Bandits for Online Calibration: An Application to Content Moderation on Social Media Platforms

Vashist Avadhanula[◇,1], **Omar Abdul Baki**[◇], **Hamsa Bastani**[◇,†,2], **Osbert Bastani**[◇,†,3], **Caner Gocmen**[◇], **Daniel Haimovich**[◇], **Darren Hwang**[◇], **Dima Karamshuk**[◇] , **Thomas Leeper**[◇], **Jiayuan Ma**[◇], **Gregory Macnamara**[◇], **Jake Mullett**[◇], **Christopher Palow**[◇], **Sung Park**[◇], **Varun S Rajagopal**[◇], **Kevin Schaeffer**[◇], **Parikshit Shah**[◇], **Deeksha Sinha**[◇], **Nicolas Stier-Moses**[◇], **Peng Xu** [◇]

[◇]Meta, [†]University of Pennsylvania

Abstract

We describe the current content moderation strategy employed by Meta to remove policy-violating content from its platforms. Meta relies on both handcrafted and learned risk models to flag potentially violating content for human review. Our approach aggregates these risk models into a single ranking score, calibrating them to prioritize more reliable risk models. A key challenge is that violation trends change over time, affecting which risk models are most reliable. Our system additionally handles production challenges such as changing risk models and novel risk models. We use a contextual bandit to update the calibration in response to such trends. Our approach increases Meta's top-line metric for measuring the effectiveness of its content moderation strategy by 13%.

1 Introduction

Meta has nearly 3.71 billion (monthly) active users worldwide, with billions of pieces of content shared by users every day [Meta, 2022c]. While most content is benign, a small share—e.g., hate speech, promoting terrorism, or graphic pornography—violates the platform's community standards. Thus, a key goal is to promptly remove such content. Determining whether a piece of content is policy-violating is a difficult decision that often requires manual review, making it challenging to scale to Meta's prolific content streams. As of December 2021, over 2 million pieces of content are reviewed per day by around 15,000 human reviewers across the globe [Meta, 2022b]. Manually reviewing all content is infeasible; thus, Meta relies heavily on a combination of risk models (based on machine learning or handcrafted rules) to flag potentially violating content pieces. A subset of this content is deemed unambiguously violating and is automatically removed; the remainder undergoes manual review.

To maximize the amount of violating content removed given a fixed supply of human reviewers, we dynamically prioritize content likely to have wide reach and high severity (quantified by a metric called *integrity value*; see Section 2). Initial efforts designed separate risk scores for different content violation types (e.g., one for violations of the nudity policy, one for hate speech, etc.), allocating a fixed, pre-determined reviewer capacity to each type [Schroepfer, 2019]. However, this strategy was unable to adapt to the nonstationary and quickly varying violation trends—i.e., violating content constantly changes in appearance, focus, and wording. Exacerbating this issue, users trying to post violating content are adversarial, attempting to evade detection. Thus, the performances of the different risk models are constantly changing. Models require significant expertise and effort to retrain and can only be updated periodically, and new risk models are frequently added for emerging trends—e.g., to capture the use of emojis in racist comments directed at black English football players [Criddle, 2021].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Thus, Meta moved to a single holistic ranker "Whole Post Integrity Embeddings" (WPIE), a pretrained universal representation of content across modalities and violation types [Schroepfer, 2019, Halevy et al., 2022]. It resulted in significant performance improvements in production [Schroepfer, 2019, Rosen, 2019], and is the baseline our work improves upon. An important drawback is that, although more accurate overall, it lacked precision for specific types of violations compared to the risk model tailored to that type. Furthermore, while it was periodically retrained to handle evolving trends, it did not incorporate sufficient exploration of new violation types to ensure sufficient responsiveness to new trends.

To address these challenges, we have designed and deployed a bandit approach that combines the full set of available risk models into a single risk score; importantly, it *calibrates* the scores to prioritize ones with higher quality, doing so *dynamically* to respond to new trends. A key challenge is *bandit feedback*: we only observe the true severity of a piece of content when it is reviewed by a human. Thus, we must actively explore different types of content to obtain ground truth data on the accuracy of different risk models. Finally, the scale of the problem imposes constraints on the techniques we can leverage—we must ensure that our system can assess a large number of content every minute for potential violations under computational cost and latency constraints.

In detail, we use a nonstationary, batched contextual bandit that treats the various risk models as features for predicting severity. We address two key production-related challenges that differ from existing approaches. First, for a given piece of content, our prediction of its reach (and therefore potential prevalence) evolves over time as it is viewed and shared by users. However, traditional bandit algorithms assume that features are static. We incorporate the bandit into a queuing framework, where the queue priority is determined by both the predicted severity as well as the velocity of its predicted reach. Second, we must incorporate new risk models (i.e., features) seamlessly despite the fact that we cannot retrain our severity predictions online due to production constraints. Together, our approach effectively learns an estimate of severity that dynamically adapts to changes in the environment despite production constraints. Our approach has been deployed at scale at Meta, increasing the top-line metric of Integrity Value (which quantifies the impact of removing violating content accounting for severity and potential reach) by 13% compared to the previous WPIE approach.

2 Problem Formulation

Content arrival process. We assume content arrives sequentially. At each step t, content c_t arrives, and we observe its features $x_t = \phi(c_t) \in \mathbb{R}^d$; each feature is the output of a model that predicts the real-valued risk of c_t for a given violation type (e.g., hate speech). These features are built by various teams at Meta, and range from machine learning algorithms trained on reviewers' historical labels, to simple classifiers based on regular expressions to flag violating phrases. Each piece of content receives d predicted risk scores (from each of these models), which are then concatenated to form x_t.

Manual review decisions. Next, our system must decide whether to have a reviewer label c_t. We formalize this process as a one-armed bandit [Woodroofe, 1979], where the arm is the action

$$a_t = \mathbb{1}(\text{mark } c_t \text{ for manual review}),$$

and $\mathbb{1}$ is the indicator function. That is, pulling the arm corresponds to having a reviewer manually examine c_t, and not pulling it corresponds to leaving the content up without review.

Objective. There are several desiderata when assessing the risk of violating content: we wish to prioritize content that (i) has a high likelihood of violation, (ii) with more severe violations (e.g., terrorism or child nudity), and (iii) is likely to receive a large number of views. We focus on *Integrity Value (IV)*, which quantifies the value of taking down violating content; at a high level, IV of a piece of content has the form

$$IV = (\text{predicted future views} + \text{constant}) \times (\text{severity}).$$

The additive constant is tuned to sufficiently prioritize nonviral but violating content with high severity. Future views are predicted dynamically using past viewership trajectories of similar content by similar users. We fit a Hawkes process, which is effective at capturing "self-exciting" phenomena such as viral content on social media [Haimovich et al., 2022]. Severity is a real-valued function that maps every possible policy violation type to a non-negative real number. Once a piece of content is flagged for human review (i.e. $a_t = 1$), a human reviewer determines its severity y_t. We define

$y_t = 0$ for non-violating content. If $y_t > 0$, then we remove the underlying content and reward the underlying risk model.The system level IV is defined as the sum of the IV of all the content sent for human review.

Practical challenges. We summarize several practical challenges; see Section 4 for details. First, we must handle *content lifetime*—rather than make an immediate decision, we can defer the decision to a future time step; however, there is a cost of leaving violating content up for a longer period of time. Second, since the algorithm must run in real-time on a huge volume of content, it cannot involve overly complex computations. Third, we must handle *nonstationarity*—violating content changes form over time, meaning older data may not be representative of the current content.

3 Bandit Algorithm

Risk prediction model. We describe our algorithm for deciding whether to flag each c_t for review. Naïvely, we could flag c_t if any of its risk scores $x_{t,i}$ are positive—i.e., construct an overall risk score $\hat{y}_t = \max_i x_{t,i}$ and take $a_t = \mathbb{1}(\hat{y}_t > 0)$ (our implementation uses the predicted risk scores to prioritize content rather than making isolated decisions; see Section 4). However, different risk scores may not be directly comparable. Thus, our algorithm rescales the different risk scores according to some parametric function f_β, which we refer to as *calibrating* the set of risk models:

$$\hat{y}_t = \max_i f_{\beta_i}(x_{t,i}),$$

where $\beta_i \in \mathbb{R}^k$ are rescaling parameters for risk model i. We choose f_β to be piecewise linear:

$$f_{\beta_i}(z) = \sum_{j=1}^{k} \mathbb{1}(z \in B_j)\beta_{i,j} z \tag{1}$$

for each i, where $B_j \subseteq \mathbb{R}$ are bins over the space of risk scores. For simplicity, we have assumed that the bins are identical across different risk scores i, but in our implementation, they actually depend on i; roughly speaking, they are chosen based on quantiles of the observed scores $\{x_{t,i}\}_t$. Our goal is to learn the unknown calibration parameters $\{\beta_i\}$ to make effective labeling decisions.

Parameter estimation. Next, we describe how to estimate β_i given a fixed dataset $\{(x_t, y_t)\}_t$. We can estimate $\beta_{i,j}$ for each risk score i and bin B_j independently, since it is a linear model:

$$y_t = \beta_{i,j} x_{t,j} + \epsilon_{t,i,j}, \tag{2}$$

for some σ-subgaussian noise term $\epsilon_{t,i,j}$. Thus, we can estimate $\beta_{i,j}$ using linear regression. In addition, our implementation uses a heuristic where for the parameters β_i for risk score i, we only train on a top α quantile of examples $\{x_{t,i}\}_t$ in terms of magnitude, where α is a hyperparameter. Importantly, this strategy only depends on the relative magnitude of risk scores given by scoring function i, so it is not affected by the fact that different risk scores $x_{t,i}$ and $x_{t,i'}$ are incomparable.

Upper confidence bounds. A key challenge is *bandit feedback*: we only observe the true severity y_t for content c_t when $a_t = 1$. Thus, we must actively explore different types of content to obtain labels to estimate β_i. Intuitively, we mark content for review either when its label can provide information towards better estimating some β_i (exploration), or when it is likley to be violating (exploitation). In particular, we use UCB [Auer and Ortner, 2010], which chooses arms based on an *optimistic* estimate of its rewards. It maintains both a point estimate $\hat{\beta}_{i,j}$ of $\beta_{i,j}$, and an upper confidence bound for this estimate:

$$\mathbb{P}[\hat{\beta}_{i,j} + u_{i,j} \geq \beta_{i,j}] \geq 1 - \delta, \tag{3}$$

where $\beta_{i,j}$ are the "true" parameters, $\delta \in (0, 1)$ is a confidence level, and the probability is taken with respect to the randomness in the examples used to train $\hat{\beta}_{i,j}$. For our model, we can take

$$u_{ij} = \sigma_{i,j} \cdot \sqrt{\frac{\log(1/\delta)}{\sum_t x_{t,i}^2 \mathbb{1}(x_{t,i} \in B_j)}}. \tag{4}$$

Here, the noise variance $\sigma_{i,j}$ is unknown; we estimate it as the empirical standard error of our linear regression model. Also, as we rely on a fixed window of training data, we use a fixed choice of δ (whereas standard bandit algorithms reduce δ over time). Finally, for content with features x_i, we *optimistically* estimate its severity to be $\hat{y}_t = \max_i f_{\hat{\beta}_i + u_i}(x_{t,i})$.

4 Implementation Challenges

Content lifetime. In practice, content persists on the platform for an extended period of time. Thus, we can revisit negative decisions $a_t = 0$ at future steps—e.g., reviewers may become less busy, making it worthwhile to review the content; alternatively, the IV of the content may increase since it scales with viewership, which is time-varying. Rather than make a binary decision, our algorithm instead maintains a pool C_t of all currently available content on time step t. In this formulation, time steps correspond to events where a reviewer becomes available (rather than where a new content arrives on the platform)—e.g., multiple pieces of content may be added and/or removed from C_{t-1} to obtain C_t. Also, note that the action space is now the content c_t^* to review on step t rather than whether to review content c_t. Naïvely, we could review content with highest predicted severity/IV:

$$c_t^* = \max_{c \in C_t} \{\max_i f_{\hat{\beta}_i + u_i}(\phi(c))\}.$$

However, a key insight from the literature on job scheduling is that the optimal policy allocates jobs with the highest *rate of change* first, known as the $c\mu$ *rule* [Mandelbaum and Stolyar, 2004]. Adapting this rule to our setting, we prioritize content based on its estimated *rate of change* in IV.

Real-time parameter updates. Meta evaluates a very large volume of content for potential violations every minute. To ensure that we can dynamically adapt to new violation trends, we update our parameter estimates once every 5 minutes. For scalability, we use online updates to our parameter estimates rather than re-computing them from scratch; see Appendix A for details.

Nonstationarity. A key challenge is that violation trends are highly non-stationary, since users may learn to evade detection, and since Meta regularly updates its community standards. Thus, we exponentially downweight older content: given a discount factor γ, we weight training examples by γ^τ, where τ is the number of hours since the content arrived; see Appendix A for details.

Adding new features. Another way to address nonstationarity is to add new features (i.e., risk models). One example is the recent use of emojis in a wave of racist comments directed at black football players immediately after the UEFA European Football Championship final [Criddle, 2021]. Here, the key indicator of violating content was the use of a particular set of derogatory emojis (which were benign outside this context). Retraining existing risk models to identify such new trends is often time-consuming, slowing down our response to fast-moving violation trends. Instead, we quickly deployed a specialized risk model that flags any content including both these emojis and football discussion, combining it with sentiment analysis to estimate the violation likelihood. We have developed infrastructure to quickly launch such handcrafted rankers. In particular, our bandit algorithm quickly learns the effectiveness of this new risk model via exploration. If a violation trend is short-lived, it will also quickly learn that the new risk model's utility has decreased, and will downweight it accordingly. Subsequently, we have observed substantial improvements in the turnaround time for flagging and removing new variants of violating content.

Interpretability. A key advantage of our approach is interpretability: we can explain why certain piece of content is flagged for review by pointing to the risk model responsible for high severity. This ability helps debug issues when there are unexpected increases in certain types of violating content.

5 Deployment

Our approach has been deployed at scale at Meta. For estimating the added value, internal A/B tests were run between 9th August, 2021 and 20th August, 2021 with 1,184,526 and 963,908 jobs in the control and test groups respectively. These tests have demonstrated a statistically significant 13% (0.9%, 25.2%) lift in IV (with a fixed capacity of human reviewers) compared to the WPIE (The values in the brackets indicate 95% confidence intervals). To maintain the same IV as our bandit, the old approach would need approximately 780,000 additional people-hours per year of human reviewer capacity. Additional information about the deployment is given in Appendix B.

6 Conclusion

We have described our system deployed at Meta for identifying and removing violating content from the platform. Our system employs a bandit algorithm to dynamically adjust calibration parameters

across a set of risk models. In an internal A/B test, our system outperformed the existing approach, WPIE, by over 13% in terms of IV with a fixed capacity of human reviewers. As of December 2021, our system flags over 2 million pieces of content for review per day by over 15,000 human reviewers across the globe [Meta, 2022a]. Our work demonstrates that bandit algorithms are a promising strategy for addressing issues in deploying machine learning systems in highly nonstationary environments.

References

Peter Auer and Ronald Ortner. Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

Alberto Bietti, Alekh Agarwal, and John Langford. Practical evaluation and optimization of contextual bandit algorithms. *Statistics*, 1050:12, 2018.

Cristina Criddle. Instagram admits moderation mistake over racist comments. *BBC*, 2021. URL `https://www.bbc.com/news/technology-57848106`.

Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. In *ICML*, pages 1097–1104, 2011.

Daniel Haimovich, Dima Karamshuk, Thomas J Leeper, Evgeniy Riabenko, and Milan Vojnovic. Popularity prediction for social media over arbitrary time horizons. In *VLDB 2022: International Conference on Very Large Data Bases*, 2022.

Alon Halevy, Cristian Canton-Ferrer, Hao Ma, Umut Ozertem, Patrick Pantel, Marzieh Saeidi, Fabrizio Silvestri, and Ves Stoyanov. Preserving integrity in online social networks. *Communications of the ACM*, 65(2):92–98, 2022.

Avishai Mandelbaum and Alexander L. Stolyar. Scheduling flexible servers with convex delay costs: Heavy-traffic optimality of the generalized cμ – rule. *Operations Research*, 52(6):836–855, 2004.

Meta. Adult nudity and sexual activity. 2022a. URL `https://transparency.fb.com/policies/community-standards/adult-nudity-sexual-activity`.

Meta. Facebook community standards. 2022b. URL `https://transparency.fb.com/policies/community-standards/spam/`.

Meta. Meta q3 2022 earnings report. 2022c. URL `https://investor.fb.com/investor-events/event-details/2022/Q3-2022-Earnings/default.aspx`.

Guy Rosen. Community standards enforcement report, november 2019 edition. 2019. URL `https://about.fb.com/news/2019/11/community-standards-enforcement-report-nov-2019/`.

Mike Schroepfer. Community standards report. 2019. URL `https://ai.facebook.com/blog/community-standards-report/`.

Michael Woodroofe. A one-armed bandit problem with a concomitant variable. *Journal of the American Statistical Association*, 74(368):799–806, 1979.

A Additional Discussion

Real-time parameter updates. Let $XY_{ij}^t = \sum_t y_t x_{ti} \mathbb{1}(x_{ti} \in B_j)$ and $XX_{ij}^t = \sum_t x_{ti}^2 \mathbb{1}(x_{ti} \in B_j)$ be the running numerators and denominators computed until time t in (2). We can compute the terms XY_{ij}^t and XX_{ij}^t online as follows:

$$XY_{ij}^{t+1} = XY_{ij}^t + x_{t+1,i} y_{t+1}, \qquad XX_{ij}^{t+1} = XX_{ij}^t + x_{t+1,i}^2. \tag{5}$$

Therefore, β_{ij}^{t+1} can be computed online as the ratio of XY_{ij}^{t+1} and XX_{ij}^{t+1}. Next, let $N_{ij}^t = \sum_{\tau<t} \mathbb{1}(x_{\tau,i} \in B_j)$ be the number of labeled content pieces until time t by ranker i that belonged to bin j. Then, we can compute the standard errors online as follows:

$$(\sigma_{ij}^{t+1})^2 = \frac{N_{ij}^t}{N_{ij}^{t+1}} \Big((\sigma_{ij}^t)^2 + y_{t+1}^2 + (\beta_{ij}^t)^2 XX_{ij}^t - (\beta_{ij}^{t+1})^2 XX_{ij}^{t+1} \Big)$$

This strategy reduces the number of queries to our database from tens of thousands to a single look-up.

Nonstationarity. We compute the reweighted variants of XX_{ij}^t and XY_{ij}^t as follows:

$$XY_{ij}^t = \sum_\tau \gamma^{t-\tau} x_{\tau,i} y_\tau, \qquad XX_{ij}^t \quad = \sum_\tau \gamma^{t-\tau} x^2 \tau, i.$$

Similarly, we compute σ_{ij}^t and $u_i j$ as follows:

$$\sigma_{ij}^t = \sqrt{\frac{\sum_\tau \gamma^{2t-\tau}(y_\tau - \hat{\beta}_{ij}^t x_{\tau,i})^2}{\sum_\tau \gamma^{t-\tau} \mathbb{1}(x_{\tau,i} \in B_j)}}, \qquad u_{ij}^t = \sigma_{ij}^t \cdot \sqrt{\frac{\log(1/\delta)}{XX_{ij}^t}}.$$

Finally, we altogether remove content beyond a window $\tau \geq \tau_{\max}$.

B Additional Deployment Information

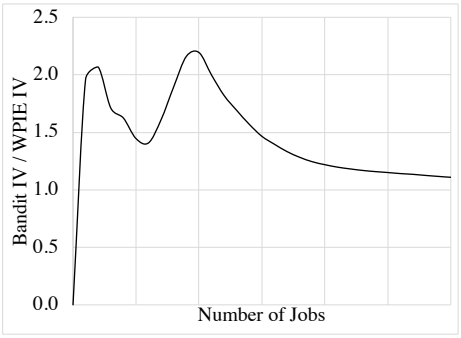

Figure 1: Relative performance of the Centralized Bandit approach

One challenge for the bandit deployment was how to choose hyperparameters, which can significantly affect performance [Bietti et al., 2018]. To do so, we developed a simulator based on a rolling 30 days of historical data. Compared to standard off-policy evaluation methods [Dudík et al., 2011] that evaluate a *static* policy, our simulator provides insight into the *dynamic* performance of different policies as they adapt to changing data. We chose UCB (which outperformed Thompson Sampling) based on the simulator, as well as δ (the confidence parameter) and γ (the discount factor). Further, using simulations we established the significant IV gains that our approach would have as compared to the existing approach in production as shown in Figure 1 (exact scale of number of jobs has been anonymized) .

The A/B test was run across multiple language based markets including Indonesian, Turkish, Portuguese, German, Thai, Korean, Mexican, Hindi, Japanese, Romanian, Dari, Filipino, Urdu, Burmese, Pashto, Hebrew and region based markets including Australia and North America. There were three markets we removed from the test midway due to world events (Dari, Pashto and Urdu). Results in these markets were hence excluded.

In these tests, the bandit approach was compared to the existing approach used in production before ours, called *Whole Post Integrity Embeddings (WPIE)* [Schroepfer, 2019]. WPIE uses a single ranker across different violation types based on a pretrained universal representation of content for integrity problems; WPIE itself significantly improved IV compared to an initial strategy that used fixed allocations for different risk models [Rosen, 2019].