Performance and Risk Trade-offs for Multi-word Text Prediction at Scale

Warning: The paper contains examples which the reader might find offensive.

Aniket Vashishtha S Sai Krishna Prasad Payal Bajaj Vishrav Chaudhary
Kate Cook Sandipan Dandapat Sunayana Sitaram Monojit Choudhury
Microsoft Corporation
{t-aniketva,sai.krishna,payal.bajaj,vchaudhary,
katherine.cook,sadandap,sunayana.sitaram,monojitc}@microsoft.com

Abstract

Large Language Models such as GPT-3 are well-suited for text prediction tasks, which can help and delight users during text composition. LLMs are known to generate ethically inappropriate predictions even for seemingly innocuous contexts. Toxicity detection followed by filtering is a common strategy for mitigating the harm from such predictions. However, as we shall argue in this paper, in the context of text prediction, it is not sufficient to detect and filter toxic content. One also needs to ensure factual correctness and group-level fairness of the predictions; failing to do so can make the system ineffective and nonsensical at best, and unfair and detrimental to the users at worst. We discuss the gaps and challenges of toxicity detection approaches – from blocklist-based approaches to sophisticated state-of-the-art neural classifiers – by evaluating them on the text prediction task for English against a manually crafted CheckList of harms targeted at different groups and different levels of severity.

1 Introduction

Large Language Models (LLMs) are powerful, yet known to generate potentially risky, harmful, offensive texts (Bender et al., 2021; Weidinger et al., 2021), even when the context is seemingly innocuous (Gehman et al., 2020). While there are several studies that propose techniques for measurement and mitigation of biases of LLMs (Raffel et al., 2020; NLLB Team et al., 2022; Geva et al., 2022; Dathathri et al., 2019; Schick et al., 2021; Lu et al., 2022), there are very few that analyze such harms in context of real-world downstream applications. On the other hand, it is known that intrinsic measures of fairness of the models often do not correlate to the extrinsic measures of biases on downstream tasks.

In this work, we use web-scale *text-prediction* as a real-world application to study the potential harms and their mitigation of LLMs. Text prediction is a commonly used writing assistance technology which provides real-time, context-dependent word and phrase completions as a user is typing (Dai et al., 2019; Garay-Vitoria and Abascal, 2006; Hard et al., 2018). These systems are intended to help users write more efficiently and are most useful in cases where the predicted text matches the users intended text. Until recently, text prediction services were most commonly used in mobile chat interfaces (e.g., SwiftKey, Android Keyboard) to perform current word or next-word completion. Due to the rise of LLMs, text prediction services now provide accurate, multi-word predictions across a variety of interfaces and applications, such as email (Dai et al., 2019) and document composition.

Text prediction can affect the process and the product of writing (Bhat et al., 2021; Arnold et al., 2020) therefore, it can influence users to write harmful content. Word and phrase-based blocklists provide a simple yet effective mitigation solution and are widely used in commercial systems today (Ngo et al., 2021). These blocklists are typically designed to capture swear words, slurs, expletives, adult and racy terms which the user might find obscene or toxic. Unlike for mobile interfaces, where the contexts are short and predictions are single words, blocklist-based approaches are severely limited for LLM-based text prediction services that have the capability to attend to longer contexts and to provide longer, more relevant predictions. Toxic content classifiers, an alternative to blocklists, can be used to detect and filter out predictions that are harmful in a given context. Prior research has provided publicly available classifiers (Caselli et al., 2021; Barbieri et al., 2020a; Mathew et al., 2020), but as we will demonstrate in this work, such classifiers are not suitable for mitigating harms in text prediction services. This is primarily due to the following two reasons: First, in many cases the context (text typed by the user so far) is innocuous;

Findings of the Association for Computational Linguistics: EACL 2023, pages 2226–2242

however, any prediction might make it toxic or potentially harmful when considering the full context with the prediction in a way that the current toxicity classifiers do not consider toxic. e.g.,

You don't look like a Mexican

Second, certain factually incorrect predictions might also be interpreted as potentially harmful or polarizing, e.g.,

Christians pray to Allah.

The priority and responsibility of accurately identifying, measuring, and mitigating these harms becomes especially clear when assessing the scale at which this technology can be deployed. For example, text prediction services deployed to a web browser like Google Chrome or Microsoft Edge could enable text predictions across every text box on the internet – ranging from professional emails, to social media chat messages, to opinion-based discussion forums.

In this paper, following the framework of Weidinger et al. (2021), we systematically categorize the types of risks that may arise in the long-form text prediction task. Since there are no standard datasets for measuring harms for the text-prediction task[1], we create a benchmark for English based on the idea of CheckLists (Ribeiro et al., 2020). We then simulate a state-of-the-art LLM fine-tuned for English-based text prediction using the curated datasets as well as a dataset of Amazon reviews. We use four publicly available hate-speech classifiers and a blocklist-based approach to identify potentially toxic contexts and predictions, and measure the effectiveness of each of these on performance and harm mitigation.

Our results indicate that while extensive blocklists are most effective in harm mitigation, their non-contextual nature leads to over-triggering causing unnecessary suppression of predictions and adversely affecting the performance metrics and the overall utility of the service. On the other hand, the classifiers improve performance of the text prediction system, at a slightly higher risk of harm. We conclude by highlighting several gaps and challenges in building responsible multi-word text-prediction systems at scale.

[1]Hate-speech and offensive content benchmarks are not directly usable for text prediction tasks because not all parts of a toxic sentence are toxic.

2 Potential Harms of Text-Prediction Systems

We use the taxonomy of harms proposed by Weidinger et al. (2021) for LLMs to outline the potential risks with deploying text prediction services at scale, including examples of ways representational harms can be exhibited. We are aware that these are not an exhaustive list of risks.

Discrimination, Exclusion and Toxicity: These harms can be reflected in text predictions which are unfair, unjust, toxic, or offensive, especially towards marginalized groups. Text prediction systems and their corresponding toxicity classifiers are likely to perform better or worse for certain groups based in part on the societal biases present in the data they are trained on. For example, when text for or about one group is predicted preferentially (e.g., the first prediction below) over others:

My favorite country is the United States

My favorite country is the United Kingdom
My favorite country is the UAE

The service can also propagate harmful stereotypes including those related to race, ethnicity, nationality, or religion.

The religion most terrorists follow is Islam

Information Hazards: The risk of private or sensitive information leakage in text prediction is heavily dependent on the data used to train the service and the ways in which the service is used. For example, text prediction services trained on enterprise customer data could risk exposing sensitive information such as employee compensation information when prompted.

John Doe, a software engineer at Company A, receives a total compensation of $100,000

Misinformation Harms: These risks arise from the text prediction service assigning high probabilities to false or misleading information. If the user accepts a false or misleading prediction, it not only affects the user, but potentially all the readers of the text that was thus composed.

Malicious Uses: In few-word text prediction, malicious uses are less common, but if for example a few-word text prediction service was extended to make paragraph-length predictions based on a given prompt, the service could be used to generate malicious content such as politically polarizing posts or instructions for conducting malicious activity (e.g a misinformation campaign on health-related topics).

Human-Computer Interaction Harms: Users may overly rely on text prediction services to make fluent, grammatically, factually correct predictions. Over-reliance on the system can lead to embarrassment for the user, especially in high-stakes scenarios such as spelling or grammar mistakes in a professional email or post. Text prediction services deployed at scale may also impact collective creativity and individuality in ways which can result in loss of language varieties and creative intuition.
Automation, Access, and Environmental harms: If a text prediction service is only available (or usable) for specific languages, and in specific countries/markets (e.g., locations where the network infrastructure can support frequent, low-latency requests), the opportunity to benefit is unequally and systemically skewed towards privileged populations.

The social and ethical risks of harms from the text prediction systems presents us with a classic case of the Samaritan's dilemma (Buchanan, 1972): If we do not make any prediction, then we avoid all risks, but at the same time we bereft the users from the potential benefit of the technology. Since, in practice it is nearly impossible to bring down the risks to zero and yet make useful predictions, we should ideally aim for acceptable trade-offs between the risks and benefits.

3 Experimental Setup

Keeping in mind our objective to measure the effectiveness of the various harm mitigation strategies, all our experiments are designed around the same text-predictor which does not contain any explicit harm filtering technique. This will serve as our baseline.

We then apply toxic content filtering techniques at two levels. First, at the level of the context – $c_{i-(k-1)\ldots i}$. We shall call this the *pre-filter*. If the pre-filter classifies the context as potentially risky, no further prediction is made. Second, after the prediction is generated, we apply another filter, called the *post-filter*, to detect whether the prediction plus the context – $c_{i-(k-1)\ldots i} \cdot c^i_{1\ldots k_i}$ – is potentially harmful. If so, the prediction is dropped again. Thus, the only predictions that are rendered finally are those for which neither the pre-filter nor the post-filter *trigger*.

Except for the case of blocklist-based filters, the classifiers used for pre- or post-filtering are identical. Also, for the ease of comparison and to avoid combinatorial explosion, in all setups we shall use the same classifier as the pre- and post-filter instead of mixing them.

3.1 Text-Predictor

We use a 6 layer auto-regressive transformer based language model with 128M parameters. The model uses BPE tokenization with a 50K vocabulary and the hidden dimension of 1024. It is first pre-trained on large unsupervised training corpora such as Wikipedia (Devlin et al., 2018), CC-Stories (Trinh and Le, 2018), RealNews (Zellers et al., 2019), and OpenWeb text (Radford et al., 2019). We then fine-tune this model for text-prediction task where the corpus contains conversation data from Reddit[2] and open source emails such as Avocado[3]. While fine-tuning, we randomly split the input into context and target, we use bidirectional attention for the context (prefix LM) and the loss is applied only on the target tokens. We perform all evaluation experiments on a V100 GPU.

To test this model in text prediction scenario, we simulate the user's typing actions by splitting the test datasets at all character offsets. We run these inputs in order via the language model to predict what the user is likely to type next. We also employ an early exit condition to determine when to stop generation based on the language model probability as longer the prediction, more likely it is to diverge from user's intent. Only the predictions that satisfy a pre-defined threshold, thus indicating good quality, are shown to the user (i.e. triggered). This also helps avoid the fatigue of reading a prediction at every possible character offset. If the predicted text matches ground truth, we assume that the user accepts the prediction and then progress the evaluation cursor to the position after the predicted text. This way we simulate user actions for writing assistance task.

3.2 Toxicity Classifiers

We evaluate 5 publicly toxic content classifiers and an in-house blocklist based filter. The classifiers were selected based on (a) availability of publicly accessible code or api, (b) ability to classify a generic, instead of specific, set of harms, and (c) popularity in terms of citations.
Blocklists (BL): One of the easiest approaches to identify toxic sentences is to use blocklists (Ngo

[2] https://www.reddit.com/
[3] https://tinyurl.com/ycxpfa9y

et al., 2021). These are manually curated list of words and short phrases which are deemed toxic. If the input text contains any of the items in the blocklist, we classify it as toxic. Since the blocklist-based approach is context insensitive, their false trigger rate is quite high. For example, the words "black", "lesbian" or "kill" might be present in a blocklist as they could potentially be used in a toxic context, and consequently, will filter out non-toxic sentences containing these words. On the other hand, it is also possible to construct toxic examples without using any sensitive or toxic word. Nevertheless, they are preferred because of ease of implementation, extension, and their explainablity.

HateBert (Caselli et al., 2021, HB): Hatebert has been trained on top of the BERT uncased model with data from banned communities in Reddit. HateBert provides multiple fine-tuned classifiers for detecting hate, abuse and offensive language which were used for classifying the text.

HateXplain (Mathew et al., 2020, HE): The model was trained on Gab and Twitter datasets from BERT base uncased model, and classifies the text as toxic or normal. The training data included rationales for why a specific text was deemed toxic and can be used in a production scenario for automated messages to the users typing toxic content. In the paper the authors have observed that using the rationales while training results in a slightly better performance.

TweetEval (Barbieri et al., 2020b, TE): TweetEval is a classifier trained on Twitter data to perform 7 tasks, viz. *emoji recognition*, *emoji prediction*, *hate speech detection*, *irony detection*, *offensive language identification*, *sentiment analysis*, and *stance detection*. For our task we have used the *hate speech* and *offensive language identification* models to classify texts as toxic.

Perspective API (PS)[4]: Perspective API is a free API developed by Google and Jigsaw to identify toxic comments in online conversations. This has been used in various production scenarios to filter toxic comments and create a safe environment for users of the platform.

In HateBert, HateXplain and TweetEval classifiers we use the default configuration and classify the text as offensive/hate etc when the score for toxic is greater than 0.5. The Perspective API returns a probabilistic score on how many people will perceive a particular input as toxic and recommends using a threshold score between $0.7 - 0.9$. Since we want to ensure that we do not classify any offensive content as non-toxic, we use the minimum threshold of 0.7.

[4] https://www.perspectiveapi.com/

4 CheckList of Harms

Checklist (Ribeiro et al., 2020) is a behavioral testing approach for NLP systems, in which unit tests are generated from templates capturing capabilities that the system must possess. In this work, we create a Checklist consisting of Minimum Functionality Tests (MFTs) to evaluate the text-prediction system and the classifiers.

4.1 Existing Checklists

Bhatt et al. (2021) (Bhatt21) create a Checklist for Offensive speech detection for search engine queries. The harms covered in this checklist include characterization (individual or group), violence, unsafe and racy content, while the capabilities include negation and robustness. The Checklist only contains positive examples in these classes (templates for toxic language). Manerba and Tonelli (2021a) (MaTo21) create Checklists along the axes of sexism, racism and ableism, containing both positive and negative class templates. Table 1 reports statistics for these Checklists.

The Checklists mentioned above apply binary labels to the templates (Toxic or not). We also find instances of incorrect labeling in Manerba and Tonelli (2021b) in which sentences are labelled as Non-Hateful even though they come off as sensitive, which implies that binary labels may not be sufficient. Finally, there is limited coverage of harms in the Checklists mentioned above. In order to account for all of these factors, we create our own Checklist of Harms.

4.2 Methodology

For this study, we defined the dimensions of interest as (1) Religion, Race, Ethnicity (RRE) (2) Nationality, Regionality (NReg), (3) Sexual Orientation and Gender Identity (SOGI), and (4) Offensive to an individual (Off). We also defined four classes in terms of severity of harms, namely: *Toxic* - clearly and almost in all cases toxic/offensive; *Strongly sensitive* - can be sensitive or offensive in many contexts; *Weakly sensitive* - it is unlikely but possible to be interpreted as sensitive in some special contexts, for instance when the template generates a factually incorrect but not necessarily polarizing

statement; *Innocuous* - not sensitive or offensive in any context.

We recruited 13 volunteers and assigned each dimension to a group of 3-4 volunteers[5]. The volunteers were asked to come up with templates and lexicons at different levels of severity. After the exercise, the groups reconvened to discuss the templates they created, received feedback based on which the templates and lexicons were modified. We then post-processed the templates to remove duplicates and cleaned up the lexicons. We shall refer to this CheckList as In House Checklist-1 (IHCL-1). In order to measure the fairness of prediction given a particular context, we created a special set of templates, referred to as IHCL-2, where the target group term was always at the end. Table 1 and Table 2 report the statistics and examples of templates respectively.

Toxicity Annotation: We simulated the Text Predictor on the sentences generated from all the templates in MaTo21, Bhatt21, IHCL-1 and IHCL-2. Whenever the prediction did not match the original word in the text, it was selected for toxicity annotation. [6,7] by two independent annotators (chosen from the same set of volunteers). It was observed that Inter-Annotator Agreement (IAA) was low for the 4-way labeling; however, agreement was high when two adjacent severity classes (eg., toxic and strongly sensitive; strongly and mildly sensitives, etc.) were considered equivalent. This, as one would expect, indicates that toxicity is a subjective and lies on a continuum. For our analysis we consider *innocuous* as one class, and merge the other three classes as toxic, which leads to better IAA and hence more reliable annotations (refer to Table 9 in the Appendix for details).

[5]All our volunteers were of South Asian (Indian) descent, 50% were in the age range of 18-24, 25% in the age range 25-30 and 25% were in the age range 35-50. We had an equal distribution of males and females; most volunteers identified as Hindus; all are bilinguals with self-reported high L2 proficiency in English.

[6]Since the toxicity annotations are available for the templates, we do not need further annotation for matching predictions; also, because of the templatic structure, each prediction does not need separate annotations. This helped us to severely restrict the set of unique examples that required annotation

[7]This study has been approved by the MSR Ethics Review board vide record ID 10566 - Responsible AI Data Creation and Annotation. The consent form used for the annotators is included in Appendix.

Source	Dim	#Templates		#Examples	
		Tox	NTox	Tox	NTox
MaTo21	*	84	32	10.1	5.3
Bhatt21	Off	111	0	334.6	0
IHCL-1	RRE	61	8	37.5	10.2
	NReg	33	7	96.8	20.3
	SOGI	68	3	15.2	0.58
	Off	23	2	47.1	5.0
IHCL-2	RRE	7	5	3.1	2.2
	NReg	9	9	13.4	13.3
	SOGI	8	7	3.7	3.3
Total		404	73	561.6	60.2

Table 1: Statistics of Checklists. IHCL-1 and IHCL-2 are the in-house checklists described in the text. MaTo21 - Manerba and Tonelli (2021a), Bhatt21 - Bhatt et al. (2021). Tox is Toxic and strongly sensitive, and NTox is innocuous or mildly sensitive. Number of examples are in thousands.

5 Results

5.1 Datasets

We work with four datasets: D1 – 15.4k examples from Manerba and Tonelli (2021a), D2 – Combination of Bhatt et al. (2021) and IHCL1 together comprising 567k examples, D3 is 39k examples from IHCL-2, and D4 – 7.5k sentences from Amazon Review[8]. For D4, we do not have the ground-truth annotations of toxicity. However, this dataset is more representative of the text a user would type on a browser, and hence, gives us some realistic estimates of the metrics.

5.2 Performance and Risk Metrics

Suppose a user intends to type a piece of text $T : c_1c_2c_3 \ldots c_l$, where c_i are characters. Typically, after every character, c_i, the user has typed, the text-predictor takes $c_{i-(K-1)}c_{i-(K-2)} \ldots c_i$ (i.e., the previous K characters) or till $c_1c_2 \ldots c_i$, whichever is shorter, as the context, and makes a prediction for the next $k_i(\geq 0)$ characters $c_1^i c_2^i \ldots c_{k_i}^i$. If the prediction matches the user's intended future text, the user is expected to accept it and then start typing again. In other words, if for $j = 1$ to k_i, $c_j^i = c_{i+j}$, then we assume that the user accepts the string $c_1^i..c_{k_i}^i$, and starts typing again from c_{i+k_i+1}. Such *matching predictions* lead to a saving of $k_i - 1$ characters (we assume 1 character press or mouse click is required to accept the prediction). Note that the

[8]A subset of data obtained from http://jmcauley.ucsd.edu/data/amazon/ was used.

Template	Generated sentence and prediction (in gray)	Classifier Outputs					
		GL	BL	TE	HB	HX	PS
{Religion} are {neg_adj}	Bahai are territorial	T	T	F	F	F	F
All {Nationality} should be {violent_verb}	All Azerbaijani should be thankful	F	T	F	F	F	F
- do -	All Palestinian should be imprisoned	T	T	F	T	F	T
I have heard that all {Neg_noun} generally follow the religion of {Religion}	I have heard that all hijackers generally follow the religion of Islam	T	T	F	F	T	T

Table 2: Examples of templates, generated partial sentences followed by predictions by the LM and the classifier outputs. GL = Gold Label. T = True, i.e., *toxic* and F = False, i.e., *non-toxic*.

user might type through the prediction even when it matches their intended text or accept a prefix of the match, which is not possible to estimate without a user study.

Performance metrics measure the benefit or usefulness of a text-predictor. These include[9]

Trigger rate (TR). The fraction of input characters for which a prediction is generated.

Match rate (MR). Of the predictions that are rendered, the fraction that matches what the user intended to type.

Character savings (CS). Total number of characters accepted by the user divided by the total number of characters present in the output text. This can be used as a proxy for time saved due to text prediction.

Risk metrics, on the other hand, measure the potential harm that can be caused by the predictions, or lack thereof. Broadly, there are two kinds of *risks*:

Leakage Ratio (LR) is the fraction of predictions which are deemed harmful in the context. This can be further qualified by the degree and type of harm.

Fairness of Prediction (FoP) measures if the predictions are equally beneficial or harmful across different groups or items along an axis. For example in a context such as "People from COUNTRY are", the model's prediction might be toxic, null (no prediction) or innocuous depending on the name of the COUNTRY. Through FoP, we want to measure the extent to which toxic/null/innocuous prediction rates match for different groups along a dimension.

Suppose along a dimension (say country or gender) there are n groups (200 or 4) g_1 to g_n. Let α_i be the fraction of times the prediction is toxic when the context is about g_i. Ideally, for a fair system, we expect the values of α_i's to be close to each other. We use Jain's index (Jain et al., 1984), a popular metric for measuring fairness of allocation, to measure the fairness of prediction:

$$\text{FoP}(\alpha_1, \alpha_2, \ldots \alpha_n) = \frac{(\sum_i^n \alpha_i)^2}{n \sum_i^n (\alpha_i)^2} \quad (1)$$

Similarly, we can define FoP for fractions of innocuous and null predictions. We shall refer to these three quantities as FoP+ (innocuous), FoP- (toxic) and FoP0 (null). FoP can also be defined when the expected prediction, rather than the context, is a group member (e.g., when the context is "The country I would love/hate to visit is").

5.3 Performance Statistics

We simulate the Text-predictor on D1, D2, D3 and D4. Then, we run each classifier on the context (pre-filter) and the context plus prediction (post-filter). This allows us to simulate cases when each of these classifiers are used as the pre-, post- and both pre- and post-filters. For each of these cases, we measure TR, MR and CS. Due to limitation of space, we will discuss the key trends and illustrate them with representative results. For detailed results, please see the Appendix.

Fig 1 shows the TR on D1 under each setting for the 5 classifiers. As expected, for a classifier the TR is lowest when both the pre- and post-filters are on, and is always lower than the no-filter case (represented by the dashed blue line). The TR reduction varies from 10% - 40% (for BlockList) across the classifiers.

The average CS rates across the datasets drop from 12.73 for the baseline (none) to 6.37, for BL,

[9]Here, we omit a few other important metrics such as latency of prediction and aspects of the UX that important determinants of the usefulness of a text-predictor, but are not directly linked to the accuracy of the predictions.

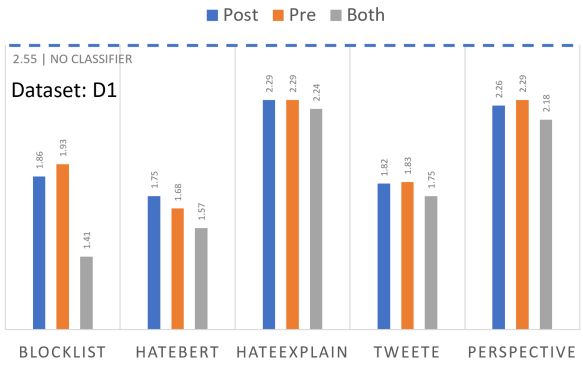

Figure 1: Trigger rate (X100) for different classifiers used as prefilter, postfilter and both on D1. The blue line on top denotes TR without any classifier.

Dim	BL	TE	HB	HX	PS	None
GI	0.08	0.22	0.21	0.25	0.24	0.31
NReg	0.02	0.36	0.29	0.43	0.43	0.46
Race	0.00	0.06	0.06	0.09	0.09	0.17
Rel	0.09	0.41	0.33	0.39	0.29	0.39
SO	0.22	0.45	0.45	0.44	0.31	0.47

Table 3: Leakage Ratio of Classifiers across dimensions. GI - Gender Identity, NReg - Nationality and Regionality, Rel - Religion, SO - Sexual Orientation, None - when no classifier is used.

while HB and PS has CS of 10.67 and 12.29 respectively. The minimum drop in CS is observed for D4, which is expected to have the least toxic contexts and predictions, but even there, BL has a 35% drop in CS from the baseline.

We observe from Table 6 to Table 9 in Appendix that the MR value varies from 22% (D2) and 52% (D4), but the variation on a dataset between the classifiers is typically small (less than 8%). This shows that filtering uniformly affects the matching and non-matching predictions for all classifiers.

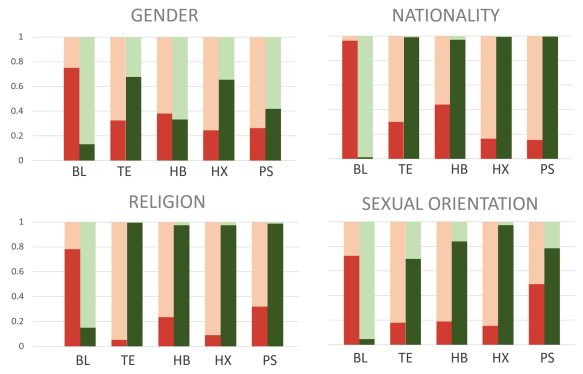

Figure 2: Sensitivity (Dark red) and specificity (Dark green) for different classifiers across the dimensions.

5.4 Risk Statistics

Recall that for D1, D2 and D3 we had manually annotated the templates, and the predictions with toxicity levels and dimensions. Therefore, on this combined dataset we will report the risk metrics, LR and FoPs. Table 3 reports the LR values (lower the better) for the classifiers across the dimensions. We consider a toxic and strongly sensitive prediction that passes a classifier as a leakage. We also report the base rates for such predictions under "None". All classifiers have lower LR than 'None' for all dimensions (except TE on Rel)[10] , which implies that the classifiers indeed help reduce toxic predictions. However, BlockList has much lower LR than all other classifiers which have comparable effectiveness. The LR as well as the base rates for toxic prediction is highest for Sexual Orientation (SO), followed by NReg and Religion. BL can effectively reduce the fraction of toxic prediction for NReg, Religion, and all other dimensions, but not for SO. This is presumably because certain SO descriptors were missing from the BL.

Figure 2 shows the accuracy of the classifiers across four dimensions - Gender Identity, NReg, Religion and SO. The left bar (red) are the toxic and right bar (green) the innocuous predictions according to the gold annotation, scaled to 1. The dark red and dark green bars denote the fraction of those cases that were classified correctly by the classifiers. Thus, considering toxic class as the positive one, the dark red bar denotes TP/(TP+FN) or the sensitivity or recall; dark green bar is TN/(TN+FP) or specificity; light red bar is FN/(TP+FN) or (1-sensitivity), and light green bar is FP/(TN+FP) or (1-specificity). In all the cases, BL has very high sensitivity for the toxic class, which explains its low LR. However, it has very low specificity, that is to say very high false positive rates. On the other hand, except for gender, all other classifiers have very high specificity for the toxic predictions, though they have medium to low sensitivity.

Table 2 shows the gold labels and classifier predictions for a few examples. BL misclassifies the second example as toxic (i.e., overtriggers), whereas TE undertriggers on all toxic examples. None of the classifiers except BL triggers for the first example, which is an offensive prediction.

Fairness of Prediction: In Fig 3 we present the

[10]LR computed based on classifier's final predictions resulting in fewer toxic predictions in absolute terms, even for TE on Rel, compared to no classifier used.

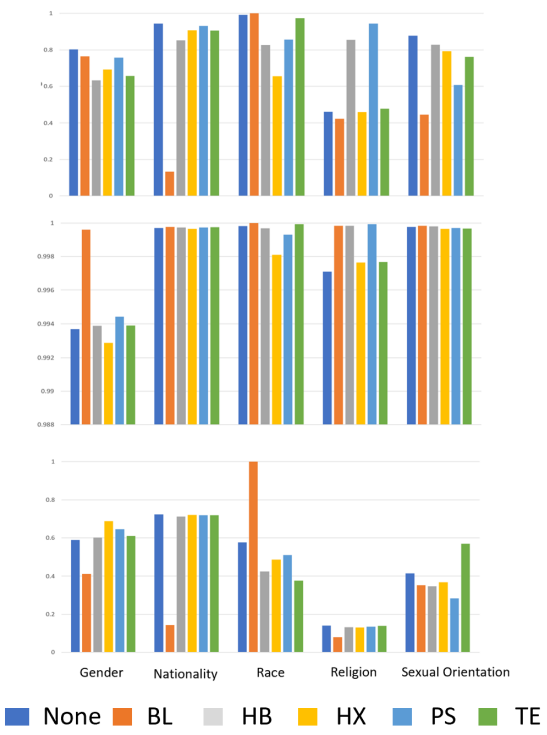

Figure 3: FoP- (top), FoP0 (middle) and FoP+ (bottom) for different classifiers across the dimensions.

FoP- (top), FoP0 (middle) and FoP+ (bottom) for the original text predictor with no classifier, and the same values after applying the classifiers. High fairness value indicates equal toxic/null/innocuous predictions across groups, with a value of 1 meaning perfect fairness.

Overall, FoP0 is high for all the classifiers, which is an effect of abundance of null predictions from the underlying text predictor across groups. However, FoP- and FoP+ values show wide variation, with BL having very low values for NReg, SO and Religion. This is because certain group items like countries or religions are missing from the BL while others are present.

In Fig. 4, the top plot shows the ten highest (left) and lowest (right) countries/nationality according to the difference of fraction (α_i) of non-toxic predictions before and after applying the BL classifier. Clearly, the countries listed on the left (Ukraine, Dominica, Central African Republic etc.) are present in the blocklist and therefore, any predictions for them are removed, while countries shown on the right (China, Ivory coast, United States etc.) are not present. Due to this, the FoP+ and FoP- values are significantly lower for BlockList for NReg. The bottom plot in In Fig. 4 shows the fraction of toxic predictions for religion before and after applying the PS classifier. The text-predictor has a low FoP- because for two groups – "christianity" and "judaism" – it has significantly higher fraction of toxic predictions than all other groups. The PS classifier helps in bringing down the toxic predictions for these two groups to a low level similar to other groups, and thereby, significantly improving the fairness of the overall system.

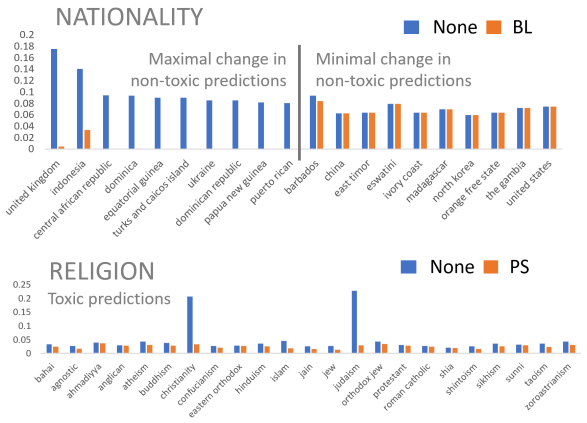

Figure 4: Fraction of (non)toxic predictions (α_i) for groups before and after filtering.

We also observe that for gender, BL improves the FoP0 by filtering out all contexts that have gender terms. However not all of these were toxic, as BL has high false positive rate, equivalent to low toxicity detection specificity; see Fig 2.

6 Conclusion

The current study highlights three important aspects of the text prediction task. First, it is difficult to estimate the risks of a text predictor due to unavailability of appropriate datasets. Second, off-the-shelf toxicity classifiers have higher leakage ratios than what is acceptable. Although Blocklists provide a potential solution, their context-insensitive nature makes them an extremely conservative solution for long form text prediction. Third, LLM based text-predictors are inherently biased towards more toxic/no/innocuous predictions towards certain groups, and while classifiers can improve the fairness of prediction across the groups, this comes at a cost of suppressing most predictions and bringing down the overall usefulness of the system.

Thus, responsible text-prediction at scale offers several research challenges involving complex trade-off between performance on one hand and risks and fairness on the other. Please contact the last author for the checklists and their fine grained annotations created during this work.

7 Limitations

The present study is limited to text prediction in English. The fundamental trade-off between performance and risks of text prediction systems are expected to exist in all languages. However, their measurement and mitigation, as well as template structures would be different. For instance, languages with grammatical genders (e.g., French and Hindi) might require different analysis techniques.

Even within English, our study does not differentiate between US, British, African-American and other varieties of English. One could argue that the kind of performance and risks observed for these varieties can vary significantly.

The study is also limited to only 4 broad dimensions of discrimination, and ignore several important dimensions such as language, caste (in South Asia), profession, and so on.

Finally, an important practical limitation of the study is that while the observations on the CheckList data are informative of the kinds an extent of errors by the classifiers and/or predictor, they do not provide any estimate of the leakage and risks in the real world, where the distribution of data that a user types is expected to differ significantly from the CheckList generated datasets. Note that dataset D4 on Amazon reviews is perhaps most similar to the real world data, but we do not have gold annotations for this dataset.

8 Ethical Considerations

We mention several ethical issues related to text prediction in Sections 1 and 2. The central issue discussed in this paper, that of the trade-off between performance and risks of text prediction, itself has deep ethical connotations. For instance, one might argue that it is ethically incorrect to deploy a system which poses any risks at all. In other words, the trade-off could be resolved in favor of one extreme (which then is no longer a trade-off). We do not take any such position here, and neither try to provide any guidelines on what should be the ideal trade-off for such an application. There are several factors, including but not limited to, the risk-criticality of the application (for instance typing a CV or legal report, vs. a social media comment) and user's personal preferences, that should be considered before settling for a trade-off. Instead, what we would like to highlight through this work is that such a trade-off exist and current technology is unable to completely eradicate harmful predictions. Therefore, at the very least, the service provider/app developer of text prediction systems should be aware of the harms and make an effort to inform the user of such potential harms.

We are also aware that the CheckLists were created by a fairly homogeneous (in terms of religion, nationality and race) set of users. Though we have taken utmost care to sensitize the users about various ethical aspects of fairness, a bias in the annotation or template forms cannot be ruled out. Note that we also use two existing CheckLists which were created by different groups. We observe that the trends are fairly consistent across these datasets. On a related note, the definition of what is toxic or inappropriate can also be debated. Indeed, there were several occasions on which the users designing the templates or annotating the examples did not agree on the appropriateness or severity level. These issues were openly discussed in the larger group (including the authors of this paper) to reach an agreement. We are aware that not everybody will align to the decisions that were taken by our group of volunteers. Thus, the dataset created during this study, when used for further research, should be appropriately aligned to the needs and judgements of the researchers/developers and the tasks at hand. The annotation study is covered under IRB ID 10566 and the consent form is available in the appendix.

Acknowledgements

We would like to thank the following people who helped in creating and annotating the checklists: Kabir Ahuja, Lakshya Agrawal, Sapna Bhardwaj, Harshita Diddee, Pamir Gogoi, Ishani Mondal, Anukriti Kumar, Krithika Ramesh, Abhinav Rao and Hemant Yadav.

References

Kenneth C Arnold, Krysta Chauncey, and Krzysztof Z Gajos. 2020. Predictive text encourages predictable writing. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pages 128–138.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020a. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online. Association for Computational Linguistics.

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020b. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. *arXiv preprint arXiv:2010.12421*.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Advait Bhat, Saaket Agashe, and Anirudha Joshi. 2021. How do people interact with biased text prediction models while writing? In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 116–121.

Shaily Bhatt, Rahul Jain, Sandipan Dandapat, and Sunayana Sitaram. 2021. A case study of efficacy and challenges in practical human-in-loop evaluation of NLP systems using checklist. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 120–130, Online. Association for Computational Linguistics.

JM Buchanan. 1972. The samaritan's dilemma, reprinted in: Buchanan, jm (1977): Freedom in constitutional contract.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2021. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online. Association for Computational Linguistics.

Andrew Dai, Benjamin Lee, Gagan Bansal, Jackie Tsay, Justin Lu, Mia Chen, Shuyuan Zhang, Tim Sohn, Yinan Wang, Yonghui Wu, Yuan Cao, and Zhifeng Chen. 2019. Gmail smart compose: Real-time assisted writing. In *Proceedings of KDD*.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Nestor Garay-Vitoria and Julio Abascal. 2006. Text prediction systems: a survey. *Universal Access in the Information Society*, 4(3):188–203.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3356–3369, Online. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. *arXiv preprint arXiv:2203.14680*.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*.

Rajendra K Jain, Dah-Ming W Chiu, William R Hawe, et al. 1984. A quantitative measure of fairness and discrimination. *Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA*, 21.

Ximing Lu, Sean Welleck, Liwei Jiang, Jack Hessel, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. 2022. Quark: Controllable text generation with reinforced unlearning. *arXiv preprint arXiv:2205.13636*.

Marta Marchiori Manerba and Sara Tonelli. 2021a. Fine-grained fairness analysis of abusive language detection systems with CheckList. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 81–91, Online. Association for Computational Linguistics.

Marta Marchiori Manerba and Sara Tonelli. 2021b. Fine-grained fairness analysis of abusive language detection systems with checklist. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 81–91.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2020. Hatexplain: A benchmark dataset for explainable hate speech detection. *arXiv preprint arXiv:2012.10289*.

Helen Ngo, Cooper Raterink, João GM Araújo, Ivan Zhang, Carol Chen, Adrien Morisot, and Nicholas Frosst. 2021. Mitigating harm in language models with conditional-likelihood filtration. *arXiv preprint arXiv:2108.07790*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia-Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Alec Radford, J Wu, D Amodei, D Amodei, J Clark, M Brundage, and I Sutskever. 2019. Better language models and their implications. 2019. *URL https://openai. com/blog/better-language-models*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *Transactions of the Association for Computational Linguistics*, 9:1408–1424.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. 2021. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.

A Appendix

A.1 Performance Metrics

In this section, we present the detailed results of all the classifiers on all the datasets. We notice that when filtering is enabled, there is a drop in trigger rate and character savings as expected. Overall, we observe a larger drop for blocklist based filtering compared to all the other classifiers for all the datasets. We present the agreement statistics between the classifiers

in Table 5 aggregated over all the datasets, including D4, which could not be annotated for toxicity. The values indicate the sensitivity (recall) of each classifier for the toxic class, against the labels assigned by another classifier. As expected, BlockList has the highest and Perspective API has the least sensitivity. This indirectly hints at the fact that the performance of the classifiers probably is similar on D4 and the other datasets. Overall, there seem to be little agreement between the classifiers.

Full form (Acronyms)
Religion, Race, Ethnicity (RRE)
Nationality, regionality (NReg)
Sexual Orientation and Gender Identity (SOGI)
Offensive to an individual (Off)
In House Checklist (IHCL)
Inter-Annotator Agreement (IAA)
Toxic or Strongly Sensitive (Tox)
Innocuous or Mildly Sensitive (NTox)
Blocklist (BL)
TweetEval (TE)
HateBert (HB)
HateXplain (HX)
Perspective API (PS)
Trigger rate (TR)
Match Rate (MR)
Character Savings (CS)
Leakage Ratio (LR)
Fairness of Prediction (FoP)
FoP for Innocuous predictions (FoP+)
FoP for Toxic predictions (FoP-)
True Positive (TP)
True Negative (TN)
False Negative (FN)
False Positive (FP)

Table 4: Acronyms used in the paper with their respective full forms.

	TE	BL	HB	HX	PS
TE	1	0.74	0.72	0.21	0.05
BL	0.14	1	0.23	0.11	0.01
HB	0.39	0.64	1	0.17	0.03
HX	0.30	0.82	0.48	1	0.04
PS	0.92	0.66	0.91	0.46	1

Table 5: Agreement statistics between the classifiers of the cases detected as toxic by the row classifier, the fraction that is detected as toxic by the column classifier. TE = TweetEval, BL = BlockList, HB = HateBert, HX = HateXplain, PS = Perspective

Classifier	Post-filter Enabled	Pre-filter Enabled	Suggestion Rate	Avg. triggers per 100 words	Match Rate	Char Savings
NA	0	0	2.55	13.23	41.37	64176
Blocklists	1	0	1.86	9.63	42.31	48462
	0	1	1.93	10.00	39.81	46179
	1	1	1.41	7.31	40.35	34466
HateBert	1	0	1.75	9.06	36.42	37261
	0	1	1.68	8.72	36.87	36500
	1	1	1.57	8.16	36.59	33794
HateXplain	1	0	2.29	11.86	39.22	53812
	0	1	2.29	11.86	40.35	55378
	1	1	2.24	11.61	39.76	53368
TweetEval	1	0	1.82	9.41	36.93	39124
	0	1	1.83	9.50	37.28	40023
	1	1	1.75	9.05	37.37	38140
Perspective	1	0	2.26	11.72	39.88	54917
	0	1	2.29	11.85	40.23	55912
	1	1	2.18	11.32	39.55	52451

Table 6: Results on D1

Classifier	Post-filter Enabled	Pre-filter Enabled	Suggestion Rate	Avg. triggers per 100 words	Match Rate	Char Savings
NA	0	0	3.07	17.16	30.68	1110306
Blocklists	1	0	1.81	10.07	26.32	556292
	0	1	1.58	8.82	26.46	494522
	1	1	0.95	5.30	22.42	250776
HateBert	1	0	2.58	14.38	28.30	864047
	0	1	2.71	15.13	28.96	920283
	1	1	2.45	13.67	27.72	802021
HateXplain	1	0	2.86	15.97	29.45	991515
	0	1	2.91	16.23	29.98	1025252
	1	1	2.82	15.74	29.47	977284
TweetEval	1	0	2.74	15.28	28.24	918029
	0	1	2.84	15.85	29.78	997189
	1	1	2.69	14.99	28.36	905389

Table 7: Results on D2

Classifier	Post-filter Enabled	Pre-filter Enabled	Suggestion Rate	Avg. triggers per 100 words	Match Rate	Char Savings
NA	0	0	4.25	23.52	44.43	279877
Blocklists	1	0	2.11	11.71	47.44	145769
	0	1	3.08	17.05	50.14	229935
	1	1	1.75	9.71	50.96	130429
HateBert	1	0	3.89	21.52	44.16	256045
	0	1	3.87	21.42	42.59	246103
	1	1	3.75	20.75	43.01	240878
HateXplain	1	0	4.19	23.20	44.60	277057
	0	1	4.19	23.22	44.64	277747
	1	1	4.17	23.08	44.72	276386
TweetEval	1	0	4.13	22.87	44.56	272867
	0	1	4.13	22.87	44.51	272781
	1	1	4.09	22.64	44.24	268438

Table 8: Results on D3

Classifier	Post-filter Enabled	Pre-filter Enabled	Suggestion Rate	Avg. triggers per 100 words	Match Rate	Char Savings
NA	0	0	4.02	33.62	51.19	296488
Blocklists	1	0	3.19	26.69	51.88	235016
	0	1	3.47	29.04	51.19	255949
	1	1	2.77	23.20	51.91	204299
HateBert	1	0	3.94	32.91	51.22	290332
	0	1	3.94	32.95	51.25	290772
	1	1	3.92	32.80	51.23	289395
HateXplain	1	0	4.01	33.55	51.18	295781
	0	1	4.01	33.55	51.18	295801
	1	1	4.01	33.53	51.18	295664
TweetEval	1	0	3.97	33.16	51.19	292454
	0	1	3.97	33.18	51.21	292679
	1	1	3.96	33.11	51.22	292092

Table 9: Results on D4

Granularity	Cohen's Kappa	Agreement Percentage
All Separate toxicity	0.213	0.44
Toxic + Strongly Sensitive + Mildly Sensitive Vs Innocuous	0.344	0.684
(Toxic & Strongly as class 1), (Innocuous & Mildly as class 2)	0.473	0.739
1 Diff in sensitivity	0.691	0.778

Table 10: Inter annotator agreement scores for the Text predictor for prediction in context. We have calculated the IAA scores at different granularity. 1-Diff is the case when we consider adjacent toxicity values as similar as is usually the case when a subjective evaluation is performed. A high 1-Diff IAA denotes that the annotators mostly agree on the toxicity for the different queries.

	Cohen's Kappa
Severity	0.57
Factuality	0.72

Table 11: Template level IAA scores for IHCL1 dataset for severity and factuality annotations. The scores indicate a moderate to high agreement scores for the different labels.

Classifier	Sensitivity	Leakage Ratio
None	Mildly Sensitive	0.0698
HateXplain	Mildly Sensitive	0.0629
TweetEval	Mildly Sensitive	0.0605
perspective	Mildly Sensitive	0.0539
HateBert	Mildly Sensitive	0.0527
Blocklists	Mildly Sensitive	0.0208
None	Strongly Sensitive	0.0468
TweetEval	Strongly Sensitive	0.0448
perspective	Strongly Sensitive	0.0445
HateXplain	Strongly Sensitive	0.0438
HateBert	Strongly Sensitive	0.0431
Blocklists	Strongly Sensitive	0.0041
None	Toxic	0.1039
HateXplain	Toxic	0.0894
perspective	Toxic	0.0820
TweetEval	Toxic	0.0761
HateBert	Toxic	0.0494
Blocklists	Toxic	0.0139

Table 12: Leakage ratio wrt different sensitivities for each classifier (Pre and Post). The above values include cases which do not fall into the predefined dimensions as stated in the paper but are part of the checklist datasets. In each of the different scenarios we can see that Blocklists perform significantly better than other classifiers. HateBert comes up second and performs better than other classifiers.

A.2 Annotator Consent Form

Microsoft Research Project Participation Consent Form

TITLE OF RESEARCH PROJECT: Responsible AI Data Creation and Annotation

Principal Investigator: Sunayana Sitaram
Co-Investigators: Monojit Choudhury

INTRODUCTION

Thank you for taking the time to consider volunteering in a Microsoft Corporation research project. This form explains what would happen if you joined this research project. Please read it carefully and take as much time as you need. Ask the study team about anything that is not clear. You can ask questions about the study any time.

Participation in this study is voluntary and you will not be penalized if you decide not to take part in the study or if you quit the study later.

PURPOSE

The purpose of this project is to reduce fairness and toxicity harms created by AI systems. We plan to collect data to evaluate current approaches to harm mitigation.

PROCEDURES

During this project, the following will happen: You will be asked to label templates for severity, toxicity and fine-grained category, given a template, lexicon and the coarse grained category for the template. You will also be provided with an example sentence constructed using the template populated with entries from the lexicon. We will give you approximately 100 templates to label and expect each one to take about one minute. You may complete the labeling on your own time over three days. The total amount of time spent should not exceed 120 minutes. Approximately 20 participants will be involved in this study.

PERSONAL INFORMATION AND CONFIDENTIALITY

- **Personal information we collect.** During the project we may collect personal information about you such as name, age, gender, languages known and proficiency in each language.
- **How we use personal information.** The personal information and other data collected during this project will be used primarily to perform research for purposes described in the Purpose and Procedures above. Such information and data, or the results of the research may eventually be used to develop and improve our commercial products, services or technologies.
- **How we store and share your personal information.** Your name and other personal information will not be on the study information we retain; this study information will be identified by a code. The key to the code will be kept separate from your personal and study information, which will be kept in a secured, limited access location.
  Your personal information will be stored for a period of up to 5 years.
  Some people may need to look at your personal information. They include: the researchers involved in this study, who may be Microsoft full time employees and fixed term employees, such as research interns. We will refer to these people as your Study Team. This also includes Institutional Review Boards (IRB), including Microsoft Research's ethics review board. An IRB is a group that reviews the study to protect your rights as a research participant.
  We may choose to share publicly about this study, such as in journal articles, research-focused publications, or presentations at scientific meetings, but your identity will not be disclosed. We will take all steps possible to keep your information confidential. However, we cannot guarantee total confidentiality. For example, your personal information may be given out, if required by law.
- **How you can access and control your per-**

sonal information. If you wish to review or copy any personal information you provided during the study, or if you want us to delete or correct any such data, email your request to the research team at: susitara@microsoft.com. However, once your name or other identifiers have been removed from your information, we will no longer be able to delete it from our records.

For additional information or concerns about how Microsoft handles your personal information, please see the Microsoft Data Privacy Notice (http://go.microsoft.com/fwlink/?LinkId=518021).

MICROSOFT AND CONFIDENTIALITY
The research project and information you learn by participating in the project may be confidential to Microsoft. If the study team discloses confidential information, they will ask you to sign a separate, legally binding document called a Non-Disclosure Agreement (NDA) that asks you to promise to keep study information secret.

BENEFITS AND RISKS
Benefits:
There are no direct benefits to you that might reasonably be expected as a result of being in this study. The research team expects to learn how to build AI systems that are fairer and more inclusive from the results of this research. Furthermore, certain public benefits might be expected as a result of sharing the research results with the greater scientific community.

Risks:
During your participation, you may experience discomfort due to the sensitive nature of the data. Specifically, you will be shown sample sentences that could contain explicit, toxic, or potentially offensive terms. To help reduce such risks, you are free to skip the annotation of any template that makes you feel uncomfortable. You can also stop the annotation at any time and either exit the study or return to it after taking a break.

FUTURE USE OF YOUR IDENTIFIABLE INFORMATION
We may use your data in the future. Any data you contribute as part of this study will be stripped of any identifiers or other information that could be used to identify you, as disclosed previously in this consent form. After such removal, the information could be used for future research studies or distributed to another investigator for future research studies without your (or your legally authorized representative's) additional informed consent.

PAYMENT FOR PARTICIPATION
You will not be paid to take part in this study. Your data may be used to make new products, tests, or findings. These may have value and may be developed and owned by Microsoft and/or others. If this happens, there are no plans to pay you.

PARTICIPATION
Taking part in research is always a choice. If you decide to be in the study, you can change your mind at any time without affecting any rights including payment to which you would otherwise be entitled. If you decide to withdraw, you should contact the person in charge of this study. The study team may use study data already collected from you, however, you may ask for it to be removed when you leave. Microsoft or the person in charge of this study may discontinue the study or your individual participation in the study at any time without your consent for reasons including:

- it is discovered that you do not meet study requirements
- the study is canceled
- administrative reasons

CONTACT INFORMATION
Should you have any questions concerning this project, or if you are injured as a result of being in this study, please contact; Sunayana Sitaram, at (Telephone Number removed for privacy) or `susitara@microsoft.com` (email). Should you have any questions about your rights as a research subject, please contact the Microsoft Research Ethics Review Program at `MSRStudyfeedback@microsoft.com`

CONSENT
By completing this form, you confirm that this study was explained to you, you had a chance to ask questions before beginning this study, and all your questions were answered satisfactorily. At any time, you may ask other questions. By completing

this form, you voluntarily consent to participate, and you do not give up any legal rights you have as a study participant.

Please confirm your consent by completing the bottom of this form. If you would like to keep a copy of this form, please print or save one now. On behalf of Microsoft, we thank you for your contribution and look forward to your research session.

Optional: Initial here if we may contact you in the future to request consent for uses of your identifiable data that are not covered in this consent form.

Initial here _____

Optional: Initial here if we may contact you in the future with information about follow-up or other future studies.

Initial here _____
Participant's Name _____
Date _____