fine-tuning LM policies is currently a very relevant downstream task. The Preference Modeling optimization objective and model selection protocol adopted in this work follow exactly the prior influential work on the topic [6, 1], which provides evidence that better preference models (in terms of validation loss) lead to improved downstream policies. Thus, we expect our models to behave similarly under the same conditions.



(a) (b)

Figure 12: **The Relationship between the Test Log-Likelihood of a Preference Model and the Performance of the corresponding fine-tuned Policy**. We show that, under simple conditions, there is a strong correlation between these two performance measures.

As additional evidence, we empirically illustrate the relationship between Log-Likelihood and policy performance on a simplified setup (Figure 12). Here, prompts x and completions y are real numbers in $[0, 1]$. The ground-truth reward function is given by a Gaussian density function $r(x, y) = \mathcal{N}(x + y \mid \mu = 1.0, \sigma = 0.4)$, and true preferences follow the Bradley-Terry model. In this setup, we progressively increase the training set size (the x-axis in Figure 12a) at which we train the preference models. This process generates different models with increasing levels of

https://doi.org/10.52202/079017-3749

test-set average Log-Likelihood. Then, similar to [64], we optimize the base policy via a Best-of-N optimizer by leveraging each of these learned preference models. Finally, we report the rate at which the fine-tuned policy completion is preferable over the base policy completion according to the ground truth reward model ("win rate"). Although simple, this setting allows us to bypass several optimization and distributional challenges and solely focus on evaluating the relationship between average Log-Likelihood and the performance of the fine-tuned policy. Figure 12a reports the Log-Likelihood (red) and the win rate against the base policy (blue). Figure 12b directly plots both measures and fits a regression line. We observe a strong correlation, which aligns with our point: **a higher test-set average Log-Likelihood means that the preference model is better at predicting the ground truth preferences, assigning higher rewards for better completions, and, therefore, improving fine-tuned policies that maximize such reward scores**.

H BAL-PM Computational Cost Description

In this section, we describe the computational cost of executing BAL-PM. We argue that computational tractability is one of the main contributions of our method. We start by providing some context: our work focuses on (Bayesian) Active Learning, which is naturally more computationally demanding than simply training predictive models. This is because **we require models that express epistemic uncertainty** to acquire informative labels for efficient training. This also **requires models to constantly update their uncertainties based on the new data via re-training**. The key is that **Active Learning reduces the number of labels needed to train a better model, which considerably overcomes the extra computational cost**. Labeling is significantly more expensive and laborious.

As described in Section 1, Preference Modeling in LLMs requires batch acquisition – it is impossible to request the label of a single point, re-train the model, and repeat this process. Still, tractable methods rely on these single-point acquisition objectives. Thus, **what BAL-PM does computationally is to replace B model re-trainings per acquired batch with computing entropy estimates** (considerably cheaper, as explained below). B is the batch size, and we set $B = 320$ in our experiments.

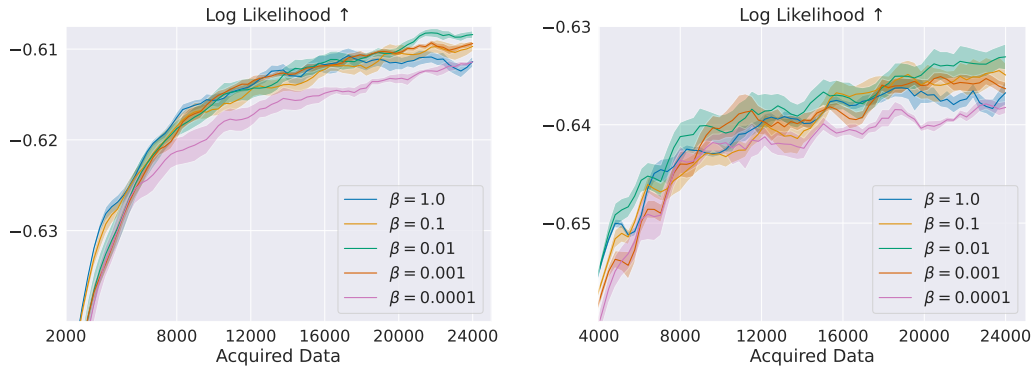
BAL-PM does not require training or inference on LLMs during the active learning loops. This considerably reduces the computational cost and allows us to scale up to 140b models in a single A100 GPU. Comparatively, even fully fine-tuning a 7b-parameter model currently requires at least several A100 GPUs. LoRA methods [65] also require new LLM inferences for every model update, while BAL-PM only requires a single time.

The computation of BAL-PM has three pieces: offline processing (LLM inference and kNN computation), updating adapters, and entropy estimation. LLM inference is done only once before Active Learning, which is the minimum for LLM adoption. Furthermore, **we may compute the features used for the preference model and sentropy estimation in the same forward pass**: every prompt-completion input concatenates prompt/completion texts. Thus, we can extract prompt features as the last layer embedding right after the last prompt token and the prompt-completion features right after the completion's last token. Hence, there is no extra cost to extract features for entropy estimation. The cost of updating adapters is minimal: it boils down to updating MLPs with two hidden layers, which is reasonably cheap for LLM research. Lastly, the entropy estimation only requires computing the di-gamma function (Equation 11) in the pool.

 https://doi.org/10.52202/079017-3749

I β Robustness Analysis

In this Section, we introduce a robustness analysis for the β hyperparameter (Figure 8) considering the values in the search space, described in Table 2. As presented in Equation 5, this hyperparameter balances the effect of the epistemic uncertainty and entropy scores.



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset (OOD)

Figure 13: β **Robustness Analysis**. We considered different scales of the β hyperparameter in the BAL-PM objective (Equation 5).

The impact of the choice is more noticeable when values are 100x greater/lower than the optimal choice. Values around 10x greater/lower still perform well, suggesting good room for choosing this hyperparameter. Furthermore, we employed the same value of β across the different datasets and LLMs, suggesting robustness across different relevant dimensions. Crucially, β trades off the contribution of two different terms. As such, it provides a spectrum of objectives and may recover the two extremes presented in the ablation of Figure 13. Naturally, different choices of β will change the uncertainty score ratio in Figure 11 on Appendix F (i.e., the contribution of each term after convergence). Nevertheless, and most importantly, the behavior of the curves — the entropy contribution progressively reducing and converging and the relevance of epistemic uncertainty estimates increasing — should remain.

J Further Baselines

In this section, we provide additional baselines for further comparison.

J.1 Full Dataset Baseline

We start by evaluating the performance of a preference model trained in the full dataset. Figure 14 presents this result in purple, with the shaded area representing the standard error computed across five seeds. BAL-PM achieves on-par performance to this baseline, although it only requires 24000 data points (the full dataset contains 92,858 points). This result is another strong evidence of the sample efficiency of our method.

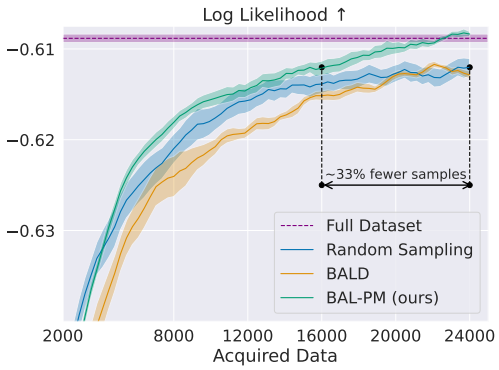
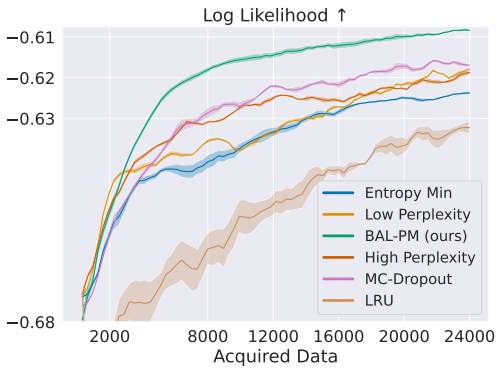
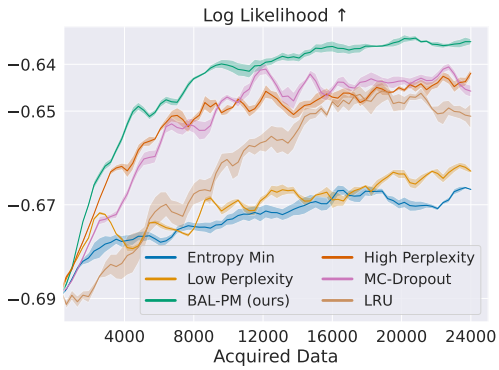


Figure 14: **Comparison with Preference Model trained on the full dataset.**

J.2 Additional Data Selection Methods



(a) Reddit TL;DR (Test)



(b) CNN/DM Dataset (OOD)

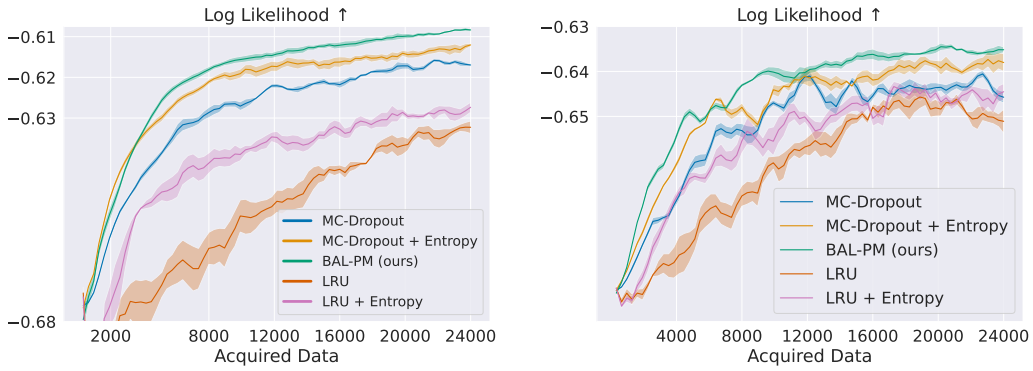
Figure 15: **Comparison with several additional baselines for Active Preference Modeling**. These baselines focus on different notions of uncertainty and diversity for acquiring samples.

We considered the following additional baselines:

- **Entropy Minimizer**: Entropy Minimizer: Inspired by Liu et al. [66], we consider an objective that, in addition to selecting points with high epistemic uncertainty, also selects points that are semantically similar to the current training points. This is equivalent to selecting points that increase the entropy of the prompt distribution **the least**, thus the name "Entropy Minimizer". It serves as a check for our central hypothesis that entropy maximization leads to better batch active learning.
- **Perplexity**: Inspired by Gonen et al. [67], we consider an objective that selects points based on the perplexity of the base LLM. We consider two versions: one that chooses points with lower perplexity (Low Perplexity) and another with higher perplexity (High Perplexity). This is an interesting baseline since perplexity is equivalent to the predictive entropy of the token distribution. Therefore, it helps to analyze how much the base LLM "knows what it does not know" in terms of preference modeling.
- **MC Dropout** [63]: This method performs approximate Bayesian inference via executing dropout at test time to generate different parameter hypotheses. Therefore, it can express epistemic uncertainty, which is used to select the most informative points.
- **Latent Reward Uncertainty** (LRU): This method computes a reward distribution over the data points by leveraging the latent reward model learned via the Bradley-Terry model.

https://doi.org/10.52202/079017-3749

Then, it selects extreme points (too high or too low rewards) as a proxy for the uncertainty of the model.



(a) Reddit TL;DR (Test) (b) CNN/DM Dataset (OOD)

Figure 16: **The effect of incorporating the entropy objective in uncertainty baselines.** This shows how our proposed objective can also boost the performance of different baselines.

Figure 15 reports performances for both test and OOD sets. In both cases, BAL-PM outperforms additional baselines. In the sequence, MC-Dropout works best as the baseline that targets the epistemic uncertainty of a Bayesian model. Unsurprisingly, Entropy Minimizer and Low Perplexity work worse since they target points with lower entropy. LRU presented mixed results, suggesting that the latent reward may not represent well the preference model's uncertainty. More interestingly, while these models can represent different uncertainties to seek informative points, they naturally cannot provide in-batch diversity - they suffer from the same challenges as BALD. In this perspective, the BAL-PM objective can also improve upon those methods, as we show in Figure 16: we combined MC-Dropout and LRU with our entropy term to provide in-batch diversity, which consistently improved both methods across the datasets.

K Batch Samples

In this Section, we present some selected samples of the first acquired batch from BALD (Table 3) and BAL-PM (Table 4) for comparison. We sorted the points alphabetically to highlight duplicated prompts. BALD consistently acquires duplicated points, sometimes sampling more than ten times the same prompt. In contrast, BAL-PM samples semantically diverse points with no duplicates.

BALD – Acquired Batch (Truncated Prompts)
A bit of backstory: I've been in only 4 real long term relationships in my past....
A bit of backstory: I've been in only 4 real long term relationships in my past....
A bit of backstory: I've been in only 4 real long term relationships in my past....
A few weeks ago my wife admitted to me that my best friend, (let's call him Marc...
A week ago I called off my relationship with my partner for a number of reasons,...
About a month ago my (23 F) boyfriend (26 M) of three and a half years and I got...
After 8 months my girlfriend decided to break up with me. Shes a very nice girl ...
For starters, its been awhile loseit, and I missed you! Things have been crazzzy...
For starters, its been awhile loseit, and I missed you! Things have been crazzzy...
For starters, its been awhile loseit, and I missed you! Things have been crazzzy...
For starters, its been awhile loseit, and I missed you! Things have been crazzzy...
Hello all I need some help regarding a friend of mine and a dream she had, well ...
Hello everyone, I am a student at a boarding school which means I am away from m...
Hi all. I am using a throwaway. I am 29f and my boyfriend is 32m. We have been d...
Hi all. I am using a throwaway. I am 29f and my boyfriend is 32m. We have been d...
Hi all. I am using a throwaway. I am 29f and my boyfriend is 32m. We have been d...
Hi first time user, and I am dyslexic so please forgive any spelling errors. T...
I am 31 years old and currently live in New York. I have been a professional tre...
I was sitting on a bus and the seat beside me was empty.. A young nun walked do...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I work inside of a bread depot, and the drivers are effectively brokers, or our ...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
I've been married to my husband for 3 years, it's been wonderful, I couldn't ask...
It was my school's annual 5K, so the runners are students, faculty, and then ran...
Ive worked with this girl once a week for almost a year. When we met we were bot...
Ive worked with this girl once a week for almost a year. When we met we were bot...
Ive worked with this girl once a week for almost a year. When we met we were bot...
Ive worked with this girl once a week for almost a year. When we met we were bot...
Ive worked with this girl once a week for almost a year. When we met we were bot...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...
My girlfriend and I have been going out for about a year and have decided to mov...

Table 3: **First Acquired Batch by BALD (baseline)**.

https://doi.org/10.52202/079017-3749

BAL-PM – Acquired Batch (Truncated Prompts)
Quick Background: As the title states, we've been together for 7 years datin...
The texts: Him: at least my mom thinks I'm cute me: I think you're cute ;)...
**Warning: Avoid this film if you only broke up very recently! I advise this fil...
i(26m) have been dating her(26f) on and off for 5 years. I have come to the...
— So we broke up as in words she had severe depression and it wasn't fair to m...
A little while back, my sister asked me why some men were homophobic. I answere...
A small background. I live in in Puerto Rico, where I haven't had to good an ex...
About a month ago we started having problems with our cable. The picture would g...
Background info: Little background. I started medical school a few years back. I...
Backstory: I'm a 17 year old student in the U.K. currently in sixth-form. Back i...
Be sure to explain in detail with line breaks. Hey my name is Matt and i honest...
Because I live in a very conservative Catholic neighborhood, I cannot come out a...
Hello all, Story: I played around with some stocks a few years back buying ...
Hello reddit My LDR girlfriend of six months told me yesterday that she wasn't ...
Hello! Last group of friends I had was back in 10th Grade. Since then my depre...
Hello! I'm a 23 y/o F dating a 30 y/o male. This is by far the best relationship...
Hello, I'm kind of new to this sub reddit but I figured I'd get an opinion from ...
Hey Reddit. I spent at least 20 mins looking for the correct sub-reddit for men'...
Hey all. My classmates and I at the SUNY Purchase Film Conservatory are in the p...
Hey everyone so I'm about 3 months in of my 6 month regimen before I get gastric...
Hey fellow revenge-lovers, here's a quick one, that happened about an hour ago. ...
Hey guys this is strange to begin with, but I"ll introduce the situation: I'm ...
I am a 24 y/o male and I have been dating a girl who is 22 years old for about 1...
I am an 18 year old college student and I have no attachments to my local area. ...
I am aware that this has been proposed before. I personally believe that it woul...
I am dating a complete dime like I get compliments all the time about her from s...
I can't focus. I can't become and remain motivated. When I've learned something ...
I know we are young but bear with me, I didn't know where else to go for this ty...
I live in California and am the co-owner of a car, with the names on the title b...
I made a previous post here but it sounded kind of stupid with the way I phrased...
I met my current girlfriend in highschool. She's the only woman I've ever been ...
I should start with saying neither of us have had a chance to travel anywhere ex...
I want to start off by saying I love my SO and I'm looking for suggestions befor...
I was in a pretty serious car accident this week, and my car was easily totaled....
I will try to make this as short as possible. a long time ago i met this girl, ...
I'll keep it short :3 I'm 18, he's 18. Dating for 3 years. When we walk togethe...
I'll keep this as succinct as possible. I moved in Sept. 1. I used to live here...
I've been with my boyfriend for 4 years, it hasn't been the best relationship, b...
I've been with my gf for almost 7 years. Lived together for about 5 years. A few...
I've been working with this girl for 2 months. it started at work where i was he...
If you want to understand the scam, here's what's happening: Okay, so I found a...
Im 27. Single. I am a productive member of society. I work full time i pay my ow...
It was New Year's Eve and my family was driving off to my grandparents' house. H...
It wasn't that long term relationships but we lived together for 6 months so we ...
Just got the new Kobo touch and they provided me with a $10 gift card for their ...
My friend's little brother is really suffering in his dorm. He's lost 15-20 poun...
My girlfriend an I have been dating for three years. Its been the best time of m...
My girlfriend and I met through family friends a year and a half ago. We've been...

Table 4: **First Acquired Batch by BAL-PM**.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Sections 4 and 5.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

https://doi.org/10.52202/079017-3749

Justification: No theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer:[Yes]

   Justification: See the Reproducibility Statement (Appendix B).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer:[Yes]

Justification: See the Reproducibility Statement (Appendix B).

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Refer to Section 5 and Appendix C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See details in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

 https://doi.org/10.52202/079017-3749

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See details in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See the Impact Statement (Appendix A).

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cited the frameworks and dataset owners used in this work. See B.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

https://doi.org/10.52202/079017-3749

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: All code is available and documented in the link provided. See Appendix B.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.