BERT-β: A Proactive Probabilistic Approach to Text Moderation

Fei Tan, Yifan Hu, Kevin Yen, Changwei Hu
Yahoo Research, New York, NY, USA
{fei.tan, yifanhu, kevinyen, changweih}@yahooinc.com

Abstract

Text moderation for user generated content, which helps to promote healthy interaction among users, has been widely studied and many machine learning models have been proposed. In this work, we explore an alternative perspective by augmenting reactive reviews with proactive forecasting. Specifically, we propose a new concept *text toxicity propensity* to characterize the extent to which a text tends to attract toxic comments. Beta regression is then introduced to do the probabilistic modeling, which is demonstrated to function well in comprehensive experiments. We also propose an explanation method to communicate the model decision clearly. Both propensity scoring and interpretation benefit text moderation in a novel manner. Finally, the proposed scaling mechanism for the linear model offers useful insights beyond this work.

1 Introduction

Text moderation is essential for maintaining a non-toxic online community for media platforms (Nobata et al., 2016). Many efforts from both academia and industry have been made to address this critical problem. Recently, the most prototypical thread is to do sophisticated feature engineering or develop powerful learning algorithms (Nobata et al., 2016; Badjatiya et al., 2017; Bodapati et al., 2019; Tan et al., 2020; Tran et al., 2020). Automatic comment moderation schemes plus human review are certainly the cornerstone of the fight against toxicity.

These existing works, however, are *reactive* approaches to handling user generated text in response to the publication of new articles. In this paper, we revisit this challenge from a *proactive* perspective. Specifically, we introduce a novel concept *text toxicity propensity* to quantify how likely an article is prone to incur toxic comments. This is a proactive outlook index for news articles prior to the publication, which differs radically from the existing reactive approaches to comments.

In this context, reactive describes comment-level moderation algorithms after the publication of news articles (e.g., Perspective (Perspectiveapi)), which quantifies whether comments are toxic and should be taken down or sent for human review. Proactive emphasizes article-level moderation effort before the publication (without access to comments), which forecasts how likely articles are to attract toxic comments in the future and gives suggestions (e.g., rephrase news articles properly) in advance. Our work can be viewed as the first machine learning effort for a proactive stance against toxicity.

Formally, we propose a probabilistic approach based on Beta distribution (Beta) to regress article toxicity propensity on article text. For previously published news articles with comments, we take the average of comments' toxicity scores as the ground-truth label for model learning. The effectiveness of this approach is shown in both test set and human labeling. We also develop a scheme that can provide convincing explanation to the decision of the deep learning model.

2 Related Works

To our best knowledge, there's no prior research thread on proactive text moderation. Nonetheless, many reactive approaches have been explored including hand-crafted feature engineering (Chen et al., 2012; Warner and Hirschberg, 2012; Nobata et al., 2016), neural networks (Badjatiya et al., 2017; Pavlopoulos et al., 2017; Agrawal and Awekar, 2018; Zhang et al., 2018) and transformer variants (Bodapati et al., 2019; Tan et al., 2020).

Recently, context, in the form of parent posts, has been studied but it is only viewed as regular text snippets for lifting the performance of toxicity classifiers (Pavlopoulos et al., 2020) while screening posts. Our work instead focuses on predicting the proactive toxicity propensity of articles before

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 8667–8675
November 7–11, 2021.

they receive user comments.

Beta distribution is usually utilized as a priori in Bayesian statistics. The most popular example in natural language processing is Topic Model (Blei et al., 2003), where the multivariate version of beta distribution (a.k.a. Dirichlet distribution) generates parameters of mixture models. Beta regression is originally proposed for modeling rate and proportion data (Ferrari and Cribari-Neto, 2004) by parameterizing mean and dispersion and regressing parameters of interest. It has been applied to evaluate grid search parameters in optimization (McKinney-Bock and Bedrick, 2019), model emotional dimensions (Aggarwal et al., 2020) and statistical processes of child-adult linguistic coordination and alignment (Misiek et al., 2020).

3 Beta Regression

In this work, both comment toxicity score and the derived article toxicity propensity score (to be detailed in the subsequent section 4.1) range from 0 to 1. Empirically, their distributions exhibit an asymmetry and may not be modelled well with the Gaussian distribution (Figs. 2 and 3 of Appendix A). Furthermore, comment toxicity score distributions of individual articles vary with article content as shown in Fig. 3 of Appendix A. Modelling the entire distribution of an article comment toxicity scores is thus a reasonable approach. Beta distribution is very flexible and it can model quite a wide range of well-known distribution families from symmetric uniform ($\alpha = \beta = 1$) and bell-shaped distributions ($\alpha = \beta = 2$) to asymmetric shapes ($\alpha \neq \beta$).

In this context, the toxicity propensity score y is assumed to follow the Beta distribution with probability density function (pdf):

$$p(y|\alpha,\beta) = \text{Beta}(\alpha,\beta) = \frac{y^{\alpha-1}(1-y)^{\beta-1}}{B(\alpha,\beta)} \quad (1)$$

where α and β are two positive shape parameters to control the distribution. $B(\alpha,\beta)$ is the normalization constant and support y meets $y \in [0,1]$. Eq. 1 holds the probabilistic randomness given α and β, we thus impose a regression structure of them on text content.

Formally, given a training set $\mathcal{D} = \{(\mathbf{x_n}, y_n)\}_{n=1}^{N}$ with raw text feature vector $\mathbf{x_n}$ and label y_n for sample n, we apply feature engineering or text embedding $g(\cdot)$ and then regress $\alpha_n(>0)$ and $\beta_n(>0)$ on $g(\mathbf{x_n})$ respectively as

$$\begin{aligned} \log(\alpha_n) &= f_\alpha(g(\boldsymbol{x}_n)) \\ \log(\beta_n) &= f_\beta(g(\boldsymbol{x}_n)) \end{aligned} \quad (2)$$

where $f_\alpha(\cdot)$ and $f_\beta(\cdot)$ are learned jointly. $g(\cdot)$ can be either pre-fixed or learned together with $f_\alpha(\cdot)$ and $f_\beta(\cdot)$, which is detailed in the subsequent section. Specifically, the learning procedure of $f_\alpha(\cdot)$, $f_\beta(\cdot)$ and $g(\cdot)$ (if applicable) is to minimize loss

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N} \log\big(p(y_n|\alpha_n,\beta_n)\big) \quad (3)$$

Substituting Eqs. 1 and 2 into it gives the final objective function.

In the inference phase, with learned $f_\alpha(\cdot)$, $f_\beta(\cdot)$ and $g(\cdot)$, α_m and β_m for a new sample $\mathbf{x_m}$ can be readily derived from Eq. 2. We take the mean of Eq. 1 as a point estimator: $\widehat{y_m} = \frac{\alpha_m}{\alpha_m+\beta_m}$ because we are predicting the average toxicity.

4 Experiments

4.1 Dataset

We collect a dataset of articles published on Yahoo media outlets, which are all written in English. We also exclude articles with low comment volume to make the distribution learning reliable. The number of comments for 99% of the analyzed articles lie in [10, 8K], with 25% quantile of 20, median of 50 and mean of 448. The employed dataset is then split into training, validation and test parts based on the publishing date with ratio of 8:1:1 as described in Table 1. It's worthwhile to note that input text $\mathbf{x_n}$ is the concatenation of article title and text body. The toxicity propensity score y_n of article n is defined as the average toxicity score of all associated comments. Comments are scored by Google's Perspective (Perspectiveapi), which lies in [0, 1]. Perspective intakes user generated text and outputs toxicity probability. It's a convolutional neural net (Noever, 2018) trained on a comments dataset [1] of wikipedia labeled by multiple people per majority rule.

Table 1: Basic statistics of dataset breakdown

	Training	Validation	Test
Sample Size	536,711	70,946	70,946
Publishing Date	2004 - 05/2020	05/2020-06/2020	06/2020-09/2020

[1] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

4.2 Experiment Setup

In Eq. 2, we set both $f_\alpha(\cdot)$ and $f_\beta(\cdot)$ to single-layer neural networks. For $g(\cdot)$, we experiment with either Bag of Words (BOW) or BERT embedding (BERT) (Devlin et al., 2019). Specifically, we take uni-gram and bi-gram words sequence and compute the corresponding Term Frequency-Inverse Document Frequency (TF-IDF) vectors, which leads to around 5.8 million tokens for BOW. For BERT, we take the base version and then fine-tune $f_\alpha(\cdot)$ and $f_\beta(\cdot)$ on top of the [CLS] embedding, which ends up with 110 million parameters. If input text exceeds the maximum length (510 as [CLS] and [SEP] are reserved), we adopt a simple yet effective truncation scheme (Sun et al., 2019). Specifically, we empirically select the first 128 and the last 382 tokens for long text. The rationale is that the informative snippets are more likely to reside in the beginning and end. Batch size is 16 and learning rate is $1e-5$ for Adam optimizer (Kingma and Ba, 2015). They are called BOW-β and BERT-β for short.

4.3 Baseline Methods and Metrics

We compare with the linear regression method using BOW features, as well as the BERT base model. Both are combined with one of two loss functions, Mean Absolute Error (MAE) or Mean Squared Error (MSE). We call them BOW-MAE, BOW-MSE, BERT-MAE, BERT-MSE, respectively. The experiment settings are same as the Beta regression.

Since we are interested in identifying articles of high toxicity propensity, we want to make sure that an article with high average toxicity is ranked higher than one with low propensity. Thus in addition to mean absolute error, root mean squared error (RMSE) and AUC@Precision-Recall curves (AUC@PR), we measure performance using two ranking metrics, Kendall coefficient (Kendall) and Spearman's coefficient (Spearman).

4.4 Results

We perform evaluation on the whole test set and on human labels.

4.4.1 Test Set

Table 2 details the performance comparisons. Overall, Beta regression stands out across different metrics regardless of feature engineering due to its modeling flexibility. BERT-based methods also outperform BOW ones in terms of feature engineering and representation. This is reasonable as the former has 20 times as large parameters as the latter and offers the contextual embedding. Interestingly, MAE and MSE schemes don't achieve the minimum MAE and RMSE although they are dedicated to this goal, which might result from the limitation of point estimator.

Table 2: Performance comparisons on test set

	Kendall		Spearman		MAE		RMSE	
	val	test	val	test	val	test	val	test
BOW-MAE	0.332	0.314	0.488	0.464	0.076	0.081	0.095	0.100
BOW-MSE	0.428	0.402	0.606	0.574	0.057	0.063	0.076	0.084
BOW-β	0.437	0.413	0.617	0.589	**0.056**	**0.061**	**0.075**	**0.081**
BERT-MAE	0.360	0.333	0.525	0.489	0.072	0.076	0.092	0.095
BERT-MSE	0.442	0.423	0.621	0.598	0.070	0.073	0.089	0.093
BERT-β	**0.462**	**0.440**	**0.642**	**0.617**	**0.056**	0.065	**0.075**	0.085

4.4.2 Human Labels

As labels are derived from machine, we want a sanity check to ensure that the model decision conforms to human intuition. Namely, when the model classifies an article as having high toxicity propensity, we want to make sure that it correlates well with human judgement. To this end, we divide test set into 10 equal buckets with an interval of 0.1 and merge the last 4 buckets into [0.6, 1] due to much fewer articles with score being above 0.6 (as shown in Fig. 2). There are thus a total of 7 buckets [0, 0.1), [0.1, 0.2), [0.2, 0.3), [0.3, 0.4), [0.4, 0.5), [0.5, 0.6) and [0.6, 1.0]. We then randomly take 100 samples per bucket and set aside 10% for human training and the remaining are labelled by the human judges as the benchmark set. We recruit two groups of people for independent annotation, which are required to pick one from five levels (a reasonable balance between smoothness and accuracy for manually labeling toxicity propensity per judges' suggestion) to describe the propensity extent to which an article is likely to attract toxic comments: Very Unlikely (VU), Unlikely (U), Neutral (N), Likely (L) and Very Likely (VL). Table 3 is the confusion matrix showing how much two groups of human judges agree with each other.

Table 3: Confusion matrix of two groups of human annotation G1 and G2

G1 \ G2	VU	U	N	L	VL	Total
VU	89	28	0	23	1	141
U	30	26	0	37	3	96
N	0	1	0	3	1	5
L	31	25	0	34	56	146
VL	18	21	0	87	124	250
Total	168	101	0	184	185	638

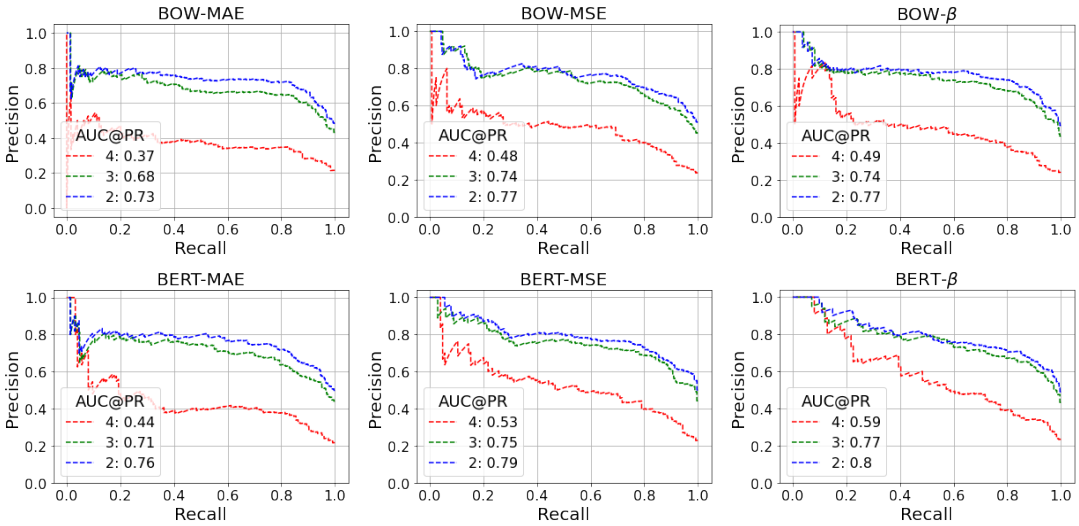

Figure 1: PR curves on human labelled data.

Interestingly, even humans don't agree with each other on all examples. Roughly, the perception of the two groups is consist on 74% samples (top left and bottom right boxes in Table 3). Moreover, Cohen's Kappa is about 0.23 by taking expected chance agreement into account [2]. In light of this, we jointly score the set by assigning $-2, -1, 0, 1$ and 2 to VU, U, N, L and VL, respectively. Since each article has two labels, the addition gives an integer score interval [-4,4]. Table 4 reports the performance with human labels as the ground truth, which confirms the previous findings that BERT-β performs the best. Additionally, we pick scores 2, 3 and 4 as thresholds to monitor precision and recall curves (Fig. 1). Likewise, the proposed schemes achieve compelling performance widely.

Taken together, our probabilistic methods agree more with both machine and human judgements.

Table 4: Performance on human labelled set

	BOW-MAE	BOW-MSE	BOW-β
Kendall	0.402	0.481	0.491
Spearman	0.562	0.635	0.649
	BERT-MAE	BERT-MSE	BERT-β
Kendall	0.441	0.508	**0.522**
Spearman	0.599	0.665	**0.679**

4.5 Explanation

As we focus on the pre-publication text moderation, a reasonable explanation is an essential step to convince stake-holders of subsequent operations. For BERT-β explanation, we adopt gradient-based saliency map variants from computer vision (Simonyan et al., 2013; Shrikumar et al., 2017). We compute the gradient $\nabla f(\mathbf{x})$ with respect to input tokens embedding $\mathbf{e}(\mathbf{x})$, where $f(\mathbf{x}) = \alpha(\mathbf{x})/(\alpha(\mathbf{x}) + \beta(\mathbf{x}))$ is the mean prediction for sample $\mathbf{x}$ (Section 3), and $\mathbf{x} = (t_1, t_2, \cdots, t_L)$ where $t_l (l = 1, 2, \cdots, L)$ is a single token. The element of $\nabla f(\mathbf{x})$ is partial derivative $\frac{\partial f}{\partial \mathbf{e}(t_l)}(\mathbf{x})$ to measure the token-level contribution to the scoring. The explanation is conducted by assuming the article is controversial, and we want to figure out which words cause some comments to be toxic. So it also makes sense to maximize the maximum toxicity of the comments. We thus experiment with $f(\mathbf{x}) = (\alpha(\mathbf{x}) - 1)/(\alpha(\mathbf{x}) + \beta(\mathbf{x}) - 2)$, which is the mode (corresponding to the peak in the PDF of Beta distribution) under reasonable assumption $(\alpha, \beta > 1)$. We denote the resulting scheme by subscript "mode".

For saliency map (SM) (Simonyan et al., 2013), the metric is $\|\frac{\partial f}{\partial \mathbf{e}(t_l)}(\mathbf{x})\|_2$ without direction. A variant is dot product (DP) between token embedding and gradient element $\mathbf{e}(t_l)^T \cdot \frac{\partial f}{\partial \mathbf{e}(t_l)}(\mathbf{x})$ with direction (Shrikumar et al., 2017). We also propose a hybrid (HB) scheme to take magnitude of SM and direction of DP to form a new metric. We perform an ablation study (AS) to delete single token t_l alternately and then compute the score discrepancy between original $\mathbf{x}$ and $\mathbf{x}_{\neg l}$ as well. As a reference, we examine the regression coefficients (RC) of linear BOW-MSE, which are easy to check for explaining the contribution of corresponding words.

A few well-trained human judges are recruited to

[2] https://en.wikipedia.org/wiki/Cohen%27s_kappa

tag k (example-specific, determined by annotators) most important words. We then prioritize tokens with different metrics and pick top k ones as candidates. Hit rate (proportion of human annotated tokens covered by schemes) is used to compare different tools. We take $1,000$ examples for human review and compute the average hit rate, as compared in Table 5.

All schemes for BERT-β are much better than linear scheme RC, which is consistent with the predictive performance discrepancy. SM and HB are close and outperform black-box ablation study, which implies the valuable role of model-aware gradients in the explanation. DP is inferior to AS and seems not consistent with human annotation as well as other gradient based methods. In practice, we take SM for the explanation (Appendix B) due to its out-performance and simplicity. As expected, mode (SM_{mode}) covers more annotated words than mean (SM) on average (more discussions in Appendix C).

Table 5: Performance (average hit rate) comparison

SM	DP	HB	AS	RC	SM_{mode}
0.549	0.430	0.543	0.467	0.382	**0.553**

5 Additional Study

Linear regression (BOW-MSE) is inferior to BERT-β. Nonetheless, it is much faster in training, inference and explanation as it is about 20 times as small as BERT-β. Thus, we investigate if the performance of the linear model could be improved for industrial deployment.

Inspired by NBSVM (Wang and Manning, 2012), we scale TF-IDF vectors of BOW-MSE by a weight vector defined as $\mathbf{w} = \frac{\left(\tau(\mathbf{X})-\overline{\tau(\mathbf{X})}\right)^T\left(\mathbf{y}-\overline{\mathbf{y}}\right)}{\left\|\tau(\mathbf{X})-\overline{\tau(\mathbf{X})}\right\|^2}$ where $\mathbf{X}$ is the training corpus. $\mathbf{y} \in [0,1]^{N\times 1}$ and $\tau(\mathbf{X}) \in \mathbb{Z}^{N\times M}$ ($M = 5.8$ million) are the training labels and TF-IDF matrix. $\overline{\mathbf{y}}$ and $\overline{\tau(\mathbf{X})}$ are their column-wise means. The pre-computed $\mathbf{w}$ can be viewed as a surrogate of the regression coefficient for the linear regression problem, which is used to scale TF-IDF of BOW-MSE in both training and inference phases. We call it Naive Bayes Linear Regression (NBLR) for short.

The scaling benefits the performance, as compared in Table 6. As can be seen, NBLR improves upon BOW-MSE significantly, although it is not as good as BERT-β.

Table 6: Performance on test set and human labels

	Test Set		Human Label	
	Kendall	Spearman	Kendall	Spearman
BOW-MSE	0.402	0.574	0.481	0.635
NBLR	**0.413** (+.011)	**0.588** (+.014)	**0.501** (+.020)	**0.656** (+.021)
BERT-β	0.440	0.617	0.522	0.679
	Human Label			
	AUC@PR at 2	AUC@PR at 3	AUC@PR at 4	
BOW-MSE	0.77	0.74	0.48	
NBLR	**0.78** (+.01)	**0.76** (+.02)	**0.53** (+.05)	
BERT-β	0.80	0.77	0.59	

6 Discussion and Future Work

Our work can benefit text moderation. The proactive propensity offers a toxicity outlook for comments, which could be utilized in multiple ways. For example, stricter moderation rules are enforced for articles that are predicted to have a high toxicity propensity. Furthermore, the propensity could be used as an additional feature for the downstream reactive toxicity recognition models, as well as for allocation of appropriate human resources.

The explanation tool can also be used to remind editors to rephrase some controversial words to mitigate the odds of attracting toxic comments. Text moderation is an important yet challenging task, our proactive work is attempting to open up a new perspective to augment the traditional reactive procedure. Our current model, however, is not perfect as shown by article b in Fig. 3 of Appendix A where the learned distribution doesn't fit well the observed histogram. Technically, NBLR is an encouraging lightweight extension to Linear Regression. Likewise, we will continue to work towards the improvement of the non-linear Beta regression.

7 Conclusion

We approach text moderation by developing a well-motivated probabilistic model to learn a proactive toxicity propensity. An explanation scheme is also proposed to visually explain the connection between this new prospective score, and text content. Our experiment shows the superior performance of the proposed BERT-β algorithm, compared with a number of baselines, in predicting both the average toxicity score, and the human judgement.

References

Jai Aggarwal, Ella Rabinovich, and Suzanne Stevenson. 2020. Exploration of gender differences in covid-19 discourse on reddit. In Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020.

Sweta Agrawal and Amit Awekar. 2018. Deep learning for detecting cyberbullying across multiple social media platforms. In European Conference on Information Retrieval, pages 141–153. Springer.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In Proceedings of the 26th International Conference on World Wide Web Companion, pages 759–760.

Wiki Distribution Beta. https://en.wikipedia.org/wiki/Beta_distribution.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. the Journal of machine Learning research, 3:993–1022.

Sravan Babu Bodapati, Spandana Gella, Kasturi Bhattacharjee, and Yaser Al-Onaizan. 2019. Neural word decomposition models for abusive language detection. arXiv preprint arXiv:1910.01043.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom), pages 71–80. IEEE.

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

Silvia Ferrari and Francisco Cribari-Neto. 2004. Beta regression for modelling rates and proportions. Journal of applied statistics, 31(7):799–815.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In ICLR (Poster).

Katy McKinney-Bock and Steven Bedrick. 2019. Classification of semantic paraphasias: Optimization of a word embedding model. In Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP, pages 52–62.

Thomas Misiek, Benoit Favre, and Abdellah Fourtassi. 2020. Development of multi-level linguistic alignment in child-adult conversations. In Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, pages 54–58.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In Proceedings of the 25th international conference on world wide web, pages 145–153.

David Noever. 2018. Machine learning suites for online toxicity detection. arXiv preprint arXiv:1810.01869.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017. Deeper attention to abusive user content moderation. In Proceedings of the 2017 conference on empirical methods in natural language processing, pages 1125–1135.

John Pavlopoulos, Jeffrey Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity detection: Does context really matter? In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4296–4305.

Google Perspectiveapi. https://www.perspectiveapi.com.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In International Conference on Machine Learning, pages 3145–3153. PMLR.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In China National Conference on Chinese Computational Linguistics, pages 194–206. Springer.

Fei Tan, Yifan Hu, Changwei Hu, Keqian Li, and Kevin Yen. 2020. Tnt: Text normalization based pre-training of transformers for content moderation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4735–4741.

Thanh Tran, Yifan Hu, Changwei Hu, Kevin Yen, Fei Tan, Kyumin Lee, and Se Rim Park. 2020. Habertor: An efficient and effective deep hatespeech detector. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7486–7502.

Sida I Wang and Christopher D Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 90–94.

William Warner and Julia Hirschberg. 2012. Detecting hate speech on the world wide web. In Proceedings of the Second Workshop on Language in Social Media, pages 19–26. Association for Computational Linguistics.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network. In

European Semantic Web Conference, pages 745–760. Springer.

A Toxicity Score and Beta Distribution

The distribution of news articles' toxicity propensity score is reported in Fig. 2. Comment score distributions of two articles with predictive distribution are given in Fig. 3.

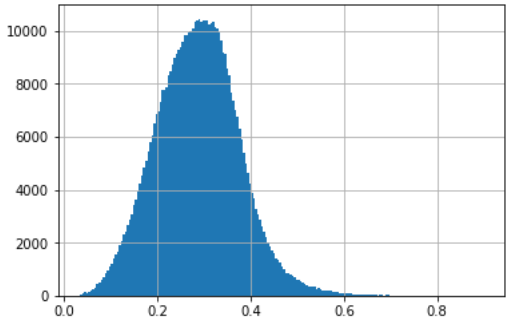

Figure 2: Toxicity propensity score (mean comment toxicity scores) distribution of news articles.

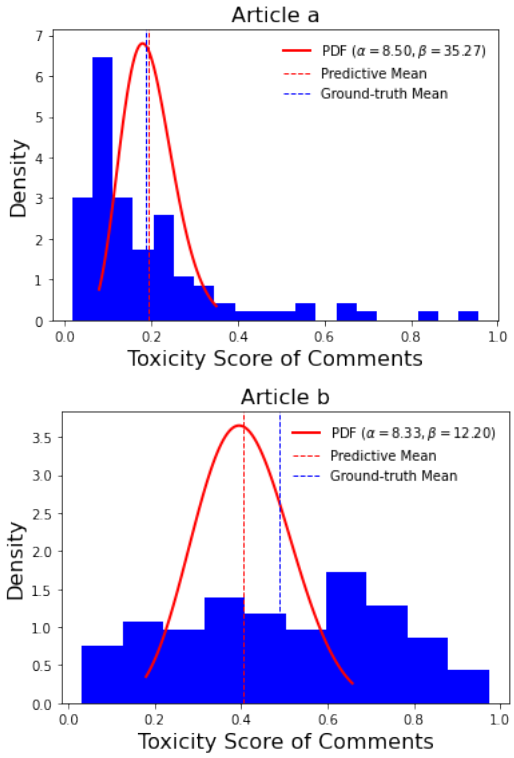

Figure 3: Toxicity score histogram density of comments for articles a (top) and b (bottom). Solid red lines represent predictive beta distribution for individual articles.

B SM Explanation Examples

We pick two samples from the test set and then leverage SM in section 4.5 to highlight key words for the illustration purpose, as shown in Fig. 4. The color intensity is proportional to the normalized saliency map value. The darker the color of a token is, the more important it's to the scoring. There's also a positional bias towards the first sentence as it's the article title.

C BERT-β mode

We also explore the mode of BERT-β as a point estimator and compare it with the mean. Table 7 details the performance discrepancy between the test set and human labels. For the toxicity propensity prediction in the test set, it does make sense for mean to slightly outperform mode as ground-truth labels are the score mean of comments. When it comes to human labels and explanation, people annotate news articles based on the perceived controversial words most likely to incur toxic comments. Mode is thus able to capture the worst case better and agrees more with human annotations. This finding is in line with the better explanation performance, as compared in Table 5.

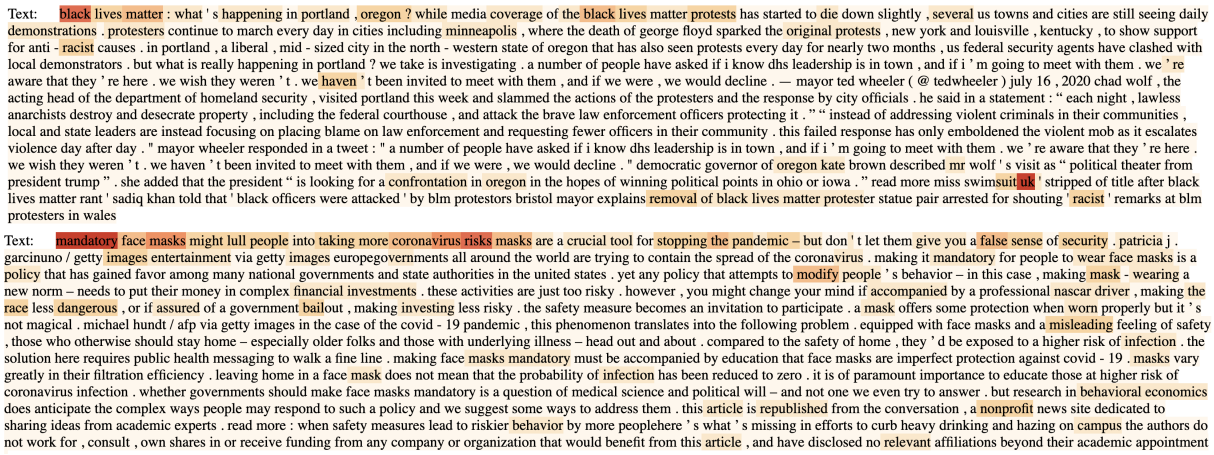

Figure 4: Model explanation examples.

Table 7: Performance of BERT-β point estimators on test set and human labels

	Test Set				Human Label	
	Kendall	Spearman	RMSE	MAE	Kendall	Spearman
mean	**0.440**	**0.617**	**0.065**	**0.085**	0.522	0.679
mode	0.439	0.614	0.077	0.099	**0.543**	**0.704**
	Human Label					
	AUC@PR at 2	AUC@PR at 3	AUC@PR at 4			
mean	0.8	0.77	0.59			
mode	**0.82**	**0.79**	**0.63**			