Measuring and Improving Model-Moderator Collaboration using Uncertainty Estimation

Ian D. Kivlichan[*] Jigsaw kivlichan@google.com

Zi Lin[*†] Google Research lzi@google.com

Jeremiah Liu[*] Google Research jereliu@google.com

Lucy Vasserman Jigsaw lucyvasserman@google.com

Abstract

Content moderation is often performed by a collaboration between humans and machine learning models. However, it is not well understood *how* to design the collaborative process so as to maximize the combined moderator-model system performance. This work presents a rigorous study of this problem, focusing on an approach that incorporates model uncertainty into the collaborative process. First, we introduce principled metrics to describe the performance of the collaborative system under capacity constraints on the human moderator, quantifying how efficiently the combined system utilizes human decisions. Using these metrics, we conduct a large benchmark study evaluating the performance of state-of-the-art uncertainty models under different collaborative review strategies. We find that an uncertainty-based strategy consistently outperforms the widely used strategy based on toxicity scores, and moreover that the choice of review strategy drastically changes the overall system performance. Our results demonstrate the importance of rigorous metrics for understanding and developing effective moderator-model systems for content moderation, as well as the utility of uncertainty estimation in this domain.[1]

1 Introduction

Maintaining civil discussions online is a persistent challenge for online platforms. Due to the sheer scale of user-generated text, modern content moderation systems often employ machine learning algorithms to automatically classify user comments based on their toxicity, with the goal of flagging a collection of likely policy-violating content for human experts to review (Etim, 2017). However, modern deep learning models have been shown to suffer from reliability and robustness issues, especially in the face of the rich and complex sociolinguistic phenomena in real-world online conversations. Examples include possibly generating confidently wrong predictions based on spurious lexical features (Wang and Culotta, 2020), or exhibiting undesired biases toward particular social subgroups (Dixon et al., 2018). This has raised questions about how current toxicity detection models will perform in realistic online environments, as well as the potential consequences for moderation systems (Rainie et al., 2017).

In this work, we study an approach to address these questions by incorporating model uncertainty into the collaborative model-moderator system's decision-making process. The intuition is that by using uncertainty as a signal for the likelihood of model error, we can improve the efficiency and performance of the collaborative moderation system by prioritizing the least confident examples from the model for human review. Despite a plethora of uncertainty methods in the literature, there has been limited work studying their effectiveness in improving the performance of human-AI collaborative systems with respect to application-specific metrics and criteria (Awaysheh et al., 2019; Dusenberry et al., 2020; Jesson et al., 2020). This is especially important for the content moderation task: real-world practice has unique challenges and constraints, including label imbalance, distributional shift, and limited resources of human experts; how these factors impact the collaborative system's effectiveness is not well understood.

In this work, we lay the foundation for the study of the uncertainty-aware collaborative content moderation problem. We first (1) propose rigorous met-

[*] Equal contribution; authors listed alphabetically.

[†] This work was done while Zi Lin was an AI resident at Google Research.

[1] Complete code including metric implementations and experiments is available at http://github.com/google/uncertainty-baselines/tree/master/baselines/toxic_comments.

Proceedings of the Fifth Workshop on Online Abuse and Harms, pages 36–53
August 6, 2021. ©2021 Association for Computational Linguistics

rics *Oracle-Model Collaborative Accuracy* (OC-Acc) and *AUC* (OC-AUC) to measure the performance of the overall collaborative system under capacity constraints on a simulated human moderator. We also propose *Review Efficiency*, a intrinsic metric to measure a model's ability to improve the collaboration efficiency by selecting examples that need further review. Then, (2) we introduce a challenging data benchmark, *Collaborative Toxicity Moderation in the Wild* (CoToMoD), for evaluating the effectiveness of a collaborative toxic comment moderation system. CoToMoD emulates the realistic train-deployment environment of a moderation system, in which the deployment environment contains richer linguistic phenomena and a more diverse range of topics than the training data, such that effective collaboration is crucial for good system performance (Amodei et al., 2016). Finally, (3) we present a large benchmark study to evaluate the performance of five classic and state-of-the-art uncertainty approaches on CoToMoD under two different moderation review approaches (based on the uncertainty score and on the toxicity score, respectively). We find that both the model's predictive and uncertainty quality contribute to the performance of the final system, and that the uncertainty-based review strategy outperforms the toxicity strategy across a variety of models and range of human review capacities.

2 Related Work

Our collaborative metrics draw on the idea of classification with a reject option, or learning with abstention (Bartlett and Wegkamp, 2008; Cortes et al., 2016, 2018; Kompa et al., 2021). In this classification scenario, the model has the option to reject an example instead of predicting its label. The challenge in connecting learning with abstention to OC-Acc or OC-AUC is to account for how many examples have already been rejected. Specifically, the difficulty is that the metrics we present are all dataset-level metrics, i.e. the "reject" option is not at the level of individual examples, but rather a set capacity over the entire dataset. Moreover, this means OC-Acc and OC-AUC can be compared directly with traditional accuracy or AUC measures. This difference in focus enables us to consider human time as the limiting resource in the overall model-moderator system's performance.

One key point for our work is that the best model (in isolation) may not yield the best performance in collaboration with a human (Bansal et al., 2021). Our work demonstrates this for a case where the collaboration procedure is decided over the full dataset rather than per example: because of this, Bansal et al. (2021)'s expected team utility does not easily generalize to our setting. In particular, the user chooses which classifier predictions to accept after receiving all of them rather than per example.

Robustness to distribution shift has been applied to toxicity classification in other works (Adragna et al., 2020; Koh et al., 2020), emphasizing the connection between fairness and robustness. Our work focuses on how these methods connect to the human review process, and how uncertainty can lead to better decision-making for a model collaborating with a human. Along these lines, Dusenberry et al. (2020) analyzed how uncertainty affects optimal decisions in a medical context, though again at the level of individual examples rather than over the dataset.

3 Background: Uncertainty Quantification for Deep Toxicity Classification

Types of Uncertainty Consider modeling a toxicity dataset $\mathcal{D} = \{y_i, x_i\}_{i=1}^{N}$ using a deep classifier $f_W(x)$. Here the x_i are example comments, $y_i \sim p^*(y|x_i)$ are toxicity labels drawn from a data generating process p^* (e.g., the human annotation process), and W are the parameters of the deep neural network. There are two distinct types of uncertainty in this modeling process: *data uncertainty* and *model uncertainty* (Sullivan, 2015; Liu et al., 2019). *Data uncertainty* arises from the stochastic variability inherent in the data generating process p^*. For example, the toxicity label y_i for a comment can vary between 0 and 1 depending on raters' different understandings of the comment or of the annotation guidelines. On the other hand, *model uncertainty* arises from the model's lack of knowledge about the world, commonly caused by insufficient coverage of the training data. For example, at evaluation time, the toxicity classifier may encounter neologisms or misspellings that did not appear in the training data, making it more likely to make a mistake (van Aken et al., 2018). While the *model uncertainty* can be reduced by training on more data, the *data uncertainty* is inherent to the data generating process and is irreducible.

Estimating Uncertainty A model that quantifies its uncertainty well should properly capture both the data and the model uncertainties. To this end, a learned deep classifier $f_W(x)$ describes the *data uncertainty* via its predictive probability, e.g.:

$$p(y|x, W) = \text{sigmoid}(f_W(x)),$$

which is conditioned on the model parameter W, and is commonly learned by minimizing the Kullback-Leibler (KL) divergence between the model distribution $p(y|x, W)$ and the empirical distribution of the data (e.g. by minimizing the cross-entropy loss (Goodfellow et al., 2016)). On the other hand, a deep classifier can quantify *model uncertainty* by using probabilistic methods to learn the posterior distribution of the model parameters:

$$W \sim p(W).$$

This distribution over W leads to a distribution over the predictive probabilities $p(y|x, W)$. As a result, at inference time, the model can sample model weights $\{W_m\}_{m=1}^{M}$ from the posterior distribution $p(W)$, and then compute the posterior sample of predictive probabilities $\{p(y|x, W_m)\}_{m=1}^{M}$. This allows the model to express its model uncertainty through the variance of the posterior distribution $\text{Var}\big(p(y|x, W)\big)$. Section 5 surveys popular probabilistic deep learning methods.

In practice, it is convenient to compute a single uncertainty score capturing both types of uncertainty. To this end, we can first compute the marginalized predictive probability:

$$p(y|x) = \int p(y|x, W)p(W)\, dW$$

which captures both types of uncertainty by marginalizing the data uncertainty $p(y|x, W)$ over the model uncertainty $p(W)$. We can thus quantify the overall uncertainty of the model by computing the predictive variance of this binary distribution:

$$u_{\text{unc}}(x) = p(y|x) \times (1 - p(y|x)). \qquad (1)$$

Evaluating Uncertainty Quality A common approach to evaluate a model's uncertainty quality is to measure its *calibration* performance, i.e., whether the model's predictive uncertainty is indicative of the predictive error (Guo et al., 2017). As we shall see in experiments, traditional calibration metrics like the Brier score (Ovadia et al., 2019) do not correlate well with the model performance in collaborative prediction. One notable reason is that the collaborative systems use uncertainty as a ranking score (to identify possibly wrong predictions), while metrics like Brier score only measure the uncertainty's ranking performance indirectly.

		Uncertainty	
		Uncertain	Certain
Accuracy	Inaccurate	TP	FN
	Accurate	FP	TN

Figure 1: Confusion matrix for evaluating uncertainty calibration. We describe the correspondence in the text.

This motivates us to consider *Calibration AUC*, a new class of calibration metrics that focus on the uncertainty score $u_{\text{unc}}(x)$'s ranking performance. This metric evaluates uncertainty estimation by recasting it as a binary prediction problem, where the binary label is the model's prediction error $\mathbb{I}(f(x_i) \neq y_i)$, and the predictive score is the model uncertainty. This formulation leads to a confusion matrix as shown in Figure 1 (Krishnan and Tickoo, 2020). Here, the four confusion matrix variables take on new meanings: (1) True Positive (TP) corresponds to the case where the prediction is inaccurate and the model is uncertain, (2) True Negative (TN) to the accurate and certain case, (3) False Negative (FN) to the inaccurate and certain case (i.e., over-confidence), and finally (4) False Positive (FP) to the accurate and uncertain case (i.e., under-confidence). Now, consider having the model predict its testing error using model uncertainty. The precision (TP/(TP+FP)) measures the fraction of inaccurate examples where the model is uncertain, recall (TP/(TP+FN)) measures the fraction of uncertain examples where the model is inaccurate, and the false positive rate (FP/(FP+TN)) measures the fraction of under-confident examples among the correct predictions. Thus, the model's calibration performance can be measured by the area under the precision-recall curve (***Calibration AUPRC***) and under the receiver operating characteristic curve (***Calibration AUROC***) for this problem. It is worth noting that the calibration AUPRC is closely related to the intrinsic metrics for the model's collaborative effectiveness: we discuss this in greater detail for the *Review Efficiency* in Section 4.1 and Appendix A.2). This renders it especially suitable for evaluating model uncertainty in the context of collaborative content moderation.

4 The Collaborative Content Moderation Task

Online content moderation is a *collaborative* process, performed by humans working in conjunction with machine learning models. For example, the model can select a set of likely policy-violating posts for further review by human moderators. In this work, we consider a setting where a neural model interacts with an "oracle" human moderator with limited capacity in moderating online comments. Given a large number of examples $\{x_i\}_{i=1}^n$, the model first generates the predictive probability $p(y|x_i)$ and review score $u(x_i)$ for each example. Then, the model sends a pre-specified number of these examples to human moderators according to the rankings of the review score $u(x_i)$, and relies on its prediction $p(y|x_i)$ for the rest of the examples. In this work, we make the simplifying assumption that the human experts act like an oracle, correctly labeling all comments sent by the model.

4.1 Measuring the Performance of the Collaborative Moderation System

Machine learning systems for online content moderation are typically evaluated using metrics like accuracy or area under the receiver operating characteristic curve (AUROC). These metrics reflect the origins of these systems in classification problems, such as for detecting / classifying online abuse, harassment, or toxicity (Yin et al., 2009; Dinakar et al., 2011; Cheng et al., 2015; Wulczyn et al., 2017). However, they do not capture the model's ability to effectively collaborate with human moderators, or the performance of the resultant collaborative system.

New metrics, both extrinsic and intrinsic (Mollá and Hutchinson, 2003), are one of the core contributions of this work. We introduce extrinsic metrics describing the performance of the overall model-moderator collaborative system (Oracle-Model Collaborative Accuracy and AUC, analogous to the classic accuracy and AUC), and an intrinsic metric focusing on the model's ability to effectively collaborate with human moderators (Review Efficiency), i.e., how well the model selects the examples in need of further review.

Extrinsic Metrics: Oracle-model Collaborative Accuracy and AUC To capture the collaborative interaction between human moderators and machine learning models, we first propose ***Oracle-Model Collaborative Accuracy* (OC-Acc)**. OC-Acc measures the combined accuracy of this collaborative process, subject to a limited review capacity α for the human oracle (i.e., the oracle can process at most $\alpha \times 100\%$ of the total examples). Formally, given a dataset $D = \{(x_i, y_i)\}_{i=1}^n$, for a predictive model $f(x_i)$ generating a review score $u(x_i)$, the Oracle-Model Collaborative Accuracy for example x_i is

$$\text{OC-Acc}(x_i|\alpha) = \begin{cases} 1 & \text{if } u(x_i) > q_{1-\alpha} \\ \mathbb{I}(f(x_i) = y_i) & \text{otherwise} \end{cases},$$

Thus, over the whole dataset, $\text{OC-Acc}(\alpha) = \frac{1}{n}\sum_{i=1}^n \text{OC-Acc}(x_i|\alpha)$. Here $q_{1-\alpha}$ is the $(1-\alpha)^{\text{th}}$ quantile of the model's review scores $\{u(x_i)\}_{i=1}^n$ over the entire dataset. OC-Acc thus describes the performance of a collaborative system which defers to a human oracle when the review score $u(x_i)$ is high, and relies on the model prediction otherwise, capturing the real-world usage and performance of the underlying model in a way that traditional metrics fail to.

However, as an accuracy-like metric, OC-Acc relies on a set threshold on the prediction score. This limits the metric's ability in describing model performance when compared to threshold-agnostic metrics like AUC. Moreover, OC-Acc can be sensitive to the intrinsic class imbalance in the toxicity datasets, appearing overly optimistic for model predictions that are biased toward negative class, similar to traditional accuracy metrics (Borkan et al., 2019). Therefore in practice, we prefer the AUC analogue of Oracle-Model Collaborative Accuracy, which we term the ***Oracle-Model Collaborative AUC* (OC-AUC)**. OC-AUC measures the same collaborative process as the OC-Acc, where the model sends the predictions with the top $\alpha \times 100\%$ of review scores. Then, similar to the standard AUC computation, OC-AUC sets up a collection of classifiers with varying predictive score thresholds, each of which has access to the oracle exactly as for OC-Acc (Davis and Goadrich, 2006). Each of these classifiers sends the same set of examples to the oracle (since the review score $u(x)$ is threshold-independent), and the oracle corrects model predictions when they are incorrect given the threshold. The OC-AUC—both OC-AUROC and OC-AUPRC—can then be calculated over this set of classifiers following the standard AUC algorithms (Davis and Goadrich, 2006).

Intrinsic Metric: Review Efficiency The metrics so far measure the performance of the overall collaborative system, which combines both the model's predictive accuracy and the model's effectiveness in collaboration. To understand the source of the improvement, we also introduce ***Review Efficiency***, an intrinsic metric focusing solely on the model's effectiveness in collaboration. Specifically, *Review Efficiency* is the proportion of examples sent to the oracle for which the model prediction would otherwise have been incorrect. This can be thought of as the model's precision in selecting inaccurate examples for further review (TP/(TP+FP) in Figure 1).

Note that the system's overall performance (measured by the oracle-model collaborative accuracy) can be rewritten as a weighted sum of the model's original predictive accuracy and the Review Efficiency (RE):

$$\text{OC-Acc}(\alpha) = \text{Acc} + \alpha \times \text{RE}(\alpha) \qquad (2)$$

where $\text{RE}(\alpha)$ is the model's review efficiency among all the examples whose review score $u(x_i)$ are greater than $q_{1-\alpha}$ (i.e., those sent to human moderators). Thus, a model with better predictive performance and higher review efficiency yields better performance in the overall system. The benefits of review efficiency become more pronounced as the review fraction α increases. We derive Eq. (2) in Appendix B.

4.2 CoToMoD: An Evaluation Benchmark for Real-world Collaborative Moderation

In a realistic industrial setting, toxicity detection models are often trained on a well-curated dataset with clean annotations, and then deployed to an environment that contains a more diverse range of sociolinguistic phenomena, and additionally exhibits systematic shifts in the lexical and topical distributions when compared to the training corpus.

To this end, we introduce a challenging data benchmark, ***Collaborative Toxicity Moderation in the Wild*** **(CoToMoD)**, to evaluate the performance of collaborative moderation systems in a realistic environment. CoToMoD consists of a set of *train*, *test*, and *deployment* environments: the *train* and *test* environments consist of 200k comments from Wikipedia discussion comments from 2004–2015 (the Wikipedia Talk Corpus (Wulczyn et al., 2017)), and the *deployment* environment consists of one million public comments appeared on approximately 50 English-language news sites across the world from 2015–2017 (the CivilComments dataset (Borkan et al., 2019)). This setup mirrors the real-world implementation of these methods, where robust performance under changing data is essential for proper deployment (Amodei et al., 2016).

Notably, CoToMoD contains two data challenges often encountered in practice: (1) *Distributional Shift*, i.e. the comments in the training and deployment environments cover different time periods and surround different topics of interest (Wikipedia pages vs. news articles). As the CivilComments corpus is much larger in size, it contains a considerable collection of long-tail phenomena (e.g., neologisms, obfuscation, etc.) that appear less frequently in the training data. (2) *Class Imbalance*, i.e. the fact that most online content is not toxic (Cheng et al., 2017; Wulczyn et al., 2017). This manifests in the datasets we use: roughly 2.5% (50,350 / 1,999,514) of the examples in the CivilComments dataset, and 9.6% (21,384 / 223,549) of the examples in Wikipedia Talk Corpus examples are toxic (Wulczyn et al., 2017; Borkan et al., 2019). As we will show, failing to account for class imbalance can severely bias model predictions toward the majority (non-toxic) class, reducing the effectiveness of the collaborative system.

5 Methods

Moderation Review Strategy In measuring model-moderator collaborative performance, we consider two review strategies (i.e. using different review scores $u(x)$). First, we experiment with a common toxicity-based review strategy (Jigsaw, 2019; Salganik and Lee, 2020). Specifically, the model sends comments for review in decreasing order of the predicted toxicity score (i.e., the predictive probability $p(y|x)$), equivalent to a review score $u_{\text{tox}}(x) = p(y|x)$. The second strategy is uncertainty-based: given $p(y|x)$, we use uncertainty as the review score, $u_{\text{unc}}(x) = p(y|x)(1 - p(y|x))$ (recall Eq. (1)), so that the review score is maximized at $p(y|x) = 0.5$, and decreases toward 0 as $p(x)$ approaches 0 or 1. Which strategy performs best depends on the toxicity distribution in the dataset and the available review capacity α.

Uncertainty Models We evaluate the performance of classic and the latest state-of-the-art probabilistic deep learning methods on the CoToMoD benchmark. We consider BERT$_{\texttt{base}}$ as the base model (Devlin et al., 2019), and select five methods based on their practical applicability for transformer models. Specifically, we consider (1) *Deterministic* which computes the sigmoid probability $p(x) = \text{sigmoid}(\text{logit}(x))$ of a vanilla BERT model (Hendrycks and Gimpel, 2017), (2) *Monte Carlo Dropout* (*MC Dropout*) which estimates uncertainty using the Monte Carlo average of $p(x)$ from 10 dropout samples (Gal and Ghahramani, 2016), (3) *Deep Ensemble* which estimates uncertainty using the ensemble mean of $p(x)$ from 10 BERT models trained in parallel (Lakshminarayanan et al., 2017), (4) *Spectral-normalized Neural Gaussian Process* (*SNGP*), a recent state-of-the-art approach which improves a BERT model's uncertainty quality by transforming it into an approximate Gaussian process model (Liu et al., 2020), and (5) *SNGP Ensemble*, which is the Deep Ensemble using SNGP as the base model.

Learning Objective To address class imbalance, we consider combining the uncertainty methods with *Focal Loss* (Lin et al., 2017). Focal loss reshapes the loss function to down-weight "easy" negatives (i.e. non-toxic examples), thereby focusing training on a smaller set of more difficult examples, and empirically leading to improved predictive and uncertainty calibration performance on class-imbalanced datasets (Lin et al., 2017; Mukhoti et al., 2020). We focus our attention on focal loss (rather than other approaches to class imbalance) because of how this impact on calibration interacts with our moderation review strategies.

6 Benchmark Experiments

We first examine the prediction and calibration performance of the uncertainty models alone (Section 6.1). For prediction, we compute the predictive accuracy (Acc) and the predictive AUC (both AUROC and AUPRC). For uncertainty, we compute the Brier score (i.e., the mean squared error between true labels and predictive probabilities, a standard uncertainty metric), and also the Calibration AUPRC (Section 3).

We then evaluate the models' collaboration performance under both the uncertainty- and the toxicity-based review strategies (Section 6.2). For each model-strategy combination, we measure the model's collaboration ability by computing Review Efficiency, and evaluate the performance of the overall collaborative system using Oracle-Model Collaborative AUROC (OC-AUROC). We evaluate all collaborative metrics over a range of human moderator review capacities, with their review fractions (i.e., fraction of total examples the model sends to the moderator for further review) ranging over $\{0.001, 0.005, 0.01, 0.02, 0.05, 0.1, 0.15, 0.20\}$.

Results on further uncertainty and collaboration metrics (Calibration AUROC, OC-Acc, OC-AUPRC, etc.) are in Appendix D.

6.1 Prediction and Calibration

Table 1 shows the performance of all uncertainty methods evaluated on the testing (the Wikipedia Talk corpus) and the deployment environments (the CivilComments corpus).

First, we compare the uncertainty methods based on the predictive and calibration AUC. As shown, for prediction, the ensemble models (both SNGP Ensemble and Deep Ensemble) provide the best performance, while the SNGP Ensemble and MC Dropout perform best for uncertainty calibration. Training with focal loss systematically improves the model prediction under class imbalance (improving the predictive AUC), while incurring a trade-off with the model's calibration quality (i.e. decreasing the calibration AUC).

Next, we turn to the model performance between the test and deployment environments. Across all methods, we observe a significant drop in predictive performance (~ 0.28 for AUROC and ~ 0.13 for AUPRC), and a less pronounced, but still noticeable drop in uncertainty calibration (~ 0.05 for Calibration AUPRC). Interestingly, focal loss seems to mitigate the drop in predictive performance, but also slightly exacerbates the drop in uncertainty calibration.

Lastly, we observe a counter-intuitive improvement in the non-AUC metrics (i.e., accuracy and Brier score) in the out-of-domain deployment environment. This is likely due to their sensitivity to class imbalance (recall that toxic examples are slightly less rare in CivilComments). As a result, these classic metrics tend to favor model predictions biased toward the negative class, and therefore are less suitable for evaluating model performance in the context of toxic comment moderation.

6.2 Collaboration Performance

Figure 2 and 3 show the Oracle-model Collaborative AUROC (OC-AUROC) of the overall collaborative system, and Figure 4 shows the Review Efficiency of uncertainty models. Both the toxicity-based (dashed line) and uncertainty-based review strategies (solid line) are included.

		Testing Env (Wikipedia Talk)					Deployment Env (CivilComments)				
	Model	AUROC ↑	AUPRC ↑	Acc. ↑	Brier ↓	Calib. AUPRC ↑	AUROC ↑	AUPRC ↑	Acc. ↑	Brier ↓	Calib. AUPRC ↑
XENT	Deterministic	0.9734	0.8019	0.9231	0.0548	0.4053	0.7796	0.6689	0.9628	0.0246	0.3581
	SNGP	0.9741	0.8029	0.9233	0.0548	0.4063	0.7695	0.6665	0.9640	0.0253	0.3660
	MC Dropout	0.9729	0.8006	**0.9274**	**0.0508**	0.4020	0.7806	0.6727	**0.9671**	**0.0241**	**0.3707**
	Deep Ensemble	0.9738	0.8074	0.9231	0.0544	0.4045	**0.7849**	**0.6741**	0.9625	0.0242	0.3484
	SNGP Ensemble	**0.9741**	**0.8045**	0.9226	0.0549	**0.4158**	0.7749	0.6719	0.9633	0.0248	0.3655
Focal	Deterministic	0.9730	0.8036	0.9476	0.0628	0.3804	0.8013	0.6766	**0.9795**	0.0377	0.3018
	SNGP	0.9736	0.8076	0.9455	0.0388	0.3885	0.8003	0.6820	0.9784	**0.0264**	0.3181
	MC Dropout	0.9741	0.8076	0.9472	0.0622	**0.3890**	0.8009	0.6790	0.9790	0.0360	0.3185
	Deep Ensemble	0.9735	0.8077	**0.9479**	0.0639	0.3840	**0.8041**	0.6814	**0.9795**	0.0381	0.3035
	SNGP Ensemble	**0.9742**	**0.8122**	0.9467	**0.0379**	0.3846	0.8002	**0.6827**	0.9790	0.0266	**0.3212**

Table 1: Metrics for models evaluated on the testing environment (the Wikipedia Talk corpus, left) and deployment environment (the CivilComments corpus, right). XENT (top) and Focal (bottom) indicate models trained with cross-entropy and focal losses, respectively. The best metric values for each loss function are shown in bold.

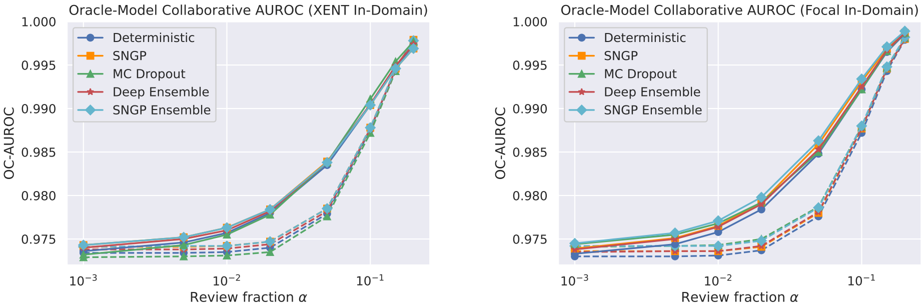

Figure 2: Semilog plot of oracle-model collaborative AUROC as a function of review fraction (the proportion of comments the model can send for human/oracle review), trained with cross-entropy (XENT, left) or focal loss (right) and evaluated on the Wikipedia Talk corpus (i.e., the in-domain testing environment). **Solid line:** uncertainty-based review strategy. **Dashed line:** toxicity-based review strategy. The best performing method is the SNGP Ensemble trained with focal loss and uses the uncertainty-based strategy.

Effect of Review Strategy For the AUC performance of the collaborative system, *the uncertainty-based review strategy consistently outperforms the toxicity-based review strategy*. For example, in the in-domain environment (Wikipedia Talk corpus), using the uncertainty- rather than toxicity-based review strategy yields larger OC-AUROC improvements than any modeling change; this holds across all measured review fractions. We see a similar trend for OC-AUPRC (Appendix Figure 7-8).

The trend in Review Efficiency (Figure 4) provides a more nuanced view to this picture. As shown, the efficiency of the toxicity-based strategy starts to improve as the review fraction increases, leading to a cross-over with the uncertainty-based strategy at high fractions. This is likely caused by the fact that in toxicity classification, the false positive rate exceeds the false negative rate. Therefore sending a large number of positive predictions eventually leads the collaborative system to capture more errors, at the cost of a higher review load on human moderators. We notice that this transition occurs much earlier out-of-domain on CivilComments (Figure 4 right). This highlights the impact of the toxicity distribution of the data on the best review strategy: because the proportion of toxic examples is much lower in CivilComments than in the Wikipedia Talk Corpus, the cross-over between the uncertainty and toxicity review strategies correspondingly occurs at lower review fractions. Finally, it is important to note that this advantage in review efficiency does not directly translate to improvements for the overall system. For example, the OC-AUCs using the toxicity strategy are still lower than those with the uncertainty strategy even for high review fractions.

Effect of Modeling Approach Recall that the performance of the overall collaborative system is the result of the model performance in both prediction and calibration, e.g. Eq. (2). As a result, the model performance in Section 6.1 translates to performance on the collaborative metrics. For example, the ensemble methods (SNGP Ensemble and Deep Ensemble) consistently outperform on the OC-AUC metrics due to their high performance in predictive AUC and decent performance in calibration (Table 1). On the other hand, MC Dropout has

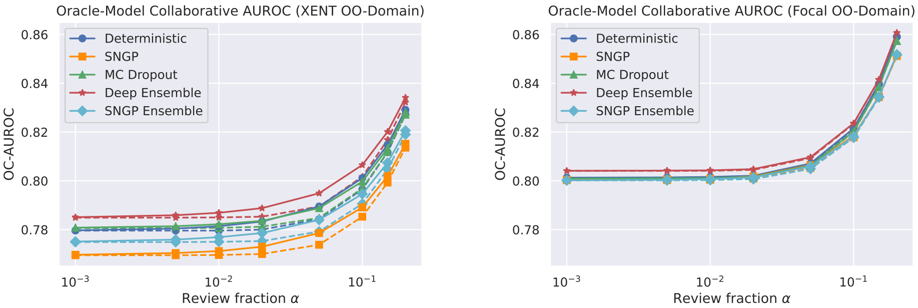

Figure 3: Semilog plot of oracle-model collaborative AUROC as a function of review fraction, trained with cross-entropy (XENT, left) or focal loss (right) and evaluated on CivilComments corpus (i.e., the out-of-domain deployment environment). **Solid line:** uncertainty-based review strategy. **Dashed line:** toxicity-based review strategy. Training with focal rather than cross-entropy loss yields a large improvement. The best performing method is the Deep Ensemble trained with focal loss and uses the uncertainty-based review strategy.

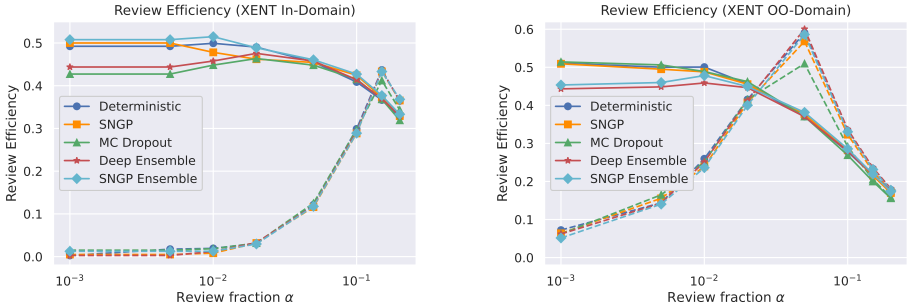

Figure 4: Semilog plot of review efficiency as a function of review fraction, trained with cross-entropy and evaluated on the Wikipedia Talk corpus (i.e., the in-domain testing environment, left) and CivilComments (i.e., the out-of-domain deployment environment, right). **Solid line:** uncertainty-based review strategy. **Dashed line:** toxicity-based review strategy.

good calibration performance but sub-optimal predictive AUC. As a result, it sometimes attains the best Review Efficiency (e.g., Figure 4, right), but never achieves the best overall OC-AUC. Finally, comparing between training objectives, the focal-loss-trained models tend to outperform their cross-entropy-trained counterparts in OC-AUC, due to the fact that focal loss tends to bring significant benefits to the predictive AUC (albeit at a small cost to the calibration performance).

7 Conclusion

In this work, we presented the problem of collaborative content moderation, and introduced *CoToMoD*, a challenging benchmark for evaluating the practical effectiveness of collaborative (model-moderator) content moderation systems. We proposed principled metrics to quantify how effectively a machine learning model and human (e.g. a moderator) can collaborate. These include *Oracle-Model Collaborative Accuracy* (OC-Acc) and *AUC* (OC-AUC), which measure analogues of the usual accuracy or AUC for interacting human-AI systems subject to limited human review capacity. We also proposed *Review Efficiency*, which quantifies how effectively a model utilizes human decisions. These metrics are distinct from classic measures of predictive performance or uncertainty calibration, and enable us to evaluate the performance of the full collaborative system as a function of human attention, as well as to understand how efficiently the collaborative system utilizes human decision-making. Moreover, though we focused here on measuring the combined system's performance through metrics analogous to accuracy and AUC, it is trivial to extend these to other classic metrics like precision and recall.

Using these new metrics, we evaluated the performance of a variety of models on the collaborative content moderation task. We considered two canonical strategies for collaborative review: one based on the toxicity scores, and a new one using model uncertainty. We found that the uncertainty-based review strategy outperforms the toxicity strategy across a variety of models and range of human review capacities, yielding a $>30\%$ absolute increase in how efficiently the model uses human decisions and ~ 0.01 and ~ 0.05 absolute increases in the collaborative system's AUROC and AUPRC, respectively. This merits further study and consideration of this strategy's use in content moderation. The interaction between the data distribution and best review strategy demonstrated by the crossover between the two strategies' performance out-of-domain) emphasizes the implicit trade-off between false positives and false negatives in the two review strategies: because toxicity is rare, prioritizing comments for review in order of toxicity reduces the false positive rate while potentially increasing the false negative rate. By comparison, the uncertainty-based review strategy treats false positives and negatives more evenly. Further study is needed to clarify this interaction. Our work shows that the choice of review strategy drastically changes the collaborative system performance: evaluating and striving to optimize only the model yields much smaller improvements than changing the review strategy, and misses major opportunities to improve the overall system.

Though the results presented in the current paper are encouraging, there remain important challenges for uncertainty modeling in the domain of toxic content moderation. In particular, dataset bias remains a significant issue: statistical correlation between the annotated toxicity labels and various surface-level cues may lead models to learn to overly rely on e.g. lexical or dialectal patterns (Zhou et al., 2021). This could cause the model to produce high-confidence mispredictions for comments containing these cues (e.g., reclaimed words or counter-speech), resulting in a degradation in calibration performance in the deployment environment (cf. Table 1). Surprisingly, the standard debiasing techniques we experimented in this work (specifically, focal loss (Karimi Mahabadi et al., 2020)) only exacerbated this decline in calibration performance. This suggests that naively applying debiasing techniques may incur unexpected negative impacts on other aspects of the moderation system. Further research is needed into modeling approaches that can achieve robust performance both in prediction and in uncertainty calibration under data bias and distributional shift (Nam et al., 2020; Utama et al., 2020; Du et al., 2021; Yaghoobzadeh et al., 2021; Bao et al., 2021; Karimi Mahabadi et al., 2020).

There exist several important directions for future work. One key direction is to develop better review strategies than the ones discussed here: though the uncertainty-based strategy outperforms the toxicity-based one, there may be room for further improvement. Furthermore, constraints on the moderation process may necessitate different review strategies: for example, if content can only be removed with moderator approval, we could experiment with a hybrid strategy which sends a mixture of high toxicity and high uncertainty content for human review. A second direction is to study how these methods perform with real moderators: the experiments in this work are computational and there may exist further challenges in practice. For example, the difficulty of rating a comment can depend on the text itself in unexpected ways. Finally, a linked question is how to communicate uncertainty and different review strategies to moderators: simpler communicable strategies may be preferable to more complex ones with better theoretical performance.

Acknowledgements

The authors would like to thank Jeffrey Sorensen for extensive feedback on the manuscript, and Nitesh Goyal, Aditya Gupta, Luheng He, Balaji Lakshminarayanan, Alyssa Lees, and Jie Ren for helpful comments and discussions.

References

Robert Adragna, Elliot Creager, David Madras, and Richard Zemel. 2020. Fairness and robustness in invariant learning: A case study in toxicity classification. *arXiv preprint arXiv:2011.06485*.

Betty van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 33–42, Brussels, Belgium. Association for Computational Linguistics.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*.

Abdullah Awaysheh, Jeffrey Wilcke, François Elvinger, Loren Rees, Weiguo Fan, and Kurt L. Zimmerman. 2019. Review of Medical Decision Support and Machine-Learning Methods. *Veterinary Pathology*, 56(4):512–525.

Gagan Bansal, Besmira Nushi, Ece Kamar, Eric Horvitz, and Daniel S. Weld. 2021. Is the most accurate ai the best teammate? optimizing ai for team-work. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(13):11405–11414.

Yujia Bao, Shiyu Chang, and Regina Barzilay. 2021. Predict then interpolate: A simple algorithm to learn stable classifiers. In *International Conference on Machine Learning*. PMLR.

Peter L. Bartlett and Marten H. Wegkamp. 2008. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(59):1823–1840.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. *CoRR*, abs/1903.04561.

Justin Cheng, Michael Bernstein, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2017. Anyone can become a troll: Causes of trolling behavior in online discussions. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, page 1217–1230, New York, NY, USA. Association for Computing Machinery.

Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2015. Antisocial behavior in online discussion communities. *Proceedings of the International AAAI Conference on Web and Social Media*, 9(1).

Corinna Cortes, Giulia DeSalvo, Claudio Gentile, Mehryar Mohri, and Scott Yang. 2018. Online learning with abstention. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1059–1067, Stockholmsmässan, Stockholm Sweden. PMLR.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. 2016. Learning with rejection. In *Algorithmic Learning Theory - 27th International Conference, ALT 2016, Proceedings*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pages 67–82. Springer Verlag. 27th International Conference on Algorithmic Learning Theory, ALT 2016 ; Conference date: 19-10-2016 Through 21-10-2016.

Jesse Davis and Mark Goadrich. 2006. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 233–240, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Karthik Dinakar, Roi Reichart, and Henry Lieberman. 2011. Modeling the detection of textual cyberbullying. *Proceedings of the International AAAI Conference on Web and Social Media*, 5(1).

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 67–73, New York, NY, USA. Association for Computing Machinery.

Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. 2021. Towards interpreting and mitigating shortcut learning behavior of NLU models. *CoRR*, abs/2103.06922.

Michael W. Dusenberry, Dustin Tran, Edward Choi, Jonas Kemp, Jeremy Nixon, Ghassen Jerfel, Katherine Heller, and Andrew M. Dai. 2020. Analyzing the role of model uncertainty for electronic health records. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, CHIL '20, pages 204–213, New York, NY, USA. Association for Computing Machinery.

Bassey Etim. 2017. The times sharply increases articles open for comments, using google's technology. *The New York Times*, 13.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *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. PMLR.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Andrew Jesson, Sören Mindermann, Uri Shalit, and Yarin Gal. 2020. Identifying Causal-Effect Inference Failure with Uncertainty-Aware Models. In *Advances in Neural Information Processing Systems*, volume 33, pages 11637–11649. Curran Associates, Inc.

Jigsaw. 2019. How latin america's second largest social platform moderates more than 150k comments a month. `https://medium.com/jigsaw/how-latin-americas-second-largest-social-platform-moderates-more-than-150k-comments-a-month-df0d8a3ac242`. Accessed: 2021-04-26.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2020. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8706–8716, Online. Association for Computational Linguistics.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Sara Beery, et al. 2020. Wilds: A benchmark of in-the-wild distribution shifts. *arXiv preprint arXiv:2012.07421*.

Benjamin Kompa, Jasper Snoek, and Andrew L. Beam. 2021. Second opinion needed: communicating uncertainty in medical machine learning. *npj Digital Medicine*, 4(1):4.

Ranganath Krishnan and Omesh Tickoo. 2020. Improving model calibration with accuracy versus uncertainty optimization. *Advances in Neural Information Processing Systems*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. 2020. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. In *Advances in Neural Information Processing Systems*, volume 33, pages 7498–7512. Curran Associates, Inc.

Jeremiah Liu, John Paisley, Marianthi-Anna Kioumourtzoglou, and Brent Coull. 2019. Accurate uncertainty estimation and decomposition in ensemble learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Diego Mollá and Ben Hutchinson. 2003. Intrinsic versus extrinsic evaluations of parsing systems. In *Proceedings of the EACL 2003 Workshop on Evaluation Initiatives in Natural Language Processing: Are Evaluation Methods, Metrics and Resources Reusable?*, Evalinitiatives '03, page 43–50, USA. Association for Computational Linguistics.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. 2020. Calibrating deep neural networks using focal loss. In *Advances in Neural Information Processing Systems*, volume 33, pages 15288–15299. Curran Associates, Inc.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2901–2907. AAAI Press.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: De-biasing classifier from biased classifier. In *Advances in Neural Information Processing Systems*, volume 33, pages 20673–20684. Curran Associates, Inc.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Lee Rainie, Janna Anderson, and Jonathan Albright. 2017. The Future of Free Speech, Trolls, Anonymity and Fake News Online.

Matthew J. Salganik and Robin C. Lee. 2020. To apply machine learning responsibly, we use it in moderation. `https://open.nytimes.com/to-apply-machine-learning-responsibly-we-use-it-in-moderation-d001f49e0644/`. Accessed: 2021-04-26.

T. J. Sullivan. 2015. *Introduction to uncertainty quantification*. Springer.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards debiasing NLU models from unknown biases. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7597–7610, Online. Association for Computational Linguistics.

Zhao Wang and Aron Culotta. 2020. Identifying spurious correlations for robust text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3431–3440, Online. Association for Computational Linguistics.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 1391–1399, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet des Combes, T. J. Hazen, and Alessandro Sordoni. 2021. Increasing robustness to spurious correlations using forgettable examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3319–3332, Online. Association for Computational Linguistics.

Dawei Yin, Zhenzhen Xue, Liangjie Hong, Brian D Davison, April Kontostathis, and Lynne Edwards. 2009. Detection of harassment on web 2.0. *Proceedings of the Content Analysis in the WEB*, 2:1–7.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah Smith. 2021. Challenges in automated debiasing for toxic language detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3143–3155, Online. Association for Computational Linguistics.

A Details on Metrics

A.1 Expected Calibration Error

For completeness, we include a definition of the expected calibration error (ECE) (Naeini et al., 2015) here. We use the ECE as a comparison for the uncertainty calibration performance alongside the Brier score in the tables in Appendix D.

ECE can be computed by discretizes the probability range $[0, 1]$ into a set of B bins, and computes the weighted average of the difference between confidence (the mean probability within each bin) and the accuracy (the fraction of predictions within each bin that are correct),

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} |\text{conf}(b) - \text{acc}(b)|, \qquad (3)$$

where $\text{acc}(b)$ and $\text{conf}(b)$ denote the accuracy and confidence for bin b, respectively, n_b is the number of examples in bin b, and $N = \sum_b n_b$ is the total number of examples.

A.2 Connection between Calibration AUPRC and Collaboration Metrics

As discussed in Section 3, Calibration AUPRC is an especially suitable metric for measuring model uncertainty in the context of collaborative content moderation, due to its close connection with the intrinsic metrics for the model's collaboration effectiveness.

Specifically, the *Review Efficiency* metric (introduced in Section 4.1) can be understood as the analog of **precision** for the calibration task. To see this, recall the four confusion matrix variables introduced in Figure 1: (1) True Positive (TP) corresponds to the case where the prediction is inaccurate and the model is uncertain, (2) True Negative (TN) to the accurate and certain case, (3) False Negative (FN) to the inaccurate and certain case (i.e., over-confidence), and finally (4) False Positive (FP) to the accurate and uncertain case (i.e., under-confidence).

Then, given a review capacity constraint α, we see that

$$\text{ReviewEfficiency}(\alpha) = \frac{TP_\alpha}{TP_\alpha + FP_\alpha},$$

which measures the proportion of examples that were sent to human moderator that would otherwise be classified incorrectly.

Similarly, we can also define the analog of **recall** for the calibration task, which we term *Review Effectiveness*:

$$\text{ReviewEffectiveness}(\alpha) = \frac{TP_\alpha}{TP_\alpha + FN_\alpha}.$$

Review Effectiveness is also a valid intrinsic metric for the model's collaboration effectivess. It measures the proportion of incorrect model predictions that were successfully corrected using the review strategy. (We visualize model performance in Review Effectiveness in Section D.)

To this end, the calibration AUPRC can be understood as the area under the Review Efficiency v.s. Review Effectiveness curve, with the usual classification threshold replaced by the review capacity α. Therefore, calibration AUPRC serves as a threshold-agnostic metric that captures the model's intrinsic performance in collaboration effectiveness.

A.3 Further Discussion

For the uncertainty-based review, an important question is whether classic uncertainty metrics like Brier score capture good model-moderator collaborative efficiency. The SNGP Ensemble's good performance contrasts with its poorer Brier score (Table 1). By comparison, the calibration AUPRC successfully captures this good performance, and is highest for that model. More generally, the low-review fraction review efficiency with cross-entropy is exactly captured by the calibration AUPRC (same ordering for the two measures). This correspondence is not perfect: though the SNGP Ensemble with focal loss has the highest review efficiency overall, its calibration AUPRC is lower than the MC Dropout or SNGP models (models with next highest review efficiencies). This may reflect the reshaping effect of focal loss on SNGP's calibration (explored in Appendix C). Overall, calibration AUPRC much better captures the relationship between collaborative ability and calibration than do classic calibration metrics like Brier score (or ECE, see Appendix D). This is because classic calibration metrics are population-level averages, whereas calibration AUPRC measures the ranking of the predictions, and is thus more closely linked to the review order problem.

B Connecting Review Efficiency and Collaborative Accuracy

In this appendix, we derive Eq. (2) from the main paper, which connects the Review Efficiency and Oracle-Collaborative Accuracy.

Given a trained toxicity model, a review policy and a dataset, let us denote r as the event that an example gets reviewed, and c as the event that model prediction is correct. Now, assuming the model sends $\alpha \times 100\%$ of examples for human review, we have:

$$\text{Acc} = P(c), \qquad \alpha = P(r).$$

Also, we can write:

$$\text{RE}(\alpha) = P(\neg c | r)$$

i.e., review efficiency RE(α) is the percentage of incorrect predictions among reviewed examples. Finally:

$$\text{OC-Acc}(\alpha) = P(c \cap \neg r) + P(c \cap r) + P(\neg c \cap r)$$

i.e., an example is predicted correctly by the collaborative system if either the model prediction itself is accurate ($c \cap \neg r$), or it was sent for human review ($c \cap r$ or $\neg c \cap r$).

The above expression of OC-Acc leads to two different decompositions of the OC-Acc. First,

$$\begin{aligned}\text{OC-Acc}(\alpha) &= P(c \cap \neg r) + P(r)\\ &= P(c|\neg r)P(\neg r) + P(r)\\ &= \text{Acc}(1-\alpha) * (1-\alpha) + \alpha,\end{aligned}$$

where Acc$(1-\alpha)$ is the accuracy among the $(1-\alpha) \times 100\%$ examples that are not sent to human for review.

Alternatively, we can write

$$\begin{aligned}\text{OC-Acc}(\alpha) &= P(c) + P(\neg c \cap r)\\ &= P(c) + P(\neg c|r)P(r)\\ &= \text{Acc} + \text{RE}(\alpha) * \alpha,\end{aligned}$$

which coincides with the expression in Eq. (2).

C Reliability Diagrams for Deterministic and SNGP models

We study the effect of focal loss on calibration quality for SNGP in further detail. We plot the reliability diagrams for the deterministic and SNGP models trained with cross-entropy and focal cross-entropy. Figure 5 shows the reliability diagrams in-domain and Figure 6 shows them out-of-domain. We see that focal loss fundamentally changes the models' uncertainty behavior, systematically shifting the uncertainty curves from overconfidence (the lower right, below the diagonal) and toward the calibration line (the diagonal). However, the exact pattern of change is model dependent. We find that the deterministic model with focal loss is over-confident for predictions under 0.5, and under-confident above 0.5, while the SNGP models are still over-confident, although to a lesser degree compared to using cross-entropy loss.

D Complete metric results

We give the results for the remaining collaborative metrics not included in the main paper in this appendix. These give a comprehensive summary of the collaborative performance of the models evaluated in the paper. Table 2 and Table 3 give values for all review fraction-independent metrics, both in- and out-of-domain, respectively. We did not include the ECE and calibration AUROC in the corresponding table in the main paper (Table 1) for simplicity. Similarly, Figures 9 and 7 show the in-domain results (the OC-Acc and OC-AUPRC), and the out-of-domain plots (in the same order, followed by Review Efficiency) are Figures 10 through 12.

The in- and out-of-domain OC-AUROC figures are included in the main paper as Figure 2 and Figure 3, respectively; the in-domain Review Efficiency is Figure 4. Additionally, we also report results on the Review Effectiveness metric (introduced in Section A.2) in Figures 13-14. Similiar to Review Efficiency, we find little difference in performance between different uncertainty models, and that the uncertainty-based policy outperforms toxicity-based policy especially in the low review capacity setting.

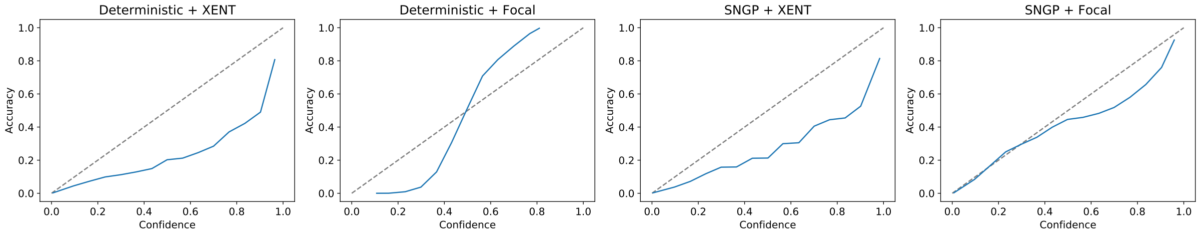

Figure 5: In-domain reliability diagrams for deterministic models and SNGP models with cross-entropy (XENT) and focal cross-entropy.

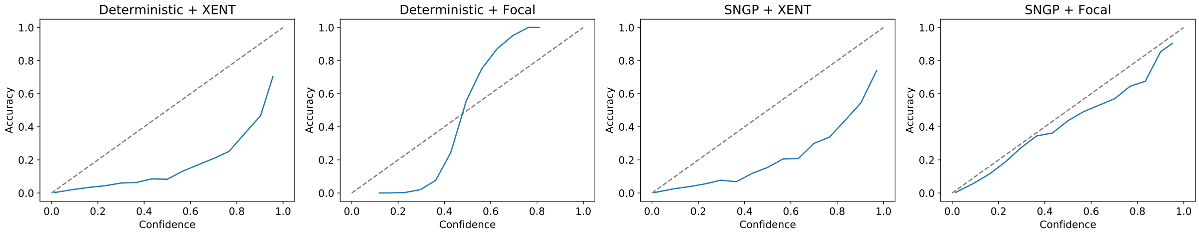

Figure 6: Reliability diagrams for deterministic models and SNGP models with cross-entropy (XENT) and focal cross-entropy on the CivilComments dataset.

	Model (Test)	AUROC ↑	AUPRC ↑	Acc. ↑	ECE ↓	Brier ↓	Calib. AUROC ↑	Calib. AUPRC ↑
XENT	Deterministic	0.9734	0.8019	0.9231	0.0245	0.0548	0.9230	0.4053
	SNGP	0.9741	0.8029	0.9233	0.0280	0.0548	0.9238	0.4063
	MC Dropout	0.9729	0.8006	**0.9274**	**0.0198**	**0.0508**	**0.9282**	0.4020
	Deep Ensemble	0.9738	**0.8074**	0.9231	0.0235	0.0544	0.9245	0.4045
	SNGP Ensemble	**0.9741**	0.8045	0.9226	0.0281	0.0549	0.9249	**0.4158**
Focal	Deterministic	0.9730	0.8036	0.9476	0.1486	0.0628	0.9405	0.3804
	SNGP	0.9736	0.8076	0.9455	0.0076	0.0388	0.9385	0.3885
	MC Dropout	0.9741	0.8076	0.9472	0.1442	0.0622	**0.9425**	**0.3890**
	Deep Ensemble	0.9735	0.8077	**0.9479**	0.1536	0.0639	0.9418	0.3840
	SNGP Ensemble	**0.9742**	**0.8122**	0.9467	**0.0075**	**0.0379**	0.9400	0.3846

Table 2: Metrics for models on the Wikipedia Talk corpus (in-domain testing environment), all numbers are averaged over 10 model runs. XENT and Focal indicate models trained with the cross-entropy and focal losses, respectively. The best metric values for each loss function are shown in bold.

	Model (Deployment)	AUROC ↑	AUPRC ↑	Acc. ↑	ECE ↓	Brier ↓	Calib. AUROC ↑	Calib. AUPRC ↑
XENT	Deterministic	0.7796	0.6689	0.9628	0.0128	0.0246	0.9412	0.3581
	SNGP	0.7695	0.6665	0.9640	**0.0070**	0.0253	0.9457	0.3660
	MC Dropout	0.7806	0.6727	**0.9671**	0.0136	**0.0241**	**0.9502**	**0.3707**
	Deep Ensemble	**0.7849**	**0.6741**	0.9625	0.0141	0.0242	0.9420	0.3484
	SNGP Ensemble	0.7749	0.6719	0.9633	0.0076	0.0248	0.9463	0.3655
Focal	Deterministic	0.8013	0.6766	**0.9795**	0.1973	0.0377	0.9444	0.3018
	SNGP	0.8003	0.6820	0.9784	0.0182	**0.0264**	0.9465	0.3181
	MC Dropout	0.8009	0.6790	0.9790	0.1896	0.0360	**0.9481**	0.3185
	Deep Ensemble	**0.8041**	0.6814	**0.9795**	0.1998	0.0381	0.9461	0.3035
	SNGP Ensemble	0.8002	**0.6827**	0.9790	**0.0176**	0.0266	**0.9481**	**0.3212**

Table 3: Metrics for models on the CivilComments corpus (out-of-domain deployment environment), all numbers are averaged over 10 model runs. XENT and Focal indicate models trained with the cross-entropy and focal losses, respectively. The best metric values for each loss function are shown in bold.

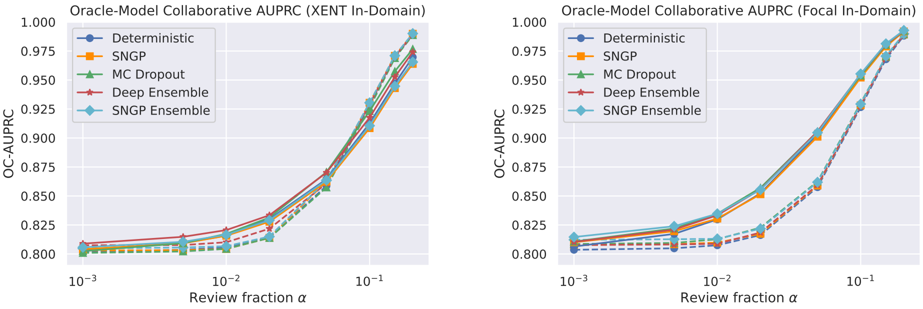

Figure 7: Oracle-model collaborative AUPRC as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on Wikipedia Toxicity corpus (in-domain test environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. Overall, the SNGP Ensemble with focal loss using the uncertainty review performs best across all α. Restricted to cross-entropy loss, the Deep Ensemble using uncertainty-based review performs best until $\alpha \approx 0.1$, when some of the toxicity-based reviews (e.g. SNGP Ensemble) begin to outperform it.

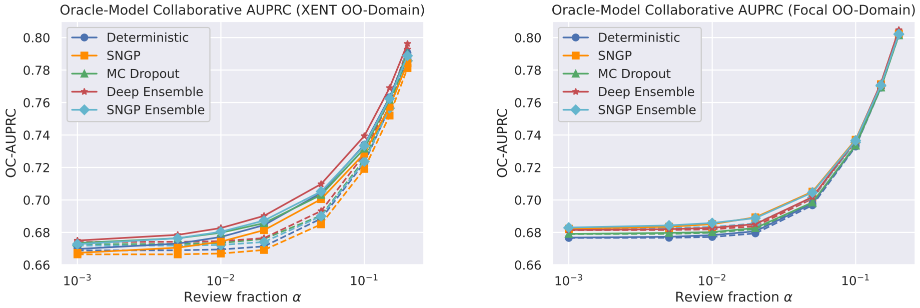

Figure 8: Oracle-model collaborative AUPRC as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on CivilComments corpus (out-of-domain deployment environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. Similar to the out-of-domain OC-AUROC results in Figure 3, of the models trained with cross-entropy loss the Deep Ensemble performs best. Training with focal loss yields a small baseline improvement, but surprisingly results in the SNGP Ensemble performing best. The uncertainty-based review strategy uniformly outperforms toxicity-based review, though the difference is small when training with focal loss.

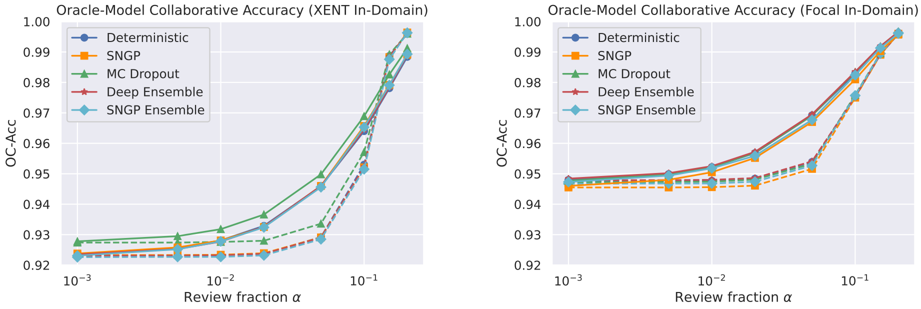

Figure 9: Oracle-model collaborative accuracy as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on Wikipedia Toxicity corpus (in-domain test environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. Focal loss yields a significant improvement, equivalent to using a 10% review fraction with cross-entropy. For most review fractions (below $\alpha = 0.1$), MC Dropout using the uncertainty review strategy performs trained with cross-entropy, while overall the Deep Ensemble with focal loss (again using the uncertainty review) performs best. For large review fractions ($\alpha > 0.1$), the toxicity-based review in fact outperforms the uncertainty review.

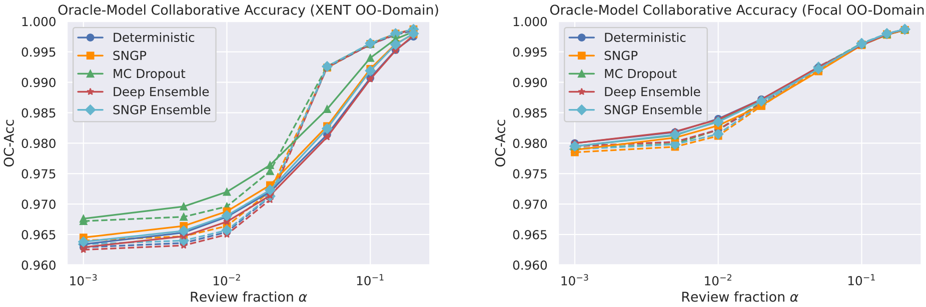

Figure 10: Oracle-model collaborative accuracy as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on CivilComments corpus (out-of-domain deployment environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. Training with cross-entropy, MC Dropout using uncertainty-based review performs best until the SNGP Ensemble using the toxicity-based review overtakes it at $\alpha = 0.05$. Training with focal loss gives significant baseline improvements (by mitigating the class imbalance problem); the Deep Ensemble is best for small α while the SNGP Ensemble is best for large α. Despite these baseline improvements, they appear to come at a cost of collaborative accuracy in the intermediate region around $\alpha \approx 0.05$, where the SNGP Ensemble trained with cross-entropy briefly performs best overall, apart from that region the models with focal loss and the uncertainty-based review perform best (Deep Ensemble for $\alpha \leq 0.02$, SNGP Ensemble for $\alpha \geq 0.1$).

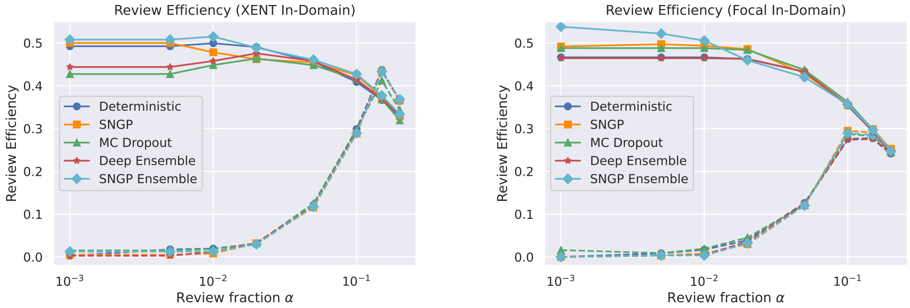

Figure 11: Review efficiency as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on Wikipedia Toxicity corpus (in-domain test environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. This is the only plot for which we observe a major crossover: training with cross-entropy, the efficiency for toxicity-based review spikes above the uncertainty-based review efficiency at $\alpha = 0.02$ before converging back toward it with increasing α. There is no corresponding crossover when training with focal loss; rather, the efficiencies of the two strategies converge at $\alpha = 0.02$ instead.

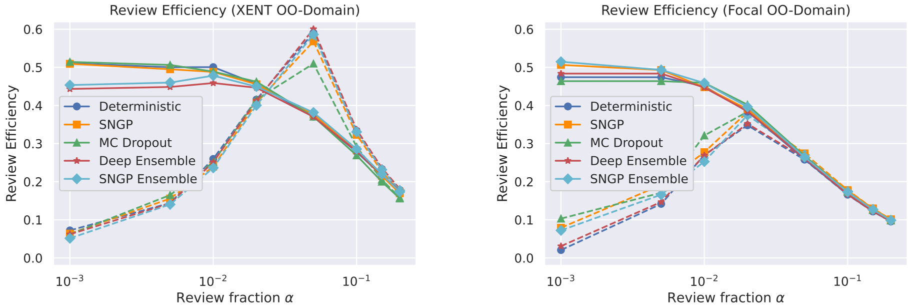

Figure 12: Review efficiency as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on CivilComments corpus (out-of-domain deployment environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. This is the only plot for which we observe a major crossover: training with cross-entropy, the efficiency for toxicity-based review spikes above the uncertainty-based review efficiency at $\alpha = 0.02$ before converging back toward it with increasing α. There is no corresponding crossover when training with focal loss; rather, the efficiencies of the two strategies converge at $\alpha = 0.02$ instead.

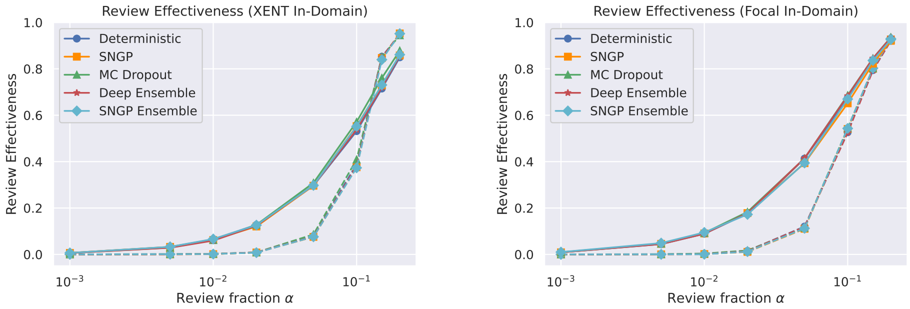

Figure 13: Review effectiveness as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on Wikipedia Toxicity corpus (in-domain test environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. There is little difference between models here: the uncertainty-based review strategy successfully catches more incorrect model decisions until $\alpha \approx 0.15$.

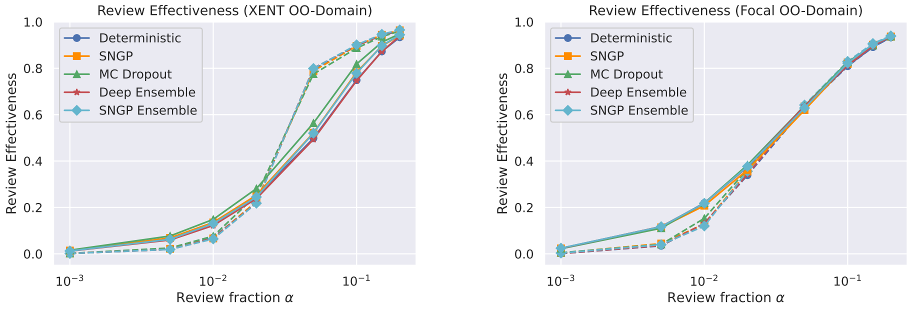

Figure 14: Review effectiveness as a function of review fraction, trained with cross-entropy (left) or focal loss (right) and evaluated on CivilComments corpus (out-of-domain deployment environment). **Solid Line**: uncertainty-based strategy. **Dashed Line**: toxicity-based strategy. Here, the uncertainty review performs better until a crossover at $\alpha \approx 0.02$, much lower than in Figure 4. The SNGP Ensemble performs best with either cross-entropy or focal loss (slightly better with cross-entropy).