Deploying Unified BERT Moderation Model for E-Commerce Reviews

Ravindra Nayak N
Flipkart
Bangalore, India
ravindra.n@flipkart.com

Nikesh Garera
Flipkart
Bangalore, India
nikesh.garera@flipkart.com

Abstract

Moderation of user-generated e-commerce content has become crucial due to the large and diverse user base on the platforms. Product reviews and ratings have become an integral part of the shopping experience to build trust among users. Due to the high volume of reviews generated on a vast catalog of products, manual moderation is infeasible, making machine moderation a necessity. In this work, we described our deployed system and models for automated moderation of user-generated content. At the heart of our approach, we outline several rejection reasons for review & rating moderation and explore a unified BERT model to moderate them. We convey the importance of product vertical embeddings for the relevancy of the review for a given product and highlight the advantages of pre-training the BERT models with monolingual data to cope with the domain gap in the absence of huge labelled datasets. We observe a 4.78% F1 increase with less labelled data and a 2.57% increase in F1 score on the review data compared to the publicly available BERT-based models. Our best model In-House-BERT-vertical sends only 5.89% of total reviews to manual moderation and has been deployed in production serving live traffic for millions of users.

1 Introduction

The Internet has enabled the easy flow of information across the globe, but it has its downside too. It has led to increased hate speech and abusive communication(Veglis, 2014). It is necessary to prevent people from accessing our personal information, as it can be used for malicious purposes. The platforms that enable people to communicate and convey their opinions are also responsible for preventing profane content from affecting their users. So such platforms must have strict guidelines and strong moderation of user-generated content.

The downside of manual moderation involves inconsistency in labelling, the inability to real-time moderation, lack of domain knowledge, and multilingual vocabulary. Due to the immense scale of the data that has been ingested on such platforms, auto-moderation becomes vital as manual moderation is not economical.

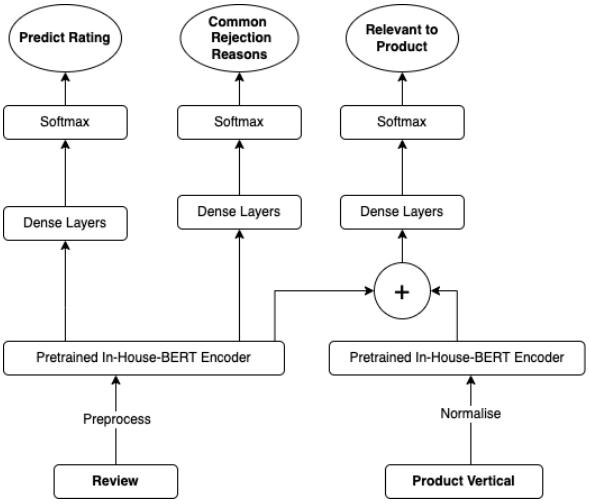

Figure 1: Model Architecture

The E-commerce domain accepts multi-modal data such as text, images, and videos (Ueta et al., 2020). It is crucial to moderate them before the platform users consume the data. This paper mainly concentrates on the moderation of textual review data. Reviews and ratings build trust in the product and help platforms promote good products (Kumar, 2017). Thus eliminating reviews that do not talk about the product becomes necessary. The aim of moderating reviews is not only to detect abusive or hate speech content but also to check whether a review follows other guidelines before posting it. Before rejecting a review, it is necessary to predict the reason for rejection as feedback to the users.

We have multiple reasons for rejecting a review. These are mentioned in Table 1 along with examples. Commonly used moderation reasons include detecting profane and hate speech content (Pavlopoulos et al., 2017; Glazkova et al., 2021).

Proceedings of EMNLP 2022 Industry Track, pages 550–557
December 9–11, 2020.

Table 1: Rejection reasons with examples

Moderation Reasons	**Example 1**	**Example 2**
approved	Just go for the good quality !! I am happy	Ok product but top coat was bad
Poorly formatted content	?????!	Nce prdddct, mast buy !!
Irrelevant review for the product	Thank you, good luck (for "watch" vertical)	I have not used it yet, dont know.. (for "mobile" vertical)
Mismatch between user-provided rating & sentiment of the review	Poor quality product (user gave a rating of 5)	Most value for money product among in ths range (user gave a rating of 1)
Profane/abusive content	Product is bull sh*t	I hate you all ******
Contains Email address(es)	abcd@gmail.co.in	Mail me raa @ outlook.com !!
Contains HTML/CSS character(s)	THis is good buy.	 Link
Contains Phone Number(s)	Nine 7383 S892	Contact 92992**009 for more info
Contains URL(s)	https://docs.google.com/forms/d/e/ Please fill this form for my friend and share	https://yotu.be/uuY Unboxing video for the product

In our work, we introduce new rejection reasons (Table 1) to detect poorly formatted content, and irrelevant reviews for the product, and detect personal information like email addresses, phone numbers, and URLs. The mismatch between the rating and the sentiment of the review creates confusion in the buyer's mind (Kumar, 2017). Hence, we predict the rating to eliminate the reviews with such a mismatch.

We start with regex parsing and list-based matching methods. These are not robust enough to capture all rejection reasons. We train a BERT (Devlin et al., 2019) based model, which predicts the rejection reasons and the rating for the given comment. We build a unified model which adheres to the review moderation guidelines set by the platform.

Publicly available base BERT(Devlin et al., 2019) is considered the baseline, and we try different architectures and configurations that help in better moderation. We use a pre-trained In-House-BERT model, which has been trained on monolingual review text and product descriptions. Pre-training helps create generic representations and adds robustness to the model (Erhan et al., 2010). We freeze embedding and initial 8 layers (Lee et al., 2019) as it helps in faster training time without degrading the model's performance. We use product vertical / category names as an embedding to help understand the relevance of the review for that given product. We augment data with various obfuscations and noise to make the model robust to hard rejection reasons such as detecting profane/abusive content. Finally, we incorporate all these techniques to fine-tune a unified In-House-BERT moderation model to obtain an F1 score of, which is 2.57% improvement on the publicly available baseline models.

There are multiple scenarios where an auto-moderation model may fail, such as significantly morphed text, sarcastic content, or unseen data. In such a scenario, we fall back to manual moderation (Link et al., 2016). Our aim is not to fully eliminate manual moderation but instead to decrease the volume of data that goes to the moderators. When the model is not confident of its predictions, we send it for manual checks before approving it, considering it as the last line of defence.

Our major contributions from the work include:

1. Overview of our deployed text moderation system for e-commerce product reviews.

2. Unified BERT model architecture combined with deterministic approaches for moderation.

3. Demonstrating the benefits of pre-training In-House-BERT models when labelled data is scarce.

4. Illustrating the merits of adding product vertical embeddings to relevant classification heads.

5. Exhibiting the importance of using hybrid approaches with the machine and manual moderation in inference setup.

2 Related work

Moderation use cases started as early as the email era and the need increased with the rise of social-media(Veglis, 2014). Traditionally hand-crafted rules were used along with basic profane word list matching. People started finding different ways to format and morph the text to bypass these systems. This paved the way sophisticated approaches with machine learning algorithms like TF-IDF (Gaydhani et al., 2018), SVM (Veloso et al., 2007) and deep learning algorithms (Saude et al., 2014; Badjatiya et al., 2017; Korencic et al., 2021; Turki and Roy, 2022).

Most of the research has been around detection of profane, hate-speech and abuse detection in the user-generated content (Pavlopoulos et al., 2017; Caselli et al., 2020; Glazkova et al., 2021). To the best of our knowledge, we haven't found any guidelines for review moderation other than detecting profane content and fake reviews(Danilchenko et al., 2022; Jindal and Liu, 2007; Rastogi and Mehrotra, 2017). We introduce sophisticated moderation guidelines for reviews and ratings in the e-commerce domain.

Dataset creation is a huge challenge as there will be imbalanced classes across various rejection reasons. Huge datasets are available for profane and hate speech content which can be curated from Twitter, Reddit, and other social media texts (Qian et al., 2019; Hee et al., 2015). These include monolingual, multilingual (i Orts, 2019; Bhattacharya et al., 2020) and code-mixed data (Bohra et al., 2018). Emojis are an important part of expressing emotions and are used to spread hate. Hatemoji(Kirk et al., 2022), is an abusive emoji dataset that has been created adversarially.

Various BERT(Devlin et al., 2019) based approaches have been taken to detect profane and hate speech content. HATE-BERT(Caselli et al., 2020), is a fine-tuned BERT model on abusive content from Reddit comments. Deep-BERT (Wadud et al., 2023), is a multilingual hate detection approach using transfer learning methods. Google has come up with their perspective 3 API (Lees et al., 2022) which uses a multilingual charformer model (Tay et al., 2021) to detect hateful content in a range of languages, domains and tasks.

Table 2: Data statistics

Dataset	Sentences	Sentences small-set
Train	34,080,768	16,384
Eval	172,544	2,234
Test	28,235	28,235

These models are generally prone to various noise attacks like adding small obfuscations or randomly changing a few characters, and its case (Hosseini et al., 2017). Significant research has been done to prevent adversarial attacks (Jain et al., 2018) on these models, and approaches like adding obfuscations and transformations to the text have shown improvements (Lees et al., 2022). Hybrid approaches of keeping humans in the loop along with the auto-moderation are also explored, which we too make use of (Link et al., 2016).

3 Proposed approach

We propose an end-to-end approach that uses a hybrid of deterministic and model-based approaches, and the data flow is shown in Figure 2.

3.1 Deterministic approaches

It is helpful to have blacklisted common profane words/phrases to do a list-based matching. We create n-gram phrases from the reviews and match them with our existing list of profane words, racial slurs, religious phrases, and political content. We maintain profane smileys, which indirectly express hate and sexual content on the platforms. We follow hybrid approaches of using the model and deterministic approaches for profane content.

We reject reviews that contain only punctuations, single letters, and random character sequences as poorly formatted content. Email addresses, phone numbers, and URLs are rejected using regex parser matching.

3.2 Domain adaptation

In the absence of abundant labelled data, we leverage the unlabelled monolingual review data by using them to pre-train the model. Pre-training helps the model understand better representation compared to the publicly available BERT. To address the domain gap, we train the BERT model from scratch as the vocabulary is updated to handle emojis and punctuations along with more relevant subwords in the e-commerce domain. We refer to this model as In-House-BERT.

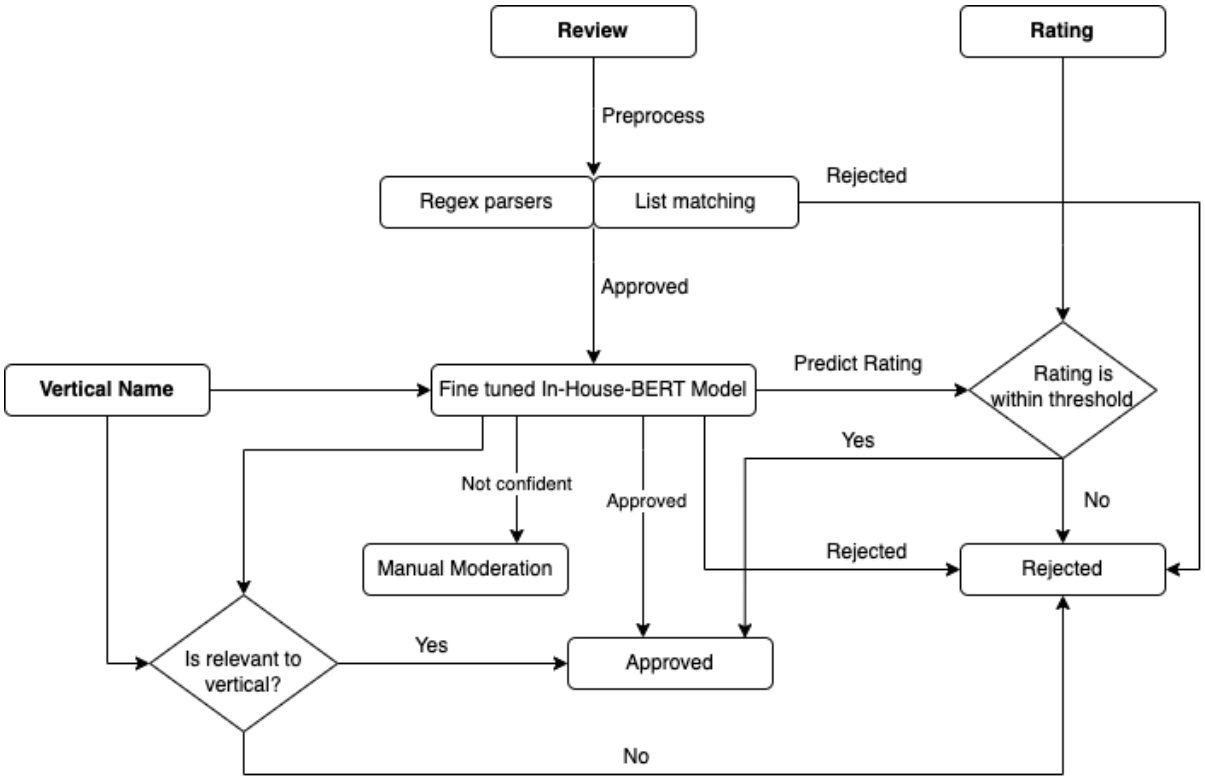

Figure 2: Dataflow of our Proposed approach

3.3 Product vertical embeddings

Product vertical information helps determine whether the given review is relevant to the product. The concatenation of review and vertical embeddings is passed to the dense layers of the classification head to detect irrelevant reviews.

3.4 Data Augmentation

Data augmentation is necessary for making the model robust to adversarial attacks. We augment only those rejection reasons which demand a high recall, i.e., profane content. We apply basic augmentations such as replacing the characters, dropping vowels, repeating characters, converting random characters to uppercase and adding profane smileys to the approved reviews. We substitute similar-looking characters such as 'i' with l,!, | to mimic the human perturbations.(Lees et al., 2022).

3.5 Rating prediction

Instead of having a sentiment detector separately, we reuse the rating data to predict the review's rating. We segment the 1 to 5-star rating into 3 buckets, considering it has positive, negative, and neutral. This is a separate classification head attached to the model, which will help determine the mismatch between the sentiment of the review and the user-given rating.

3.6 Model architecture

We develop a unified architecture, as shown in Figure 1, which can detect the various guidelines initially set to moderate the reviews. We initially have a BERT encoder(Devlin et al., 2019) which outputs a review and vertical embeddings, which are then connected to the 3 classification heads. All the heads contain dense layers followed by softmax, and they predict their respective classes. The irrelevancy detection head will get an extra vertical embedding as an input. We use the addition of 3 cross-entropy losses for back-propagation.

4 Experimental setup

4.1 Dataset

The user reviews contain text from different scripts and languages. We filter out the data to extract English text written in Roman using an in-house language classifier, which eliminates code-mixed data. We create a manually labelled corpus based on our moderation guidelines. We split the review data into train, validation, and test data, and the statistics are given in Table 2.

Table 3: F1 scores of experiments across various architectures and datasets

Models	Precision	Recall	F1 score
BERT-base	86.29	87.36	86.17
In-House-BERT	86.72	87.42	87.06
In-House-BERT-freeze	86.58	87.32	86.94
In-House-BERT-vertical	89.29	88.23	**88.45**
BERT-base-smallset	76.22	79.61	76.69
In-House-BERT-smallset	80.84	81.9	**80.36**

We create a smaller dataset of 16k training examples and name it as smallset. This dataset is created to evaluate the benefits of pre-training on monolingual data when there is a scarcity of labelled datasets. We use the same test set as before to evaluate the models.

4.2 Preprocessing

We start with the basic preprocessing of cleaning non-Roman characters and retaining emojis and punctuations. Emojis and punctuations play a vital role in understanding the review's sentiment. We normalize the numbers to a specific format n and nd for ordinal numbers to help models learn generic patterns. We did an empirical analysis and found that nearly 23% of the reviews contain spelling mistakes, formatting issues, and repeating characters. Even though variations of the data will make the model robust, noise-like repetitive characters/emojis/punctuations don't add much value to the model; hence we remove them.

4.3 Baseline and evaluation metrics

We use publicly available bert-base-cased[1] as our baseline model for evaluation with 2 classification heads, one for predicting the rating and another for the rejection reasons. This model takes a vertical name, and the text as the input and deterministic approaches are made part of the model. Training loss is the sum of cross-entropy across individual classification heads. We evaluate the models with weighted F1 scores across all the rejection reasons. The model aims to have high rejection recall while having high approval precision and decrease the volume for manual moderation, calculated by the percentage of data sent to the manual approval.

4.4 Pre-training on Monolingual data

We use product descriptions and reviews of monolingual data consisting of nearly 1B tokens to pre-train an In-House-BERT language model with 15% masking probability and Next Sentence Prediction task. We trained the model with a learning rate of 1e-5 for 2 epochs and observed the loss converge.

4.5 Fine-tuning on labelled data

We fine-tuned the In-House-BERT model by adding 2 classification heads and trained for 2 epochs with a batch size of 512 and a learning rate of 3e-5. We tried 2 different approaches, training the whole network and freezing the embeddings and initial 4 layers. As there was no significant degradation in accuracy by freezing the weights, we used this approach for further experiments as it helped in reducing training time.

As vertical information is not necessary for common rejection reasons, we added one more classification head for detecting irrelevant product reviews by concatenating reviews and vertical embeddings before passing them to the dense layers. Finally, we train a unified model with the learning from different approaches to fine-tune a unified In-House-BERT-vertical model with 3 classification heads and freeze the initial few layers.

We experiment with a smaller test set to evaluate the importance of pre-training BERT with limited labelled data. We use similar model configurations but train on nearly 16k training samples.

4.6 Thresholds for inference setup

For inference, we set thresholds for different rejection reasons. It is always better to have lesser thresholds for stricter rejection reasons, where compromising on recall is not an option. So we empirically set the thresholds on our evaluation set and then use the same thresholds across all the models. If the model is not confident in surpassing the threshold, it will be sent to manual moderation.

[1] https://huggingface.co/bert-base-cased

Table 4: F1 scores of experiments considering it as a binary classification problem along with Inference setup F1 scores using thresholds for better precision, and the percentage of data sent to manual moderation (lesser the better)

Models	F1 score (binary)	F1 score (with threshold)	Manual %ge
BERT-base	91.90	89.38	6.21
In-House-BERT	92.47	89.94	**5.67**
In-House-BERT-vertical	**93.02**	**90.32**	5.89
BERT-base-smallset	87.93	83.34	12.1
In-House-BERT-smallset	**89.37**	**85.49**	**9.02**

5 Results & Discussions

In Table 3, we can observe that the domain gap is being addressed using the pre-trained In-House-BERT model on smaller labelled datasets, observing an uplift of 4.78% in the F1 score. However, we don't see any significant difference with pre-training when abundant training data is available. Freezing the initial few layers of the BERT model doesn't degrade its accuracy numbers, and this can be used to reduce the training time by almost 40%. Product vertical embeddings play a better role in improving the rejection reason F1 score of individual reasons. Overall, our best model, In-House-BERT-vertical can beat the publicly available dataset by 2.57%.

5.1 Evaluating as a 2 class problem

We observe a lot of confusion for the model between rejection reasons, such as poorly formatted content being confused with irrelevant content. Further analysis revealed minor issues in the manual tagging of rejection reasons. We evaluate the model considering it as a binary classification problem with approved and reject labels. The results can be found in the first column of Table 4, where we see our best model has an F1 score of 93.02.

5.2 Impact of thresholding

It is always better to have a hybrid approach during inference because we can send the reviews for manual moderation when the model is not confident. Due to cost concerns and a longer turnaround time, it is desirable to minimise the volume of data sent to them. We set thresholds for different rejection reasons, and we observe that pre-training helps the model to be more confident at predicting the outputs reducing manual moderation load.

5.3 Deployment and Business Impact

The previously deployed system included rule-based methods and fasttext models but did not cover all the rejection reasons we introduced. Our current deployed system also significantly reduced the volume of manually moderated reviews from 23% to 5.89%. We have tested the system up to 10 queries per second with a P95 latency of 120 ms on 2 core CPUs with 2 GB RAM. We run multiple replicas to handle the volume of live review traffic.

We measure business impact based on cost reduction and revenue generation. Reducing the manual moderation percentage led to saving millions of dollars so far and we have also externalised moderation APIs to our group companies for providing additional revenues to the company.

6 Conclusion

Pre-training BERT on large monolingual data from a similar distribution as fine-tuning gives similar results if we have large enough training data. When labelled data is scarce, we observe the advantages of pre-training the BERT models with the monolingual corpora giving a 4.78% increase in F1. Freezing the embedding layer and a few of the initial layers of the In-House-BERT model helps reduce the training time while not compromising the model's performance. Decoupling some of the rejection reasons by adding extra embeddings boosts the F1 scores. Our hybrid approach achieves an F1 score of 88.45 and sends 5.89% for manual moderation.

Limitations and Future work

As our platform supports multilingual user-generated content, it becomes essential to support multilingual, multi-script, and code-mixed moderation. We are working on the explainability of the model to convey the reasons for rejection and make the model robust to various adversarial attacks and noisy label tagging. We plan to create more data for imbalanced datasets and focus on adding other rejection reasons like sarcasm and opinion spam detection.

Acknowledgements

We thank Sudhanshu Shekhar Singh, Shreyas Shetty and Raviraj Joshi for their helpful insights and suggestions on this work. We thank Sonal Bansal, Sakshi Bhatia, Subodh Kumar, Anupam Singh, Himanshu Agarwal, Prasad Pai, Vinay Lodha, and Flipkart UGC Ops team for their constant support. We thank Amey Patil for providing the pre-trained In-House-BERT models.

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. *CoRR*, abs/1706.00188.

Shiladitya Bhattacharya, Siddharth Singh, Ritesh Kumar, Akanksha Bansal, Akash Bhagat, Yogesh Dawer, Bornini Lahiri, and Atul Kr. Ojha. 2020. Developing a multilingual annotated corpus of misogyny and aggression. *CoRR*, abs/2003.07428.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A dataset of hindi-english code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, PEOPLES@NAACL-HTL 2018, New Orleans, Louisiana, USA, June 6, 2018*, pages 36–41. Association for Computational Linguistics.

Tommaso Caselli, Valerio Basile, Jelena Mitrovic, and Michael Granitzer. 2020. Hatebert: Retraining BERT for abusive language detection in english. *CoRR*, abs/2010.12472.

Kiril Danilchenko, Michael Segal, and Dan Vilenchik. 2022. Opinion spam detection: A new approach using machine learning and network-based algorithms. In *Proceedings of the Sixteenth International AAAI Conference on Web and Social Media, ICWSM 2022, Atlanta, Georgia, USA, June 6-9, 2022*, pages 125–134. AAAI Press.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Dumitru Erhan, Aaron C. Courville, Yoshua Bengio, and Pascal Vincent. 2010. Why does unsupervised pre-training help deep learning? In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 201–208. JMLR.org.

Aditya Gaydhani, Vikrant Doma, Shrikant Kendre, and Laxmi Bhagwat. 2018. Detecting hate speech and offensive language on twitter using machine learning: An n-gram and TFIDF based approach. *CoRR*, abs/1809.08651.

Anna Glazkova, Michael Kadantsev, and Maksim Glazkov. 2021. Fine-tuning of pre-trained transformers for hate, offensive, and profane content detection in english and marathi. *CoRR*, abs/2110.12687.

Cynthia Van Hee, Els Lefever, Ben Verhoeven, Julie Mennes, Bart Desmet, Guy De Pauw, Walter Daelemans, and Véronique Hoste. 2015. Detection and fine-grained classification of cyberbullying events. In *Recent Advances in Natural Language Processing, RANLP 2015, 7-9 September, 2015, Hissar, Bulgaria*, pages 672–680. RANLP 2015 Organising Committee / ACL.

Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. 2017. Deceiving google's perspective API built for detecting toxic comments. *CoRR*, abs/1702.08138.

Òscar Garibo i Orts. 2019. Multilingual detection of hate speech against immigrants and women in twitter at semeval-2019 task 5: Frequency analysis interpolation for hate in speech detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, June 6-7, 2019*, pages 460–463. Association for Computational Linguistics.

Edwin Jain, Stephan Brown, Jeffery Chen, Erin Neaton, Mohammad Baidas, Ziqian Dong, Huanying Gu, and Nabi Sertac Artan. 2018. Adversarial text generation for google's perspective api. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1136–1141.

Nitin Jindal and Bing Liu. 2007. Review spam detection. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 1189–1190. ACM.

Hannah Kirk, Bertie Vidgen, Paul Röttger, Tristan Thrush, and Scott A. Hale. 2022. Hatemoji: A test suite and adversarially-generated dataset for benchmarking and detecting emoji-based hate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1352–1368. Association for Computational Linguistics.

Damir Korencic, Ipek Baris, Eugenia Fernandez, Katarina Leuschel, and Eva Salido. 2021. To block or not to block: Experiments with machine learning for news comment moderation. In *Proceedings of the*

EACL Hackashop on News Media Content Analysis and Automated Report Generation, EACL 2021, Online, April 19, 2021, pages 127–133. Association for Computational Linguistics.

Shashank Kumar. 2017. Research on product review analysis and spam review detection.

Jaejun Lee, Raphael Tang, and Jimmy Lin. 2019. What would elsa do? freezing layers during transformer fine-tuning. *CoRR*, abs/1911.03090.

Alyssa Lees, Vinh Q. Tran, Yi Tay, Jeffrey Sorensen, Jai Prakash Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective API: efficient multilingual character-level transformers. *CoRR*, abs/2202.11176.

Daniel Link, Bernd Hellingrath, and Jie Ling. 2016. A human-is-the-loop approach for semi-automated content moderation. In *13th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Rio de Janeiro, Brasil, May 22-25, 2016*. ISCRAM Association.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017. Deep learning for user comment moderation. In *Proceedings of the First Workshop on Abusive Language Online, ALW@ACL 2017, Vancouver, BC, Canada, August 4, 2017*, pages 25–35. Association for Computational Linguistics.

Jing Qian, Anna Bethke, Yinyin Liu, Elizabeth M. Belding, and William Yang Wang. 2019. A benchmark dataset for learning to intervene in online hate speech. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4754–4763. Association for Computational Linguistics.

Ajay Rastogi and Monica Mehrotra. 2017. Opinion spam detection in online reviews. *J. Inf. Knowl. Manag.*, 16(4):1750036:1–1750036:38.

Marcos Rodrigues Saude, Marcelo de Medeiros Soares, Henrique Gomes Basoni, Patrick Marques Ciarelli, and Elias Oliveira. 2014. A strategy for automatic moderation of a large data set of users comments. In *XL Latin American Computing Conference, CLEI 2014, Montevideo, Uruguay, September 15-19, 2014*, pages 1–7. IEEE.

Yi Tay, Vinh Q. Tran, Sebastian Ruder, Jai Prakash Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. 2021. Charformer: Fast character transformers via gradient-based subword tokenization. *CoRR*, abs/2106.12672.

Turki Turki and Sanjiban Sekhar Roy. 2022. Novel hate speech detection using word cloud visualization and ensemble learning coupled with count vectorizer. *Applied Sciences*, 12(13).

Shunya Ueta, Suganprabu Nagaraja, and Mizuki Sango. 2020. Auto content moderation in C2C e-commerce. In *2020 USENIX Conference on Operational Machine Learning, OpML 2020, July 28 - August 7, 2020*. USENIX Association.

Andreas A. Veglis. 2014. Moderation techniques for social media content. In *Social Computing and Social Media - 6th International Conference, SCSM 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014. Proceedings*, volume 8531 of *Lecture Notes in Computer Science*, pages 137–148. Springer.

Adriano Veloso, Wagner Meira Jr., Tiago Alves Macambira, Dorgival O. Guedes, and Hélio Marcos Paz de Almeida. 2007. Automatic moderation of comments in a large on-line journalistic environment. In *Proceedings of the First International Conference on Weblogs and Social Media, ICWSM 2007, Boulder, Colorado, USA, March 26-28, 2007*.

Md. Anwar Hussen Wadud, Muhammad F. Mridha, Jungpil Shin, Kamruddin Nur, and Aloke Kumar Saha. 2023. Deep-bert: Transfer learning for classifying multilingual offensive texts on social media. *Comput. Syst. Sci. Eng.*, 44(2):1775–1791.

A Obfuscation techniques

Augmentation techniques are used to create more data for profane and hate speech content by adding multiple obfuscation techniques described in Table 5. Data augmentation certainly gives a boost to profane content F1 scores by 18%.

Table 5: Various data augmentation techniques that we used on an example profane word "bullshit"

Technique	Augmented phrases
Replace characters with *	bu**sh*t, bullsh*t
Drop vowels randomly	bllsht, bullsht
Repeating characters	bullllshiiiitt
Random case changing	buLLshIT
Add random spaces	bu llshi t, bull shit
Replace similar-looking characters	bull 5h!t, bu!! śhît