AI ASSURANCE WITH DOUBLE MACHINE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

AI assurance is the problem of testing whether artificial intelligence (AI) and machine learning (ML) systems adhere to a set of policies. Machine learning systems are particularly hard to assure since they are error-prone, unfair and biased. For example, is an auditor's claim that somebody is manipulating the system actually true or is there a more mundane explanation owing to an inaccurate or unfair classifier? To this end, we advocate the use of double machine learning (DML) as a framework for testing falsifiable assurance claims against ML systems. The approach directs ML upon itself to explain away its own workaday behaviors — behaviors that otherwise interfere with behavioral ML assurance tests. We study DML on the problem of detecting manipulation in content moderation systems. We find that DML allows us to isolate the effects of manipulation even in the presence of other interfering phenomena. On manipulation detection, DML achieves an AUC of 0.92 compared to fairness and statistical baselines which achieve AUCs around 0.76. We find DML succesfully detects a variety of manipulation attempts, including loss function manipulation, intentional bias amplification and the direct manipulation of its moderation decisions. Strikingly, in the latter case, DML estimates directly reflects the percent of manipulated content moderation decisions.

1 INTRODUCTION

Machine learning and AI systems pervade decision making in many areas of society, government and industry. These decision making processes must abide by relevant policies such as industry standards, laws, or internal guidelines set by the organizations themselves. This is especially important as tech companies in general, and social media platforms in particular, face increasing government and media scrutiny, sometimes in the form of recurring congressional hearings (Congress, 2025). The problem of AI assurance is to test whether machine learning and AI systems actually *align* with the policies. We consider the particular problem of behavioral AI assurance, by which we mean that the machine learning system behaves as expected and that we can assure this behavior through observing its associated input and output data.

The problem of AI assurance is difficult because machine learning systems are biased, they are unfair, and they make mistakes (Caton & Haas, 2024; Maynez et al., 2020). These behaviors are "expected" and make it difficult to distinguish between ordinary model behavior and genuine system manipulation. Manipulation along an axis produces unfair results along that same axis, but how do we know if the unfairness observed in the data is due to manipulation and not inherent to the data itself or due the confounding factors of ML? This problem is a rather severe form of the "crud factor" which is the idea that there is an ambient non-zero correlation between all random variables (Meehl, 1990; Recht, 2024). Statistically test any two subsets of the data along some hypothesized axis of manipulation and given enough content moderation decisions, you almost certainly will find a statistically significant difference between them. There is a lot of crud.

To make it more concrete, social media platforms such as Facebook, Instagram and TikTok rely heavily on ML and AI to moderate, recommend, and monetize the content posted on the platform. Such platforms typically have policies, such as community guidelines, describing what content is allowed. For example, the community guidelines might disallow hate speech, bullying and violence. Then it is the content moderation system's job to only allow content not in violation of these guidelines. The sheer scale of these systems means ML and AI are involved in most of these decisions.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Consider the particular assurance problem of determining from the input/output data whether or not the content moderation system is manipulated to suppress some specific content pertaining to a particular topic or idea. First we start with a testable hypothesis, that might take the form of a *claim* affirming that the system "behaves as expected" (Kapusta et al., 2025). For example, that the content moderation system is not manipulated to suppress speech about a topic of interest. In order to test this claim, we might consider comparing the rate at which content about that topic is banned with the rate at which content about other topics is banned. If the content of the topic is "unfairly" banned at a higher rate than other topics, say, in violation of the 4/5ths rule, then we might conclude that the claim is false and some manipulation is taking place to suppress the topic.

But what if the topic *should* be banned at a higher rate because it has more hateful content? The classifier appears "unfair" or "biased" even though it's behaving correctly by banning hateful content. Moreover, the unfairness might be due to random chance or a machine learning classifier simply behaving unfairly as it sometimes does. In these cases we have a false alarm — we might conclude that the content moderation system is manipulated when in fact it is not.

To address this problem we need to explain away the mundane expected behavior of the machine learning system so that we can more easily look for systematic suppression of content. An obvious way to do this is to train our own ML classifier to perform the content moderation based on the inputs and outputs of the system. Then, if one topic tends to contain more hateful content and our classifier uses text-based features, it will then account for the disparity since posts on that topic contain more racial slurs or other textual signals associated with harmful content. Moreover, our classifier is based on ML so it too is biased, it too is unfair and it too makes mistakes. Having accounted for ordinary ML behavior (both desirable and undesirable), we can look at the remaining errors and see if there are any patterns of suspicious activity. We propose to do this with double machine learning (DML) (Chernozhukov et al., 2024). In DML we train one classifier to predict the outcome variable (e.g., ban or not) and another classifier to predict the treatment variable (e.g., topic of interest or not). We can then analyze the relationship between treatment and outcome after removing the information already explained by the content moderation features which are responsible for policy adherence.

We evaluate DML as a way to detect manipulation in content moderation classifiers. In particular, we employ the Jigsaw toxicity dataset and train our own content moderation classifiers, some of which we manipulate and some of which we do not. In the absence of manipulation, we find that statistical tests like the $4/5$ths rule mistake variation in toxicity rates for manipulation, resulting in a false positive claim that there is manipulation when in fact there is not. Even when there is little or no variation in the actual base rates, there are still occasional false positives just due to random chance. However, if we employ an interactive regression model from DML, and compute an adjusted selection rate ratio from its average treatment effect, we find that after applying the $4/5$ths rule, we eliminate false positives entirely. Moreover, when considering both manipulated and unmanipulated classifiers, we find that DML accounts for the varying base rates between topics, achieving a manipulation detection improvement on baseline methods that do not (0.92 AUC-PR for DML and 0.75 AUC-PR for baselines). DML is able to detect both direct manipulation, which is a manipulation that flips content moderation decisions based on some rule, and indirect manipulation, which injects manipulation through modification of the loss function, features and hyperparameters. For the case of direct manipulation, the DML estimates are on average in a linear $y = x$ identity function with the percent of labels flipped by the manipulation rule.

2 Related Work

Despite its importance, the problem of AI assurance is relatively unexplored. Some recent initiatives on the subject include those at NIST and MITRE (Laplante & Kuhn, 2022; Robbins et al., 2024). Recent work proposes a "claim-based" framework which we also employ (Kapusta et al., 2025). However, we focus on DML as a specific way of addressing a claim within a claim-based framework.

Among other things, policies may discuss the accuracy, fairness and privacy of the machine learning components and data. Consequently fairness and privacy in machine learning are related areas (Kamiran & Calders, 2009; Dwork et al., 2011; Dwork, 2008; Caton & Haas, 2024; Rigaki & Garcia, 2023). For privacy, assurance may involve membership inference attacks or epsilon estimates. For fairness, assurance may involve measurements like demographic parity, or identifying

bias in the data collection/labeling process. However, in our work, fairness is a poor proxy for suppression via manipulation and acts as a confounder that interferes with our assurance claims.

Alignment of artificial intelligence (AI) and large language models (LLMs) is another related topic. In such work, human feedback is incorporated via fine-tuning, reinforcement learning or via a "constitution" (Ziegler et al., 2019; Bai et al., 2022). Here, alignment is the process of constructing a machine learning system that adheres to the policy, in this case, a repertory of human values including helpfulness and harmlessness. However, we view AI assurance as a verification problem, in which we verify that the model adheres to the policy rather than as an enforcement or mitigation problem in which we force the system to adhere to the policy. Of course AI assurance and alignment go hand in hand here because evaluation of the alignment invariably involves AI assurance. Similarly, an AI assurance report may highlight the need for better policy alignment.

AI assurance is a special but interesting case of software assurance which includes coding best practices, testing, benchmarking as well as more exotic method such as formal verification. Formal verification is a powerful tool for software assurance and can be employed to guarantee the semantic correctness of a program; the CompCert C compiler is well known example (Leroy, 2009). Formal verification of AI systems is more challenging due to randomness, but progress is being made in the areas of probabilistic programming, simple ML systems, and even some of the very foundations of differential privacy (Tristan et al., 2020; Tassarotti et al., 2021; de Medeiros et al., 2025). While formal verification is a tool to assess properties such as privacy, fairness or performance, it is less clear how to employ it for behavioral AI assurance in which we must collect evidence for or against an assurance claim based on the observed input output data. Moreover, even simple verification problems on transformers are intractable (Bergsträßer et al., 2026).

Randomly controlled trials (RCTs), A/B testing and causal inference frameworks are also potentially useful tools for AI assurance (Kohavi et al., 2009). We consider the assurance problem of verifying behavior from the input/output data only. This precludes the ability to intervene on the system and the ability to randomly assign treatments. For this reason, we turn to the causal inference framework of DML to help control for confounding variables, allowing us to turn a natural experiment into something more like an RCT or A/B test. While a natural experiment is not always ideal, they are still useful, as John Snow famously demonstrated in identifying cholera (Tulchinsky, 2018). Other causal inference techniques such as propensity score matching are also a helpful tool, but we do not consider other techniques in this work (Rosenbaum & Rubin, 1983; Deng et al., 2013).

3 Problem Statement and Approach

More formally, we apply double machine learning (DML) to the AI assurance problem of manipulation detection as follows. We consider a content moderation system $f(x) \mapsto y$ that inputs content x and outputs a binary decision y, allow or ban, which determine whether or not the content is allowed on the platform. In our setting, we observe the data (input X and output Y) corresponding to content moderation, but we do not have access to the inner workings of the content moderation system (f) nor the ability to intervene on its inputs or outputs.

We start with a falsifiable claim that content moderation behaves as expected and does not suppress content of a specific type. That *type* could be some class that indicates the identity of an individual, a geographic location, a political topic, or something else of interest. Without loss of generality we refer to this somewhat informally as the "topic" t. We say the topic is present if $t = 1$ and absent if $t = 0$. We ask the question: when holding the set of content moderation features $\Phi = \{\phi(x)\}$, does the presence of the topic ($t = 1$) positively influence the decision to ban the content $(y = 1)$. If we find such influence after accounting for content moderation features, we say that this is evidence against our claim and for the hypothesis that the system is manipulated to suppress the topic.

Next, to answer this question, we map our claim into the causal structure of DML by treating the topics T as a binary treatment variable (e.g., present or absent), the content moderation's binary output Y as the outcome variable, and a set of content moderation features Φ as the "nuisance" variables. In general we do not have access to the inner working of the content moderation system and must guess what features to use. For text data, this might be the tokenized text or embeddings. For video or audio data it might be OCR, ASR or automatically generated textual descriptions or summaries. The features are domain dependent and up to the discretion of the auditor.

There are multiple types of DML and in our work we employ an interactive regression model (IRM).

Interactive regression model Similar to partial linear regression (PLR), the interactive regression model (IRM) trains classifiers to predict the treatment and outcome variable. However, IRM allows each data point to contribute to the estimate differently via an interaction between features and treatment $Y = g(X, T) + \delta_Y$. Note that the classifier g is a function of the treatment and nuisance variables. Indeed, the IRM trains *two* classifiers for predicting the outcome Y: one classifier that predicts the outcome given the treatment is true and the other that predicts the outcome given the treatment is false. This enables us to produce counterfactuals for each individual data point so we can estimate the average treament effect (ATE):

$$\theta = E[g(X, T = 1) - g(X, T = 0)]. \tag{1}$$

Since our treatment and outcome variables are binary we have $\theta \in [-1, 1]$. For content moderation, we can interpret a positive ATE as the fraction of content that was banned because of its topic if we hold all the nuisance variables fixed. For example $\theta = 0.1$ means ten percent is banned because of the topic assignment.

Our interpretation of θ relies on the assumption that the features ($\phi(X)$) capture the factors that jointly influence the treatment and outcome variable. Because these features may not capture every relevant factor, this assumption may not always hold; in particular, unobserved factors could still confound the estimates. We therefore interpret θ causally to the extent that the observed features sufficiently account for such confounders, while treating violations of this assumption as a limitation of our assurance approach. We study the problem of missing confounders in our experiments.

4 Experiments

We study the behavior of double machine learning (DML) on the problem of detecting the manipulation of content moderation classifiers. We focus on manipulation that suppresses specific types of content. Our objective is to evaluate whether DML can distinguish genuine manipulation from other sources of moderation discrepancy, e.g., due to differences in content distribution across treatment groups. We compare DML with statistical baselines under challenging conditions in which treatment groups have substantially different toxicity rates. Such differences can produce false positives in AI assurance tests, leading an auditor to incorrectly attribute observed disparities in moderation outcomes to manipulation. We investigate whether DML can reduce false positive manipulation finding as well as its overall capability on this problem.

4.1 Data

We employ the Jigsaw dataset, which is a set of online news comments annotated for both toxicity and identity (e.g., jewish, gay, male, female, etc) (Adams et al., 2019). We retain comments for which both toxicity and identity annotation are available resulting in about 400K comments. We then embed each comment with EmbeddingGemma 300M and employ these representations as features for training content moderation classifiers, building nuisance models and constructing topic-based treatment variables (Vera et al., 2025). We employ an 80/20 train test split. The data is skewed, only 5.6% are labeled as toxic (toxic threshold of 0.5).

For treatment variables, we augment the dataset with synthetic topics designed to satisfy three properties: (1) semantic relevance to the underlying text, (2) balanced treatment assignment, and (3) a controlled ratio of toxicity rates between the treated and untreated groups. This construction allows us to systematically vary the relationship between topic presence and toxicity while evaluating manipulation detection. Controlling this ratio is important because in realistic settings, an auditor may observe different ban rates across topics without knowing whether those differences are attributable to manipulation or different toxicity rates.

We generate binary topic variables that achieve these properties as follows. First, we employ 20% of the embedding dimension to draw a random hyperplane through the data that we then shift by an appropriate bias, the median distance to the hyperplane, to create a synthetic topic that is both semantically related to the text and also balanced 50/50. Next, to achieve the desired toxicity selection rate ratio, we solve for the number of points that we must move between the appropriate cells of

Ratio	Ground truth		Content moderation			
	SRΔ	Adjusted SRΔ	SRΔ	DML-adjusted SRΔ	4/5 fails	DML-adj. 4/5 fails
1	-0.02	0.00	-0.27	-0.10	**1**/10	0/10
1.25	2.51	0.00	1.09	0.17	**3**/10	0/10
1.5	4.53	0.00	2.18	0.38	**5**/10	0/10
2	7.56	0.00	3.79	0.64	**7**/10	0/10
4	13.63	0.00	7.00	1.04	**10**/10	0/10
8	17.67	0.00	9.12	1.18	**10**/10	0/10
16	20.05	0.00	10.33	1.19	**10**/10	0/10

Table 1: DML adjusted selection rates mitigate the "false positives" assurance problem in which different toxicity base rates between topics lead to erroneous 4/5ths rule violations. The DML-adjusted selection rate differences are closer to the ground-truth adjusted differences of 0 (no suppression in this data) and reduce the number of 4/5ths violations to zero. SRΔ stands for selection-rate differences and is the difference between the rate at which topic 1 is banned and topic 0 is banned. The adjusted versions account for valid bans and do not include them as part of the rate.

Method	Detection AUC	DML treatment			DML outcome		
		P	R	F1	P	R	F1
DML (25% features)	0.883	0.710	0.713	0.712	0.876	0.580	0.697
DML (50% features)	0.908	0.754	0.755	0.755	0.917	0.688	0.786
DML (75% features)	0.918	0.782	0.783	0.782	0.936	0.745	0.830
DML (100% features)	0.924	0.810	0.811	0.810	0.954	0.783	0.860
selection-rate ratio	0.758	–	–	–	–	–	–
topic/output correlation	0.759	–	–	–	–	–	–
mutual information	0.752	–	–	–	–	–	–

Table 2: Manipulation detection evaluation

the topic-toxic contingency table from which we determine appropriate bias terms for shifting the hyperplane when a certain contingency condition is met. This effectively creates two hyperplanes that are identical except that they have a different bias that depends on the toxicity label. Since we start out with balanced topics, the number of point we need to move between the contingency table cells δ to achieve the selection rate ratio r is $\delta = (r * n_a - n_p)/(r + 1)$ where n_a is the number of data points with the topic absent ant n_p is the number of data points with the topic present.

4.2 Detecting manipulation in the presence of confounding skew

In this experiment we evaluate DML's ability to detect content moderation manipulation under varying amounts of confounding selection rate ratio skew. We deliberately create observational differences that look like manipulation, then test if DML can distinguish those differences from actual manipulation: to be succesful, we must distinguish between selection rate skew due to one topic being genuinely more toxic and selection rate skew due to manipulation that suppresses the topic.

To this end, we exert experimental control over the toxicity rate ratio of generated topics. This ratio is the selection rate of content in which the topic is present over the selection rate of content in which the topic is absent, according to the ground-truth toxicity labels. We create topics with ground-truth selection rate ratios from the set $\{1, 1.25, 1.5, 2, 4, 8, 16\}$, representing increasing levels of confounding skew, and generate ten topics per ratio condition. For each topic and ratio we train both a manipulated and unmanipulated classifier. The manipulated classifier suppresses the topic by more frequently assigning content associated with the toxic label even when it's not toxic. While we could simply flip the classifier's outcome to achieve this, it is a bit blunt so we opt for a more implicit manipulation. For the manipulated classifiers, we employ the 80% of embedding features that were not employed to generate the topics as well as the binary topic feature in order to create an amplification effect. Including the topic as a feature successfully manipulates the classifier to suppress content from the topic with the higher selection rate, amplifying the ground-truth selection rate skew by factors greater than one. The non-manipulated classifier does not include topic as a feature and the amplification factors for this classifier are all less than or equal to 1. We employ

a DML interactive regression model in which the outcome and treatment classifiers are logistic regression classifiers with embedding features and an L2 regularizer weight of 1. We implement the classifiers and DML with sklearn and the DoubleML python API (Bach et al., 2022). We employ 5-fold cross-validation for the DML estimates.

First, in Table 1, we study the *unmanipulated* classifiers on the ten random topics per ground-truth selection-rate (SR) ratios. Since the topics have different toxicity base rates, the unmanipulated content moderation classifiers correctly reflect this fact and rightfully bans those topics at higher rates. Consequently, when we apply the $4/5$ths rule to the raw unadjusted selection rates of the content moderation classifier, we see many violations. However, after employing DML to adjust for the content moderations features, we see the selection rate delta SRΔ reduce towards zero and achieve zero violations of the $4/5$ths rule. This highlights the danger of blindly applying the $4/5$ths rule with a statistic like selection rate without first accounting for the confounding variables.

While these results highlight the ability of DML to reduce AI assurance false positives, what about false negatives and overall performance as a manipulation detector? To this end, we evaluate DML's potential as a classifier for distinguishing between manipulated and unmanipulated content moderation classifiers. We employ multiple DML runs in which we limit the subset of the feature dimensions to 25%, 50%, 75% or 100% to emulate conditions in which an auditor's classifiers were unable to employ all available features and were therefore suboptimal. For each model we calculate its DML coefficient which indicates how much the topic variable influences the toxicity prediction of the content moderation classifier. We also report baselines such as selection rate, topic-toxicity correlation and topic-toxicity mutual information. These statistical baselines measure the dependence between the topic and the content moderation's toxicity decision.

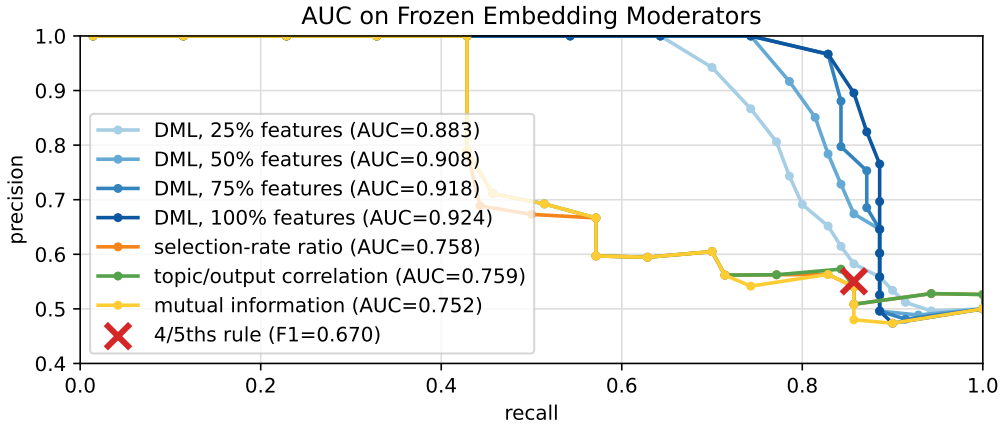


Figure 1: Manipulation detection.

We employ these scores as a manipulation detector by thresholding them accordingly: a score above the threshold means manipulation and below the threshold means no manipulation. In Figure 1, we plot the area under the precision recall curve (AUC) for each of the DML classifiers as well as the baselines: selection rate, topic-outcome correlation, topic-outcome mutual information. DML achieves the highest performance with an AUC of 0.92 compared to just $\sim$0.76 for each of the baselines. Moreover, we see that the performance of DML degrades as we lose access to the content moderation features. The AUC for DML drops to 0.86 when using only 25% of the available feature dimensions. In Table 2 we report these results along with the DML classifier performances for both the treatment variable and outcome variable. We can see that when using only 25% of the features the outcome variable predictor achieves only 0.643 F1, but this is is still sufficient to achieve an 0.86 AUC which is higher than any of the non-DML baselines. This is encouraging as it indicates DML can improve the quality of our estimates even if we cannot conjure up all the relevant features.

We also conduct a version of the detection experiment with more powerful content moderation classifiers by fine-tuning the entire GemmaEmbedding 300M model end to end for 3 epochs over the data in each condition. Due to computational resources we only run 5 random topics per ratio instead of 10 and skip the 1.25 and 1.5 ratio conditions. For manipulated versions, we include topic as a special token in the text and weight the loss on topic&toxic examples by 4x. The results follow

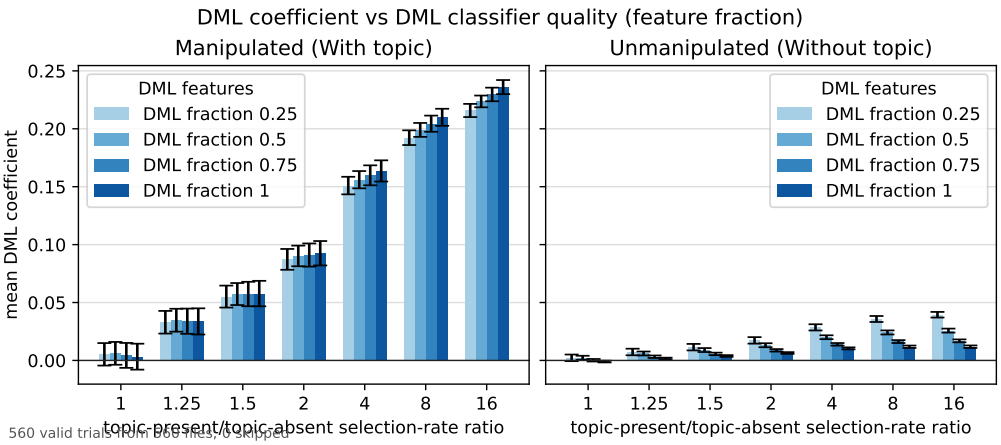
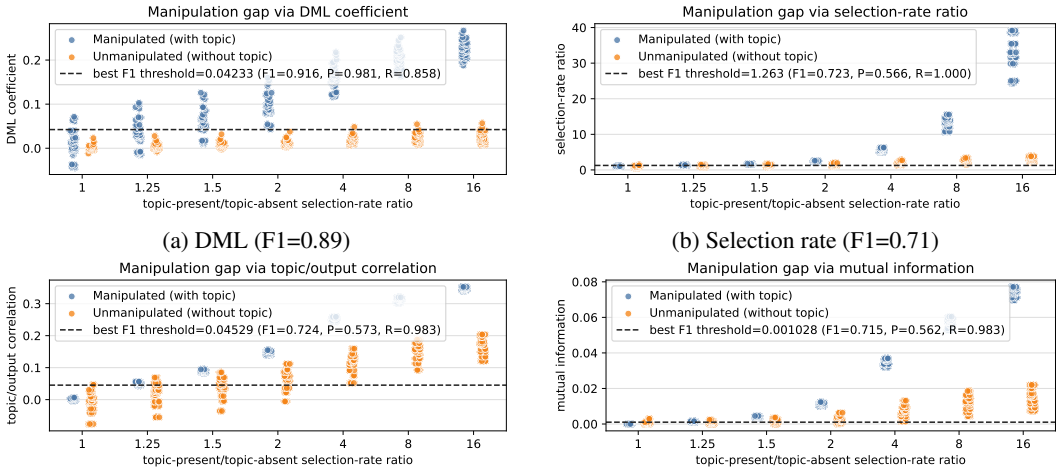


Figure 2: DML coefficient for manipulated (left) and non-manipulated classifiers (right)



(a) DML (F1=0.89)

(b) Selection rate (F1=0.71)

(c) Correlation (F1=0.71)

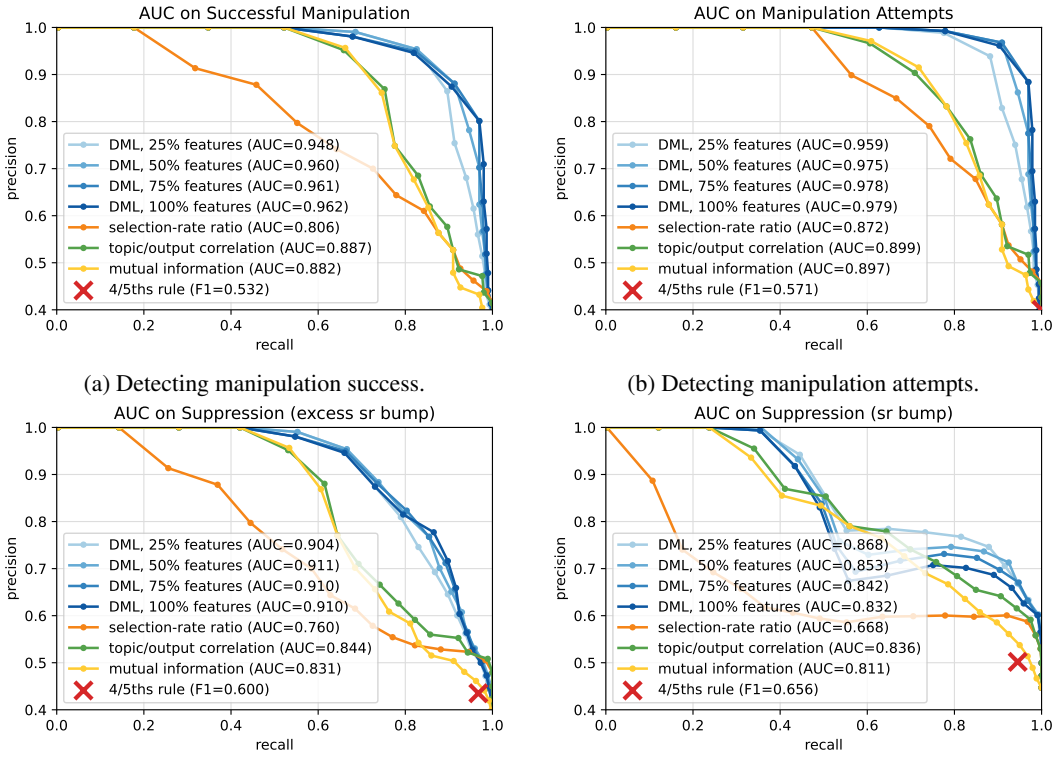
(d) Mutual information (F1=0.69)

Figure 3: The gap between the orange (unmanipulated) and blue (manipulated) points display the discriminative power of each metric for manipulation detection. The larger the ratio on the x-axis, the more manipulated the blue classifier (1=no manipulation).

a similar pattern as before: DML achieves an AUC of 94.7 while selection rate, correlation and mutual information achieve AUCs of 0.908, 0.905, and 0.905 respectively. The higher AUCs are due in part to dropping two hard conditions and in part due to the loss weight which makes it easier to distinguish manipulation from non-manipulation in the lower ratio cases.

Finally, for the frozen moderation classification detection case, we also report DML estimate for each condition on the topic variable of interest, averaged over the ten binary topics and for each selection ratio and classifier type (Figure 2). As we can see the DML estimates are much larger for the manipulated classifier and the estimate grows as the selection rate ratio grows (most pronounced for the manipulated condition). We can further compare how the DML coefficient compares to some baselines. In Figure 3 we show how each metric "scores" each classifier (manipulated or not) across the ten random trials per classifier. To get a sense of the discriminative power of each approach, we draw the decision boundary with the highest F1 through each plot. The optimal decision boundary for DML is 0.87 but only 0.69, 0.69, 0.68 for the baselines selection rate, correlation and mutual information respectively. Note that this does not give us a way of choosing a decision boundary, which we consider to be an open problem that involves various stakeholders, but does give us a sense of the potential of each method for constructing a manipulation detector.

4.3 DETECTING MANIPULATION VS SUPPRESSION



(a) Detecting manipulation success.

(b) Detecting manipulation attempts.

(c) Detecting excess suppression.

(d) Detecting suppression (includes non-excess).

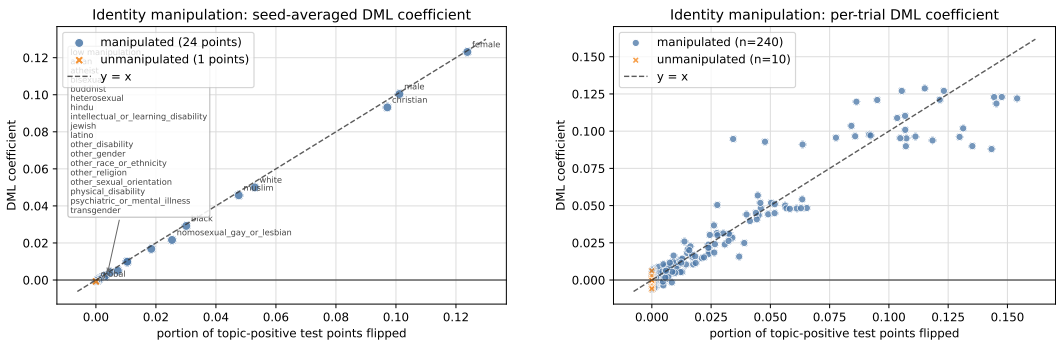
Figure 4: Detecting intentional and attempted manipulation versus suppression.

A manipulation attempt may or may not be succesul at achieving its goal of suppressing some topic. In this way, there are two aspects of manipulation: the intent to suppress and the actual suppression of the data. A manipulation attempt might not be succesfull and similarly, an unmanipulated content moderation system may inadvertently result in suppression. Our primary goal is to catch succesful intentional manipulation, but catching manipulation attempts and suppression irregardless of manipulation may also be important.

With this in mind, we create a content moderation manipulation dataset by training 980 content moderation classifiers on Jigsaw toxicity data. The classifiers differ by the hyper parameters that control the classifier as well as the data on which they are trained: classifier uses topic as a feature or not (2 conditions), classifier loss modified to give different weight to examples that have the topic and is also toxic (7 conditions: $\{0.125, 0.25, 0.5, 1.0, 2, 4, 8\}$), selection rate ratio between topic and rest of data (7 conditions: $\{1, 1.25, 1.5, 2, 4, 8, 16\}$) and topic of interest (10 conditions, randomly generated topics). We further employ DML in four capacities by controlling access to the full set of content moderation features: $0.25, 0.5, 0.75, 1.0$.

Some classifiers we intentionally manipulate and some we do not. Further, some classifiers result in suppression and some do not. We define a manipulated classifier as a specific set of hyperparameters that we could reasonable construe as a manipulation attempt: for example, including the topic as a feature to amplify selection rate skew and also increasing the loss function for data points in which the topic is observed and is toxic. Since we have the ground-truth toxicity rates for all the topics, we define suppression with respect to this. In particular, we say there is suppression if there is an *excess selection-rate bump*. That is, the classifier increases the selection rate (toxicity rate) of the topic more than it increases the selection rate of the rest of the data. Note that by design, this definition therefore excludes classifiers that generally err on the side of banning content.

In this 980 classifier dataset, 34% of classifiers are manipulated, 85% of which lead to actual suppression. Of the 66% that are not manipulated, only 8% lead to (inadvertent) suppression. The DML



(a) All topics

(b) Average over topics

Figure 5: DML coefficient vs identity suppression

estimates for successful manipulation is the highest at 0.23, but is also elevated for unsuccessful manipulation tempts at 0.09. Comparatively, for the unmanipulated classifiers, the estimates are 0.03 (lead to accidental suppression) and 0.01 (no accidental suppression) respectively.

In Figure 4 we evaluate DML on this dataset along the axes of manipulation and suppression. We see that DML is often better than the baselines at detecting manipulation and suppression together, as well as manipulation and suppression separately. Interestingly, DML is better at detecting along the manipulation axis than the suppression axis. This means we could potentially use it to detect manipulation attempts regardless of whether or not they were succesfull. We also employ another definition of "suppression" that only looks whether the classifier increases the topic's selection-rate over the ground-truth. In this case, DML only has a slight performance advantage.

4.4 Interpreting the DML coefficient under direct manipulation

Here we employ a more blunt approach of directly manipulating the content moderation classifier's output to ban a single identity in the Jigsaw dataset when that identity appears in the topic of interest. That is, the moderation classifier outputs its normal prediction if either the topic is absent or the topic is present but the target identity is absent. If the target identity and target topic is present, the moderator's prediction flips to toxic, regardless of whether it is toxic or not. While topics are generated to be balanced w.r.t. the data, the identities are not. Consequently, the amount of manipulated moderation decisions varies from identity to identity. We plot the DML coefficient against the percentage of manipulated data points (Figure 5). In Figure 5a we plot each trial separately and in Figure 5b we average over the ten topics per identity. We can see that the DML estimate closely tracks the identity function. That is it directly reflects the amount of direct manipulation to the data.

5 Limitations and Future Work

In this paper we adovcate the use of double machine learning (DML) as a tool for AI assurance. We study DML on the problem of detecting manipulation in content moderation and find it promising, overcoming the shortcomings of more basic statistical measures such as correlation, mutual information or detectors built upon selection rate ratios by reducing the number of false positives. In one of our detection problems we found DML has a detection AUC of 0.92 compared to AUCs of just 0.76 for the baselines.

There are a few limitations to our work including the following. First, while we believe that DML is widely applicable in AI assurance, we only study it on the problem of content moderation. In future work we would like to expand the scope of its applicability to AI assurance more broadly. Moreover, it is not always possible to account for all the nuisance variables. In this case, we may get misleading results; however, accounting for some of the nuisance variables is better than accounting for none of them (most existing baselines cannot account for them at all). While not studied here, in the future, we may also be able to employ instrumental variable regression to account for "hidden" confounders as done in other applications of DML (Shao et al., 2024).

AI USE STATEMENT

We did *not* use AI tools for any of the writing or ideation of this paper, not even for polishing. It was entirely written and polished by the human authors. AI was only used as a code assistant and this includes code that produced our figures. We have reviewed all AI-assisted work and take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

For reproducibility we took care to include all the relevant hyper parameters in the main paper including experimental variables. For experiments, the experimental variables are included in the relevant subsections of Section 4. Similarly, in this section we include the types of classifiers we employ and their hyperparameters along with the toolkits we employ to implement them. We also include extra results that go beyond the main research questions, which can be used in the replication process, such as the treatment and outcome performance of our nuisance classifiers in Table 2.

REFERENCES

CJ Adams, Daniel Borkan, inversion, Jeffrey Sorensen, Lucas Dixon, Lucy Vasserman, and nithum. Jigsaw unintended bias in toxicity classification. https://kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification, 2019. Kaggle.

Philipp Bach, Victor Chernozhukov, Malte S. Kurz, and Martin Spindler. Doubleml - an object-oriented implementation of double machine learning in python. *Journal of Machine Learning Research*, 23(53):1–6, 2022. URL http://jmlr.org/papers/v23/21-0862.html.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022. URL https://arxiv.org/abs/2212.08073.

Pascal Bergsträßer, Ryan Cotterell, and Anthony W. Lin. Transformers are inherently succinct, 2026. URL https://arxiv.org/abs/2510.19315.

Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ACM Comput. Surv.*, 56(7), April 2024. ISSN 0360-0300. doi: 10.1145/3616865. URL https://doi.org/10.1145/3616865.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and causal parameters, 2024. URL https://arxiv.org/abs/1608.00060.

U.S. Congress. Hearing on government social media censorship. Hearing, Commerce, Science and Transportation Committee, Washington, DC, October 2025. URL https://www.c-span.org/program/senate-committee/hearing-on-government-social-media-censorship/666821.

Markus de Medeiros, Muhammad Naveed, Tancrède Lepoint, Temesghen Kahsai, Tristan Ravitch, Stefan Zetzsche, Anjali Joshi, Joseph Tassarotti, Aws Albarghouthi, and Jean-Baptiste Tristan. Verified foundations for differential privacy. *Proc. ACM Program. Lang.*, 9(PLDI), June 2025. doi: 10.1145/3729294. URL https://doi.org/10.1145/3729294.

Alex Deng, Ya Xu, Ron Kohavi, and Toby Walker. Improving the sensitivity of online controlled experiments by utilizing pre-experiment data. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pp. 123–132, New York, NY, USA,

2013. Association for Computing Machinery. ISBN 9781450318693. doi: 10.1145/2433396.2433413. URL https://doi.org/10.1145/2433396.2433413.

Cynthia Dwork. Differential privacy: a survey of results. In *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*, TAMC'08, pp. 1–19, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 3540792279.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. Fairness through awareness, 2011. URL https://arxiv.org/abs/1104.3913.

Faisal Kamiran and Toon Calders. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication*, pp. 1–6, 2009. doi: 10.1109/IC4.2009.4909197.

Ariel S. Kapusta, David Jin, Peter M. Teague, Robert A. Houston, Jonathan B. Elliott, Grace Y. Park, and Shelby S. Holdren. A framework for the assurance of ai-enabled systems, 2025. URL https://arxiv.org/abs/2504.16937.

Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M. Henne. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery*, 18:140–181, 2009. URL https://api.semanticscholar.org/CorpusID:17165746.

P. Laplante and D. Kuhn. Ai assurance for the public – trust but verify, continuously. In *2022 IEEE 29th Annual Software Technology Conference (STC)*, Gaithersburg, MD, US, 2022.

Xavier Leroy. Formal verification of a realistic compiler. *Communications of the ACM*, 52(7): 107–115, 2009. doi: 10.1145/1538788.1538814.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.173. URL https://aclanthology.org/2020.acl-main.173/.

Paul E. Meehl. Appraising and amending theories: The strategy of lakatosian defense and two principles that warrant it. *Psychological Inquiry*, 1(2):108–141, 1990. doi: 10.1207/s15327965pli0102_1.

Benjamin Recht. Crud. https://www.argmin.net/p/crud, 2024. Substack post. Accessed: 2026-07-07.

Maria Rigaki and Sebastian Garcia. A survey of privacy attacks in machine learning. *ACM Comput. Surv.*, 56(4), November 2023. ISSN 0360-0300. doi: 10.1145/3624010. URL https://doi.org/10.1145/3624010.

Douglas Robbins, Ozgur Eris, Ariel Kapusta, Lashon Booker, and Paul Ward. Ai assurance: A repeatable process for assuring ai-enabled systems. Technical report, MITRE, June 2024.

Paul R. Rosenbaum and Donald B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 04 1983. doi: 10.1093/biomet/70.1.41. URL https://doi.org/10.1093/biomet/70.1.41.

Daqian Shao, Ashkan Soleymani, Francesco Quinzan, and Marta Kwiatkowska. Learning decision policies with instrumental variables through double machine learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 44489–44514. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/shao24d.html.

Joseph Tassarotti, Koundinya Vajjha, Anindya Banerjee, and Jean-Baptiste Tristan. A formal proof of pac learnability for decision stumps. CPP 2021, pp. 5–17, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450382991. doi: 10.1145/3437992.3439917. URL https://doi.org/10.1145/3437992.3439917.

Jean-Baptiste Tristan, Joseph Tassarotti, Koundinya Vajjha, Michael L. Wick, and Anindya Banerjee. Verification of ml systems via reparameterization, 2020. URL https://arxiv.org/abs/2007.06776.

Theodore H. Tulchinsky. John snow, cholera, the broad street pump; waterborne diseases then and now. In Theodore H. Tulchinsky (ed.), *Case Studies in Public Health*, pp. 77–99. Elsevier, 2018.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, Daniel Cer, Alice Lisak, Min Choi, Lucas Gonzalez, Omar Sanseviero, Glenn Cameron, Ian Ballantyne, Kat Black, Kaifeng Chen, Weiyi Wang, Zhe Li, Gus Martins, Jinhyuk Lee, Mark Sherwood, Juyeong Ji, Renjie Wu, Jingxiao Zheng, Jyotinder Singh, Abheesht Sharma, Divyashree Sreepathihalli, Aashi Jain, Adham Elarabawy, AJ Co, Andreas Doumanoglou, Babak Samari, Ben Hora, Brian Potetz, Dahun Kim, Enrique Alfonseca, Fedor Moiseev, Feng Han, Frank Palma Gomez, Gustavo Hernández Ábrego, Hesen Zhang, Hui Hui, Jay Han, Karan Gill, Ke Chen, Koert Chen, Madhuri Shanbhogue, Michael Boratko, Paul Suganthan, Sai Meher Karthik Duddu, Sandeep Mariserla, Setareh Ariafar, Shanfeng Zhang, Shijie Zhang, Simon Baumgartner, Sonam Goenka, Steve Qiu, Tanmaya Dabral, Trevor Walker, Vikram Rao, Waleed Khawaja, Wenlei Zhou, Xiaoqi Ren, Ye Xia, Yichang Chen, Yi-Ting Chen, Zhe Dong, Zhongli Ding, Francesco Visin, Gaël Liu, Jiageng Zhang, Kathleen Kenealy, Michelle Casbon, Ravin Kumar, Thomas Mesnard, Zach Gleicher, Cormac Brick, Olivier Lacombe, Adam Roberts, Qin Yin, Yunhsuan Sung, Raphael Hoffmann, Tris Warkentin, Armand Joulin, Tom Duerig, and Mojtaba Seyedhosseini. Embeddinggemma: Powerful and lightweight text representations, 2025. URL https://arxiv.org/abs/2509.20354.

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