Transfer Learning and Prediction Consistency for Detecting Offensive Spans of Text

Amir Pouran Ben Veyseh[1], **Ning Xu**[2],
Quan Hung Tran[2], **Varun Manjunatha**[2], **Franck Dernoncourt**[2], **Thien Huu Nguyen**[1]
[1]Department of Computer and Information Science,
University of Oregon, Eugene, OR, USA
[2]Adobe Research, San Jose, CA, USA
{apouranb,thien}@cs.uoregon.edu
{nxu,qtran,vmanjuna,franck.dernoncourt}@adobe.com

Abstract

Toxic span detection is the task of recognizing offensive spans in a text snippet. Although there has been prior work on classifying text snippets as offensive or not, the task of recognizing spans responsible for the toxicity of a text is not explored yet. In this work, we introduce a novel multi-task framework for toxic span detection in which the model seeks to simultaneously predict offensive words and opinion phrases to leverage their inter-dependencies and improve the performance. Moreover, we introduce a novel regularization mechanism to encourage the consistency of the model predictions across similar inputs for toxic span detection. Our extensive experiments demonstrate the effectiveness of the proposed model compared to strong baselines.

1 Introduction

With the proliferation of social networks, the amount of textual data posted online is also ever-increasing. This growth comes with some challenges too. One of the issues associated with social networks is the level of toxicity expressed in posts or comments shared online. The toxic/offensive languages in social networks can be realized in different forms such as insults, mockeries, threats, discrimination, or swearing. Due to their detrimental effect on users of social networks, it is desirable to identify and remove offensive text from these networks.

Since this is an important requirement, the task of offensive language detection has been extensively studied in NLP community (Schmidt and Wiegand, 2017; Feng et al., 2018; Borkan et al., 2019; Sivanaiah et al., 2020; Yasaswini et al., 2021). However, most of the existing works are limited to classifying a text snippet as offensive or not. In other words, these models fail to provide further information about what specific phrases in the text snippet contribute the most to the offensive tone of the text. This information is necessary for the moderators to decide further actions for the posts/comments flagged as offensive, especially if the text snippet is long. As such, in this work, we fill this gap by proposing a novel model for the task of offensive span detection (OSD). As an example, in the given text "*This livestreamer clearly has no brain; he is such a tool!*", the phrase "*has no brain*" and the slang word "*tool*" are two offensive spans responsible for the toxicity of the text. One of the barriers for this task is data scarcity. To address this limitation, we propose a novel model trained in multi-task setting in which the model is trained on two tasks: (1) Offensive phrase detection whose goal is to detect word(s) contributing to the toxicity of the text, (2) Opinion word extraction which is supposed to assist the main model to pinpoint word(s) conveying subjectivity. Note that the second task could help the model restrict its prediction to more likely words. As the available resources for offensive span detection do not provide any annotation for opinion words in the text, in this work, we propose to employ transfer learning to fulfill the training on the second task (i.e., opinion word extraction). In particular, a separate model is pre-trained on sentiment polarity prediction on a sentiment analysis corpus. Afterward, the pre-trained model is exerted to provide supervision for the task of opinion word extraction. In addition to the proposed multi-task setting, we also introduce a novel regularization loss in which the model is encouraged to make consistent predictions on similar inputs. Concretely, in this work, we propose to compute the similarity between samples in a mini-batch with respect to two criteria: (i) word representations (ii) prediction of offensive words. During training, the samples that have the highest similarity are encouraged to have less discrepancy with each other. In order to fulfill this goal, for the first time, we propose to employ Optimal Transport

Findings of the Association for Computational Linguistics: ACL 2022, pages 1630 - 1637

to compute the consistency loss between samples. We evaluate the proposed model on a recently released dataset for offensive span detection. Our extensive experiments show the effectiveness of the proposed model by outperforming the strong baselines.

2 Model

Formal Task Description: The input to the model is the document $D = [w_1, w_2, \ldots, w_n]$ consisting of n words. The label provided for the document is also the sequence $Y = [y_1, y_2, \ldots, y_n]$ in which y_i is the label for the word w_i in BIO format. This problem is modeled as a sequence labeling task in which the model predicts the label of every word w_i in the document D. Our proposed method is based on multi-task training with opinion word prediction as the auxiliary task. We also propose a novel regularization using Optimal Transport. The rest of this section provides details of our approach.

2.1 Main Task

For the main task of offensive span detection (OSD), we employ the BERT$_{base}$ model with fixed parameters to encode the input text. Formally, the input to the BERT model is $[CLS]w_1w_2 \ldots w_n[SEP]$ and the representation of the token at the final layer of the BERT model are used to represent them, i.e., $X = [x_1, x_2, \ldots, x_n]$. Since the parameters of the BERT model are fixed, to update the representations of the tokens for the offensive span detection task, we feed the representations X to a Bi-directional Long Short-Term Memory (BiLSTM) network. The hidden states of the BiLSTM network is used as the final representations of the words, i.e., $H = [h_1, h_2, \ldots, h_n]$. Finally, a two-layer feed-forward network is employed to obtain the label distribution $P(\cdot|D, w_i)$ for word w_i: $P(\cdot|D, w_i) = \text{softmax}(W_1 * (W_2 * h_i + b_1) + b_2)$, where W_1 and W_2 are the weight matrices, b_1 and b_2 are biases, softmax is the softmax function, and the $P(\cdot|D, w_i)$ represent the probability distribution over different labels predicted by the feed-forward layer for the word w_i. To train this model, we use cross-entropy loss in word-level (i.e., negative log-likelihood). More specifically, the following loss function is used: $\mathcal{L}_{main} = -\sum_i^n \log(P(y_i|D, w_i))$, where y_i is the gold label for the word w_i in the document D in training data.

2.2 Auxiliary Task

One of the limitations of the existing training data for OSD is their small size which could hurt the generalization ability of the model. To alleviate this issue, we propose to train the model in a multi-task setting, thus benefiting from the interaction between the main and the auxiliary task. Specifically, we choose opinion work extraction (OWE) as the auxiliary task. In this task, the goal is to find the words conveying sentiment in text. Note that opinion words are the super-set of the toxic words, as such training on OWE could help the model to restrict its predictions to more likely words. Unfortunately, the existing OSD datasets do not annotate the opinion words. Therefore, to train the model on OWE, we resort to transfer-learning, in which a pre-trained model on another related task, i.e., Sentiment Analysis (SA), is employed to guide the OSD model on the auxiliary task OWE.

Specifically, for the pre-training of the SA model, we employ the available sentiment analysis dataset, i.e., $\mathcal{D}_{SA}$. In this dataset, every sentence $S' \in \mathcal{D}_{SA}$ is labeled as "*Positive*", "*Neutral*" or "*Negative*". To train the model SA, the sentence S', represented by the GloVe embedding of its words, is encoded by a BiLSTM network, i.e., $H' = [h'_1, h'_2, \ldots, h'_m]$. Finally, a feed-forward network consumes the max-pooled representation of the sentence S' to produce the label probability distribution, i.e., $P'(\cdot|S') = FF(MAX\_POOL(H'))$. To train the model, the negative log-likelihood is employed: $\mathcal{L}_{pre} = -\log(P(l|S'))$, where l is the label of the sentence S'.

In order to employ the per-trained SA model to guide the OSD model for OWE, we posit that if the OSD model masks the opinion words of the input document D then the pre-trained model SA will predict *Neutral* label for the masked document. Note that without masking, the sentiment of the document D is always negative. To fulfill this idea, in our model, we first feed the representation of the document D, i.e., the vectors H obtained form BiLSTM of OSD model, to a feed-forward network to obtain the scores $A = [a_1, a_2, \ldots, a_n]$, where $a_i = \sigma(FF(h_i))$ and σ is the sigmoid activation function. The scores a_i represent the extent to which the OSD model predicts w_i as opinion word. Next, to mask out the opinion words, the weighted vectors X' is computed: $X' = [x'_1, x'_2, \ldots, x'_n]$, where $x'_i = a_i * x_i$ and x_i is the GloVe embedding of the word $w_i \in D$. The masked document

representation X' is fed into the pre-trained model SA to obtain the label distribution $P'(\cdot|D)$. Note that, during training of the main model, the parameters of the pre-trained SA model are fixed. To train the main model, we use the following loss function: $\mathcal{L}_{aux} = -\log(P'(l_n|D))$, where l_n is the *Neutral* label. As the training could collapse by predicting all-zero vector for A, we use the following regularization for the auxiliary task: $\mathcal{L}_{reg} = |n - \text{SUM}(A)|$, where n is the length of the document and $SUM(X)$ is the sum of all elements of the vector X.

2.3 Prediction Consistency

In order to address the data scarcity for OSD, we also propose a novel regularization in which the model is encouraged to make consistent predictions for similar input documents. Hence, the model behavior on one sample can guide it on the other samples too. To this end, we propose to compute the consistency between model's predictions on two documents D_i and D_j by the cost of converting (D_i, Y_i') to (D_j, Y_j') where Y_i' and Y_j' are predictions of the model for documents D_i and D_j, respectively. This problem can be efficiently solved by Optimal Transport (OT). OT is a method to compute the lowest cost of converting a probability distribution to another one. Formally, given the probability distributions $p(x)$ and $q(y)$ over the domains $\mathcal{X}$ and $\mathcal{Y}$, and the cost function $C(x, y) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ for mapping $\mathcal{X}$ to $\mathcal{Y}$, OT finds the optimal joint distribution $\pi^*(x, y)$ (over $\mathcal{X} \times \mathcal{Y}$) with marginals $p(x)$ and $q(y)$, i.e., the cheapest transportation from $p(x)$ to $q(y)$, by solving the following problem:

$$\pi^*(x,y) = \min_{\pi \in \Pi(x,y)} \int_{\mathcal{Y}} \int_{\mathcal{X}} \pi(x,y) C(x,y) dx dy \quad (1)$$
$$\textbf{s.t. } x \sim p(x) \text{ and } y \sim q(y),$$

where $\Pi(x, y)$ is the set of all joint distributions with marginals $p(x)$ and $q(y)$. Note that if the distributions $p(x)$ and $q(y)$ are discrete, the integrals in Equation 1 are replaced with a sum and the joint distribution $\pi^*(x, y)$ is represented by a matrix whose entry (x, y) represents the probability of transforming the data point $x \in \mathcal{X}$ to $y \in \mathcal{Y}$ to convert the distribution $p(x)$ to $q(y)$. By solving the problem in Equation 1[1], the cost of transforming the discrete distribution $p(x)$ to $q(y)$ (i.e., Wasserstein distance $Dist_W$) is defined as: $Dist_W = \Sigma_{x \in \mathcal{X}} \Sigma_{y \in \mathcal{Y}} \pi^*(x, y) C(x, y)$.

In our model, we use the Wasserstein distance $Dist_W$ between two documents to compute their consistency. In particular, for every pair of (D_k, D_l) where D_k and D_l are two documents in the same mini-batch, the domain $\mathcal{X}$ is defined over the word representations of the document D_k, i.e., H_k, and the domain $\mathcal{Y}$ is defined over the word representations of the document D_l, i.e., H_l. Moreover, in order to define the distributions $p(x)$ and $q(y)$, we take the probability of the label O for each word of the document D_k and D_l predicted by the main task model and feed that into a softmax function.

Finally, to define the cost function $C(x_i, y_j)$, we use the Euclidean distance between the two vector representation h_i and h_j for the word w_i of D_k and the word w_j of D_l: $C(x_i, y_j) = \|h_i - h_j\|$. Using these definitions, we can use OT to compute the Wasserstein distance $Dist_W^{k,l}$ between the document D_k and D_l in the same mini-batch. Finally, we select the document $D_{k'}$ as the most similar document to D_k where: $k' = \text{argmin}_l Dist_W^{k,l}$

Hence, we define the consistency loss for document D_k as its Wasserstein distance to the similar document $D_{k'}$: $\mathcal{L}_{cons} = Dist_W^{k,k'}$.

Finally, we use the following loss function with trade-off parameters α, β, and γ to train the entire model: $\mathcal{L} = \mathcal{L}_{main} + \alpha * \mathcal{L}_{aux} + \beta * \mathcal{L}_{reg} + \gamma * \mathcal{L}_{cons}$

3 Experiments

In order to evaluate the effectiveness of the proposed model, called TPOSD (Transfer learning and Prediction consistency for Offensive Span Detection), in our experiments, we use the dataset of SemEval 2021 Task 5 (John Pavlopoulos and Laugier, 2021). We use the official splits with 7939/690/2000 documents in train/development/test sets. Also, to pre-train the SA model for the sentiment analysis task to be used for auxiliary training, we employ the Amazon-2 dataset Zhang et al. (2015). In our model we use the (fixed) BERT$_{base}$ to encode data; 250 dimensions for the hidden states of LSTM and 2 layers for feed-forward neural networks with 250 hidden dimensions. The trade-off parameters α, β and γ are set to 0.1, 0.1, and 0.05, respectively. The learning rate is set to 0.3 for the Adam optimizer and the batch size of 64 is employed during training.

we compare the performance of TPOSD with

[1]It is worth mentioning that this problem is intractable so we solve its entropy-based approximation using the Sinkhorn algorithm (Peyre and Cuturi, 2019).

Model	Precision	Recall	F1
BiLSTM-CRF	55.31	62.57	58.72
BERT-CRF	61.45	65.03	63.19
DUAL-MRC	60.13	69.02	64.27
SANER	62.96	71.07	66.77
IITK	-	-	68.20
TPOSD (Ours)	**67.78**	**71.92**	**69.79**

Table 1: Performance of the models on the test set of the SemEval 2021 Task 5 dataset.

Model	Precision	Recall	F1
TPOSD	**66.88**	**70.85**	**68.81**
OWE^-	65.60	62.72	64.13
$AuxReg^-$	65.11	66.99	66.04
$Cons^-$	67.19	65.47	66.32
$Cons^{-sem}$	65.44	66.83	66.13
$Cons^{-pred}$	65.24	70.59	67.81

Table 2: Ablation study on the development set of the SemEval 2021 Task 5 dataset.

the following baselines: (1) **BiLSTM+CRF**: The GloVe embedded document is encoded by BiLSTM and the labels are predicted by a CRF layer; (2) **BERT+CRF**: BERT_{base} parameters are fine-tuned on OSD task and the task-specific head, i.e., CRF, is employed for label prediction; (3) **IITK** (Bansal et al., 2021): This baseline is the existing SOTA model on SemEval 2021 Task 5 dataset; (4) **SANER** (Nie et al., 2020): This baseline is the SOTA model for sequence labeling on user-generated text; (5) **DUAL-MRC** (Mao et al., 2021): This is the SOTA model for opinion and aspect term extraction. Note that since there are not target annotations in SemEval dataset, we skip the aspect term extraction task in the training of this baseline.

Results: Table 1 shows the performance of the models on the test set. There are several observations from this table. First, the BiLSTM-CRF model significantly underperforms the other baselines that employ BERT embedding. It clearly shows that the background knowledge encoded in the BERT model is necessary for the task of offensive span detection. Second, both DUAL-MRC and SANER baseline outperform the BERT-CRF model. This higher performance could be attributed to their capability to augment the representation of the words obtained from the BERT model. Third, among all baselines, our proposed model achieves the highest performance. Our hypothesis for the achieved improvement is that the proposed model is able to restrict its predictions to the more probable candidate spans, i.e., opinion words, due to the training of the auxiliary task. Moreover, it performs more consistently across different documents thanks to the consistency regularization employed during the training of the model.

Analysis: To study the contribution of the proposed techniques, we conduct an ablation study on the development set of the SemEval 2021 Task 5 dataset. Specifically, we ablate the auxiliary task (**OWE**$^-$), the regularization in the auxiliary task (**AuxReg**$^-$), i.e., $\mathcal{L}_{reg}$, the consistency loss (**Cons**$^-$), i.e., $\mathcal{L}_{cons}$. Also, we study the performance of the model when the Wasserstein distance is computed regardless of the document representations (**Cons**$^{-sem}$) or model predictions (**Cons**$^{-pred}$). The results are shown in Table 2. This table shows that all components are necessary, as removing each will hurt the performance. Specifically, the auxiliary task has the largest effect on the final performance, indicating the importance of the proposed method. For the case study analysis, see appendices.

In the proposed approach, to simultaneously train the model on OSD and OWE, as the existing training data for OSD does not provide gold labels for OWE, we resort to transfer-learning, in which a pre-trained sentiment-analysis model is employed to supervise the main model on OWE task. However, one natural question is that why transfer-learning is the optimal approach to train the model on OWE? To answer this question, in this section, we propose a baseline, in which a model pre-trained on OWE is employed to automatically annotate the existing OSD training data with opinion words for OWE task. In particular, we first train a sequence-tagger consisting of a BiLSTM encoder followed by a feed-forward layer on the available OWE dataset. Specifically, in our experiments, we use the combinations of the four benchmark datasets presented by Fan et al. (2019) as the training data to pre-train the OWE model. Note that the original datasets by Fan et al. (2019) provide opinion words with respect to a given target mention. However, in the pre-training of the OWE model, we aim to train the model to detect all opinion words in the input text. As such, in our experiments, we combine opinion words of all samples of the same sentence in the dataset. Finally, the pre-trained OWE model is employed to annotate the opinion words in the OSD dataset, i.e., SemEval 2021 Task 5 (John Pavlopoulos and Laugier, 2021). The au-

Model	Precision	Recall	F1
Pre-Train OWE	64.11	68.83	66.39
TPOSD (Ours)	**67.78**	**71.92**	**69.79**

Table 3: Performance of the models on the test set of the SemEval 2021 Task 5 dataset.

tomatically annotated OSD dataset with opinion words is next employed to jointly train the main model on OSD and OWE. Concretely, the representations of the words obtained from the main model BiLSTM, i.e., $H = [h_1, h_2, \ldots, h_n]$, are fed into two different feed-forward layers FF_{OSD} and FF_{OWE} to obtain the label probability distribution $P_{OSD}(\cdot|D, w_i)$ and $P_{OWE}(\cdot|D, w_i)$, respectively: $P_{OSD}(\cdot|D, w_i) = \text{softmax}(FF_{OSD}(h_i))$, and $P_{OWE}(\cdot|D, w_i) = \text{softmax}(FF_{OWE}(h_i))$.

Finally, the following loss functions are employed for each task: $\mathcal{L}_{OSD} = -\sum_i^n \log(P_{OSD}(y_i^{OSD}|D, w_i))$ and $\mathcal{L}_{OWE} = -\sum_i^n \log(P_{OWE}(y_i^{OWE}|D, w_i))$, where y_i^{OSD} and y_j^{OWE} are the gold labels for offensive span detection (OSD), provided in the SemEval dataset, and opinion word extraction (OWE) tasks, provided by the pre-trained OWE model, for i-th word. The overall loss to train the model jointly on both tasks is then defined by: $\mathcal{L}_{total} = \mathcal{L}_{OSD} + \alpha * \mathcal{L}_{OWE}$.

We call this baseline **Pre-Train OWE** and its performance on the test set of the SemEval dataset is reported in Table 3. This table shows that this baseline under-performs our transfer-learning approach. Our hypothesis for this inferior performance is that compared to our transfer-learning approach that utilizes soft filtering of the input text to identify the opinion words, the pre-trained model employs the discrete labels for OWE generated by the pre-trained model. As the pre-trained OWE model could be erroneous, thus the errors can more easily deflect the training of the main model. Unlike this baseline, in our proposed model, the opinion words are denoted by the scores A discussed in section 2.2. As such, the soft opinion word extraction mechanism in our proposed model has more potential to overcome errors in the OWE task.

4 Related Work

Prior works related to this task can be categorized into two groups: (i) Toxicity Detection: These works aim to classify a piece of text as toxic or non-toxic (Wulczyn et al., 2017; Borkan et al., 2019; Schmidt and Wiegand, 2017; Pavlopoulos et al., 2017a,b, 2019; Zampieri et al., 2019). The main limitation of these works is that they cannot recognize the spans in the text that are responsible for the toxicity of the text. (ii) Opinion Word Extraction: In this group, models perform a sequence labeling task to identify the spans in the text that convey the sentiment (Liu et al., 2015; Xu et al., 2018; Yin et al., 2016; Wang et al., 2016, 2017; Li and Lam, 2017; Mao et al., 2021). The major limitation of all these models is that they require the existence of the target opinion (i.e., the word or phrase that the text has a sentiment polarity toward it).

5 Conclusion

In this work, we proposed a novel model for offensive span detection. To train the model, in a novel framework, we propose to exploit the interaction with the related task of opinion word extraction. Specifically, in a multi-task learning setting, we train the model for offensive and opinion word extraction. Also, we introduce a novel regularization loss based on optimal transport which encourages the consistency of the model prediction on similar documents. Our experiments on the available benchmark dataset show the effectiveness of the proposed model and outperform strong baselines.

Acknowledgments

This research has been supported by the Army Research Office (ARO) grant W911NF-21-1-0112 and the NSF grant CNS-1747798 to the IUCRC Center for Big Learning. This research is also based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA Contract No. 2019-19051600006 under the Better Extraction from Text Towards Enhanced Retrieval (BETTER) Program. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ARO, ODNI, IARPA, the Department of Defense, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein. This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

References

Archit Bansal, Abhay Kaushik, and Ashutosh Modi. 2021. Iitk@ detox at semeval-2021 task 5: Semi-supervised learning and dice loss for toxic spans detection. *arXiv preprint arXiv:2104.01566*.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pages 491–500.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. 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 2509–2518, Minneapolis, Minnesota. Association for Computational Linguistics.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. *arXiv preprint arXiv:1804.07781*.

Jeffrey Sorensen John Pavlopoulos, Ion Androutsopoulos and Léo Laugier. 2021. Toxic span detection at semeval 2021. In *SemEval 2021 (To Appear)*.

Xin Li and Wai Lam. 2017. Deep multi-task learning for aspect term extraction with memory interaction. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2886–2892.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1433–1443.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. *arXiv preprint arXiv:2101.00816*.

Yuyang Nie, Yuanhe Tian, Xiang Wan, Yan Song, and Bo Dai. 2020. Named entity recognition for social media texts with semantic augmentation. In *EMNLP*.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017a. Deep learning for user comment moderation. *arXiv preprint arXiv:1705.09993*.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017b. 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, Nithum Thain, Lucas Dixon, and Ion Androutsopoulos. 2019. Convai at semeval-2019 task 6: Offensive language identification and categorization with perspective and bert. In *Proceedings of the 13th international Workshop on Semantic Evaluation*, pages 571–576.

Gabriel Peyre and Marco Cuturi. 2019. Computational optimal transport: With applications to data science. In *Foundations and Trends in Machine Learning*.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10.

Rajalakshmi Sivanaiah, Angel Suseelan, S Milton Rajendram, and Mirnalinee Tt. 2020. Techssn at semeval-2020 task 12: Offensive language detection using bert embeddings. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2190–2196.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. *arXiv preprint arXiv:1603.06679*.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer tensor network for co-extraction of aspect and opinion terms. In *Proceedings of AAAI*, pages 3316–3322.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*.

Konthala Yasaswini, Karthik Puranik, Adeep Hande, Ruba Priyadharshini, Sajeetha Thavareesan, and Bharathi Raja Chakravarthi. 2021. Iiitt@ dravidianlangtech-eacl2021: Transfer learning for offensive language detection in dravidian languages. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 187–194.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. *arXiv preprint arXiv:1605.07843*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). *arXiv preprint arXiv:1903.08983*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

A Case Study

To qualitatively study the improvement achieved by the proposed model, in Table 4, we present some cases that the proposed model could successfully identify the offensive spans while the other baselines fail. Specifically, cases 1 and 2 show that the baseline *BERT-CRF* incorrectly predict non-opinion words/phrase "*gross reliance*" and "*joke*" as the offensive spans. On the other hand, TPOSD successfully predicts the offensive spans. This improvement could be attributed to the training of the main model on opinion word extraction which could restrict model decisions to more likely words. Moreover, in case 3, the baseline *BERT-CRF* incorrectly predicts the word "*strange*" as the offensive word. However, the proposed TPOSD model successfully identifies the word "*idiot*" as the only offensive word in the text. Among other reasons, the better performance of the proposed model for this case could be due to the regularization enforced during training which helps the model learns from other samples that "*strange*" is less likely to be used as the offensive word.

ID	Document	BERT-CRF	TPOSD	Gold
1	Sorry. Damn spell checker and my gross reliance on it! My humblest apologies :-)	gross reliance	Damn	Damn
2	Yeah, what a joke. They can't confirm the gunshot wounds were inflicted by the police? Ridiculous.	joke	Ridiculous	Ridiculous
3	Hard to believe this strange comment! He is such an idiot.	strange, idiot	idiot	idiot

Table 4: Case study in SemEval 2021 Task 5 dataset. Predicted offensive words by the *BERT-CRF* and the proposed model, *TPOSD*, along with gold labels are provided.