HITSZ-HLT at SemEval-2021 Task 5: Ensemble Sequence Labeling and Span Boundary Detection for Toxic Span Detection

Qinglin Zhu[1,†], **Zijie Lin**[1,†], **Yice Zhang**[1,†], **Jingyi Sun**[1], **Xiang Li**[1]
Qihui Lin[1], **Yixue Dang**[2], **Ruifeng Xu**[1,‡]
[1]Joint Lab of HITSZ-CMS, Harbin Institute of Technology(Shenzhen), China
[2]China Merchants Securities Co., Ltd
{zhuqinglin,20S051050,xiangli,1171000607}@stu.hit.edu.cn
{lzjjeffery, zhangyc_hit}@163.com, dangyixue@cmschina.com.cn
xuruifeng@hit.edu.cn

Abstract

This paper presents the winning system that participated in SemEval-2021 Task 5: Toxic Spans Detection. This task aims to locate those spans that attribute to the text's toxicity within a text, which is crucial for semi-automated moderation in online discussions. We formalize this task as the Sequence Labeling (SL) problem and the Span Boundary Detection (SBD) problem separately and employ three state-of-the-art models. Next, we integrate predictions of these models to produce a more credible and complement result. Our system achieves a char-level score of 70.83%, ranking 1/91. In addition, we also explore the lexicon-based method, which is strongly interpretable and flexible in practice.

1 Introduction

41% of American adults in 2020 are reported experiencing some form of harassment[1]. Increasing incidents of online harassment and cyber violence have spurred researchers to investigate the problem of identifying and filtering offensive speech on the Internet. Most previously published insult detection tasks (Davidson et al., 2017; Xu et al., 2012) and methods (Aroyehun and Gelbukh, 2018; Modha et al., 2018) classify an entire comment (or document) to discern whether the comment is offensive or not, but cannot identify specific pieces of the toxic comment. Unlike previous studies, SemEval-2021 Task5: Toxic Span Detection(Pavlopoulos et al., 2021) requires the identification of the specific toxic spans, which is more innovative and challenging, and a key step towards a successful semi-automatic review of comments.

More formally, toxic span detection is an extraction task, which is usually formalized as a Sequential Labeling (SL) problem, as shown in Figure 1(a), locating those spans by BIO tags. However, SL methods suffer from a huge search space due to the compositionality of labels (the power set of all sentence words), which has been proven in (Lee et al., 2016; Hu et al., 2019a). Therefore, in addition to SL formalization, we also formalize the task as a Span Boundary Detection (SBD) problem, as shown in Figure 1(b), locating those spans by start and end positions. Notice that, when there are multiple spans in a sentence, the matching of start and end positions may be ambiguous during decoding. This shows that theoretically, the SBD formalization is not consistently superior to the SL formalization. Hence, we choose to combine predictions of these two kinds of formalization to produce a more credible and complement result. Our system achieves a char-level score of 70.83%, ranking 1/91.

Besides, we also explore the lexicon-based methods, which usually have high precision but rather low recall, and are strongly interpretable and flexible in practice. First, we mine a toxic lexicon from the training set by a simple statistical strategy. Next, WordNet (Fellbaum, 2010) and GloVe (Pennington et al., 2014) are utilized to extend this lexicon further. With a toxic lexicon, we extract toxic spans through word-level matching.

2 Related Work

In recent years, cyber violence has become a widespread societal concern, and how to identify and filter hate speech has become an important topic in machine learning. TRAC proposes an aggression recognition task (Kumar et al., 2018) that provides a dataset of 15,000 annotated Facebook posts and comments in English and Hindi for

[†]Authors equally contributed to this work.
[‡]Corresponding Author: xuruifeng@hit.edu.cn
[1]https://www.pewresearch.org/internet/2021/01/13/the-state-of-online-harassment/

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 521–526
Bangkok, Thailand (online), August 5–6, 2021.

(a) (b)

Figure 1: Comparison of SL and SBD, (a) denotes SL, (b) denotes SBD.

training and validation. The task aims to classify comments into three categories: non-aggressive, covertly aggressive, and overly aggressive. The Toxic Comment Classification Challenge 5 [2] is an open competition in Kaggle that provides participants with comments from Wikipedia and defines six toxic categories: toxic, severe toxic, obscene, threat, insult, identity hate. In SemEval 2019 task 6 (Zampieri et al., 2019), in addition to whether the comment is offensive, the type of the attack and the target of the attack are also included. Based on this, Semeval 2020 task 12 (Zampieri et al., 2020) further extends the dataset to 5 languages: Arabic, Danish, English, Greek, and Turkish.

3 Methods

In the section, we describe how toxic span detection is formalized and corresponding solutions in detail.

3.1 Sequence Labeling

The `BIO` tag scheme is utilized to locating toxic spans, where `B` (Begin) corresponds to the first token in a toxic span, `I` (Inside) corresponds to the inside and end tokens in a toxic span, and `O` corresponds to those no-toxic tokens. Following most existing work (Lample et al., 2016; Ma and Hovy, 2016), we leverage Conditional Random Fields (CRF) (Lafferty et al., 2001) for learning and inference.

In addition to token-level classification, CRF models the dependencies between tags in a tag sequence by the transition matrix $A \in \mathbb{R}^{K\times K}$, where K is the size of the tag space, i.e. $K = 3$. For the contextual representation $\mathbf{x} \in \mathbb{R}^{n\times h}$, the score of a tag sequence $\mathbf{y} \in \mathbb{R}^n$ in CRF is defined as:

$$S(\mathbf{x},\mathbf{y}) = h^1(y_1;\mathbf{x}) + \sum_{k=1}^{n-1}\left(h^{k+1}(y_{k+1};\mathbf{x}) + A_{y_k,y_{k+1}}\right). \quad (1)$$

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

where $h^k(y_k;\mathbf{x})$ is the score of the tag y_k at the k time step. Then, the conditional probability is obtained by a normalization operation:

$$P(\mathbf{y}|\mathbf{x}) = \frac{\exp(S(\mathbf{x},\mathbf{y}))}{\sum_{\widetilde{\mathbf{y}}\in\mathcal{Y}}\exp(S(\mathbf{x},\widetilde{\mathbf{y}}))}. \quad (2)$$

where $\mathcal{Y}$ contains all possible paths of tag sequences. During inference, the predicted tag sequence $\hat{\mathbf{y}}$ is obtained by:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in\mathcal{Y}} P(\mathbf{y}|\mathbf{x}). \quad (3)$$

We adopt BERT(Devlin et al., 2019) and BERT+LSTM(Hochreiter et al., 1997) as the language encoder respectively, resulting in two solutions: BERT+CRF and BERT+LSTM+CRF. The reason for adding LSTM is that we believe that the contextual representation refined by LSTM could be more sensitive to the position of tokens.

3.2 Span Boundary Detection

Different from SL formalization, SBD formalization utilizes the start and end positions tagging scheme to represent toxic spans. SBD formalization was originally applied in the machine reading comprehension task (Seo et al., 2016; Wang and Jiang, 2016). In these works, two n-classifiers are employed to predict the start position and end position separately, where n denotes the length of the input sentence. However, this strategy can only output a single span for an input sentence. Later, Hu et al. (2019b) extended the two n-classifiers strategy by a heuristic multi-span decoding algorithm. But this is not a concise and efficient solution for multi-span scenario, as the decoding algorithm relies on two hyper-parameters: (1) γ, the minimum score threshold, (2) K, the maximum number of spans. In addition to the two n-classifiers strategy, a more recent and popular strategy is to employ two binary classifiers to determine whether each token is the start (end) position or not (Li et al., 2020; Wei et al., 2020; Yu et al., 2019). In this paper, we adopt the binary classifiers strategy for SBD formalization and describe the details below.

Split	Train	Dev	Test
Num	6894	1723	2000

Table 1: Data statistics.

Given the contextual representation $\mathbf{x} = \{x_1, x_2, \cdots, x_n\} \in \mathbb{R}^{n \times h}$, for the location i, we calculate the probability of whether it is a start position by Equation (4) and the probability of whether it is a end position by Equation (5).

$$p_{\text{start}}(i) = \sigma(W_1^\top x_i + b_1), \quad (4)$$

$$p_{\text{end}}(i) = \sigma(W_2^\top [x_i; p_{\text{start}}(i)] + b_2), \quad (5)$$

where $W_1 \in \mathbb{R}^{h \times 1}$, $W_2 \in \mathbb{R}^{(h+1) \times 1}$ and $b_1, b_2 \in \mathbb{R}$ are model parameters.

The predictions of start and end positions are obtained by:

$$starts = \{i | p_{\text{start}}(i) > 0.5, i = 1, \cdots, n\}, \quad (6)$$

$$ends = \{i | p_{\text{end}}(i) > 0.5, i = 1, \cdots, n\}. \quad (7)$$

Then we adopt the nearest start-end matching strategy: for each predicted start position $s \in starts$, the nearest predicted end position e to the right of s is selected to formal a predicted span (s, e).

Similarly, we adopt BERT as the language encoder, and we call this model as BERT+Span.

3.3 Ensemble Strategy

Voting method is applied to integrate the results. In detail, for k different models, if no less than $k/2$ models consider a character to be in the toxic span, the character is retained.

4 Experimental Setup

4.1 Data

The given trial data and training data are merged and the duplicates are removed. In addition, we fix some annotation errors, such as the partially-labeled words. 80% of the processed data is utilized for training and the rest is the validation set. Table 1 shows the statistics of the data used.

4.2 Parameter Settings

We find that the parameter size of the pre-trained model does not have a significant effect on performance, and therefore we simple adopt BERT-base as the our language encoder, which consists of 12 transformer blocks with 12 representation heads. Three models are trained separately. The learning rate of BERT is set to 2e-5, the learning rate of CRF is set to 5e-3, and the maximum encoding length is 128. The weight decay is set to 0.01.

	P(%)	R(%)	F1(%)
BERT+LSTM+CRF	71.99	89.96	69.34
BERT+CRF	74.50	88.10	69.44
BERT+Span	76.29	86.77	69.34
Ensemble	75.01	89.66	70.83

Table 2: Performance of three benchmark models and ensemble approach.

4.3 Evaluation Metrics

We use the official metric, i.e. char-level $F1$-score, as the evaluation metric. In addition, for a more detailed analysis, we also introduce character-level Precision (P) and Recall (R). Note that $F1/P/R$ is the average over the samples, so there is no $F1 = 2PR/(P + R)$.

5 Results

5.1 Ensemble Approach

Table 2 shows the performance of three benchmark models and the ensemble approach. The experimental results show that all three models achieve similar results on $F1$-score, and integrating them results in an improvement of more than 1%, indicating that the predictions of the three models have good complementarity.

To further analyze the differences and respective advantages of SL and SBD formalization, we list their performances in single-span scenario and multi-span scenarios in Figure 2. It could be found that SBD formalization is more advantageous in single-span scenario, while SL formalization is more advantageous in multi-span scenario, which is consistent with our claim.

5.2 Lexicon-based Approach

We also explore a lexicon-based approach for predicting toxic spans. A toxic lexicon is mined from training data by a simple statistical strategy. More Specifically, the toxic score of a word w is defined as below:

$$\text{toxic\_score}(w) = \frac{\#w\_\text{in\_toxic\_span}}{\#w\_\text{in\_whole\_corpus}}, \quad (8)$$

where $\#w\_\text{in\_toxic\_span}$ is the count of appearances of word w in toxic spans, and $\#w\_\text{in\_whole\_corpus}$ is the count of appearances

	# of words	P(%)	R(%)	F1(%)
Ensemble	-	75.01	89.66	70.83
Lexicon[1](Wiegand et al., 2018)	551	75.13	44.47	33.07
Lexicon[2](Wiegand et al., 2018)	2989	66.22	72.01	50.98
Lexicon^{original}(Our)	119	76.71	82.22	64.98
Lexicon^{wordnet}(Our)	231	72.56	84.05	64.09
Lexicon^{glove}(Our)	186	73.98	83.34	64.19

Table 3: Results of Lexicon-based approaches and ensemble model on Precision, Recall and F1. Lexicon[1] and Lexicon[2] are two external lexicons. Lexicon^{original} is collected by ourselves from training set. Lexicon^{wordnet} and Lexicon^{glove} are expanded from Lexicon^{original} with WordNet and GloVe.

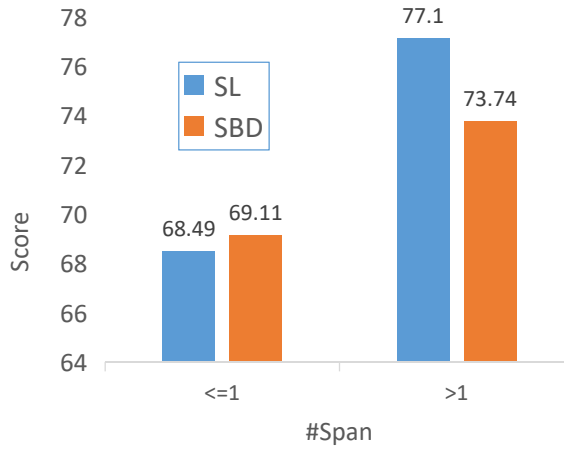

Figure 2: Comparison of the performance of SL and SBD method for data with different numbers of Spans.

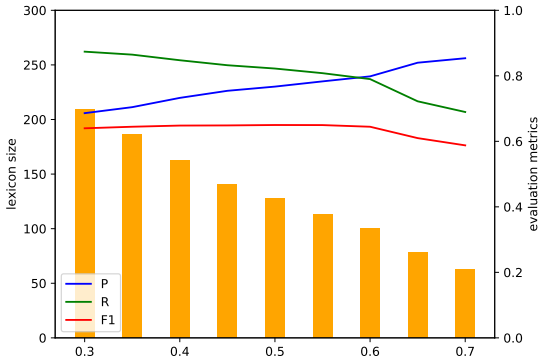

Figure 3: Performances of Lexicon^{original} model with different threshold.

of word w in the whole corpus. Then those words with a toxic score greater than a given threshold θ are selected from a lexicon.

When predicting, the words in the sentence that appear in that toxic lexicon are extracted as the predicted toxic spans. There are three lexicons in our experiment, two of which were collected by (Wiegand et al., 2018), another is collected by ourselves from the training set.

Table 3 shows the results of the lexicon-based approaches and the ensemble approach, and we can observe that our lexicon-based approaches obtain notable results in the $F1$-score. In addition, we also calculate the average precision and average recall values of different methods on the test set, and our original lexicon-based approach even outperforms ensemble approaches in average precision, but there is still a significant gap in an average recall. Since the lexicon-based approaches can only identify the toxic words in the lexicon, the recall can be improved by expanding the toxic lexicon.

To improve the recall, we use WordNet (Miller, 1995) and GloVe (Pennington et al., 2014) to expand the toxic lexicon. In detail, we collect synsets of each toxic from WordNet, and collect the nearest similar words by calculating cosine similarity of GloVe vectors. The performances of the two expanded approaches are shown in Table 3. Although the recall of two approaches improves over the original lexicon, the precision decreases significantly, which indicating that there are a considerable number of non-toxic words in the synonyms found through WordNet.

Besides, we explore the impact of threshold θ when mining the original lexicon on performance. The performances with different threshold is shown on Figure 4. As the threshold θ increases, the size of lexicon decreases, P decreases, R increases, $F1$ increases and then decreases, reaching a maximum 64.98 when $\theta = 0.5$.

6 Conclusion

In this paper, we formalize the toxic span detection as two problems separately and employ three state-of-the-art models. The strengths of each model are analyzed and a more credible and complement result is obtained through a voting approach. Our re-

sults achieve a good score (ranking 1/91). Besides, we explore a lexicon-based approach. The lexicon is mined from the annotation of the training data and then expanded by WordNet and Glove. Experiments show that the lexicon-based approach has not yet achieved the performance of the ensemble approach. We believe that future work could move towards combining deep learning-based methods and lexicon-based methods.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (61632011, 61876053, 62006062), the Guangdong Province Covid-19 Pandemic Control Research Funding (2020KZDZX1224), the Shenzhen Foundational Research Funding (JCYJ20180507183527919 and JCYJ20180507183608379), and China Postdoctoral Science Foundation (2020M670912).

References

Segun Taofeek Aroyehun and Alexander Gelbukh. 2018. Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC)*, pages 90–97.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11.

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*, pages 4171–4186.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243.

Sepp Hochreiter, J urgen Schmidhuber, and Corso Elvezia. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019a. Open-domain targeted sentiment analysis via span-based extraction and classification. *arXiv preprint arXiv:1906.03820*.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019b. Open-domain targeted sentiment analysis via span-based extraction and classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 537–546.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC)*, pages 1–11.

John Lafferty, Andrew Mccallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.

Kenton Lee, Shimi Salant, Tom Kwiatkowski, Ankur Parikh, Dipanjan Das, and Jonathan Berant. 2016. Learning recurrent span representations for extractive question answering. *arXiv preprint arXiv:1611.01436*.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified mrc framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1064–1074.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Sandip Modha, Prasenjit Majumder, and Thomas Mandl. 2018. Filtering aggression from the multilingual social media feed. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 199–207.

John Pavlopoulos, Léo Laugier, Jeffrey Sorensen, and Ion Androutsopoulos. 2021. Semeval-2021 task 5: Toxic spans detection (to appear). In *Proceedings of the 15th International Workshop on Semantic Evaluation*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905*.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488, Online. Association for Computational Linguistics.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. 2018. Inducing a lexicon of abusive words – a feature-based approach. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 1046–1056.

Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore. 2012. Learning from bullying traces in social media. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 656–666.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Yubin Wang, Tingwen Liu, Bin Wang, and Sujian Li. 2019. Joint extraction of entities and relations based on a novel decomposition strategy. *arXiv preprint arXiv:1909.04273*.

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*.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). *arXiv preprint arXiv:2006.07235*.