UIT-E10dot3 at SemEval-2021 Task 5: Toxic Spans Detection with Named Entity Recognition and Question-Answering Approaches

Phu Gia Hoang[1, 2], **Luan Thanh Nguyen**[1, 2], **and Kiet Nguyen**[1, 2]

[1]University of Information Technology, Ho Chi Minh City, Vietnam
[2]Vietnam National University Ho Chi Minh City, Vietnam
{19520215, 17520721}@gm.uit.edu.vn, kietnv@uit.edu.vn

Abstract

The increment of toxic comments on online space is causing tremendous effects on other vulnerable users. For this reason, considerable efforts are made to deal with this, and SemEval-2021 Task 5: Toxic Spans Detection is one of those. This task asks competitors to extract spans that have toxicity from the given texts, and we have done several analyses to understand its structure before doing experiments. We solve this task by two approaches, Named Entity Recognition with spaCy's library and Question-Answering with RoBERTa combining with ToxicBERT, and the former gains the highest F1-score of **66.99%**.

1 Introduction

The world of social media is overgrowing, and users easily express their opinions or feelings toward topics that they are concerned about. However, because of the freedom of speech, lots of toxic comments or contents are uncontrollably increasing. There are several kinds of research about the effect of toxic speech on users' health. In 2017, research about the impact of toxic language on health was conducted (Mohan et al., 2017). Sometimes, with toxic words, conversations can become cyberbullying, cyber threats, or online harassment, which are harmful to users. To reduce those negative impacts, there are abundant researches for classifying contents into toxic or non-toxic, and then they hide the whole text if it is toxic. However, that action may inhibit the freedom of speech. As a result, censoring only toxic spans is the better solution for this problem. Therefore, in SemEval-2021 Task 5: Toxic Spans Detection (Pavlopoulos et al., 2021) we try to realize it.

About toxic contents on the internet, researches were only about binary toxicity classification. Still, in task 5 of SemEval-2021, which is about toxic spans detection, we conduct more in-depth research into the toxicity, find exactly which parts of the text are toxic. As the NER approach and Question-Answering (QA) approach, we propose two approaches for solving this problem. We use RoBERTa (Liu et al., 2019) combining with ToxicBERT (Hanu and Unitary team, 2020), transfer learning models, for QA approach and spaCy's library (Honnibal and Montani, 2017) for NER approach.

We organize the paper as follows. Section 2 is related works that we consult for building the systems. The dataset and analyses are defined in Section 3. In section 4, we introduce our two proposed systems for toxic spans detection. Section 5 describes the results of the studies and analyses. Finally, in Section 6, we bring our work to a close.

2 Related Works

Researchers around the world these days have started to concentrate on toxic speech. It inflicts individual and group harm, damaging our social fabric (Tirrell, 2018). Several datasets for classifying toxicity on toxic speech on online forums, such as the dataset provided by Waseem and Hovy (2016) for English, BEEP! dataset for Korean by Moon et al. (2020), the dataset for Russian provided by Smetanin (2020), TolD-Br dataset for Brazilian Portuguese by Leite et al. (2020), and UIT-ViCTSD, a dataset about constructive and toxic speech detection for Vietnamese (Nguyen et al., 2021).

Besides, there are shared tasks about toxic speech as well as hate speech such as these from SemEval, includes SemEval-2019 Task 5 Multilingual Detection of Hate (Basile et al., 2019), SemEval-2019 Task 6 Identifying and Catego-

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 919–926
Bangkok, Thailand (online), August 5–6, 2021.

rizing Offensive Language in Social Media (OffensEval) (Zampieri et al., 2019), SemEval-2020 Task 12 Multilingual Offensive Language Identification in Social Media (Zampieri et al., 2020), and SemeEval-2021 Task 5 Toxic Spans Detection (Pavlopoulos et al., 2021), which is the current task we have to deal with in this paper.

3 Dataset

The origin of this SemEval-2021 Task 5 dataset comes from the publicly available Civil Comments dataset (Borkan et al., 2019), which consists of 1.2M posts and comments. The data in this public dataset have no annotation of any toxic spans in toxic posts but do have post-level toxicity annotations, which mean showing which posts or entire of them are toxic. And the holders of this task retain 30K of them, which were annotated to be toxic or severely toxic by at least half of the crowd-raters from annotations of Borkan et al.

The task holders then randomly keep 10K posts from the 30K posts for annotating toxic spans. They employ three experienced crowd-raters per post from a third-party crowd-annotation platform, and they warn them about adult content. However, task organizers also claim that not all toxic posts are annotated with toxic spans.

The task for crowd-raters is to highlight toxic sequences of the comments, and if the comment is not toxic or should annotate the whole of it, crowd-raters have to check the appropriate box without highlighting any spans. Consequently, we have two columns, the spans column and the text column. The spans column has lists of numbers or null that reference toxic character offsets in the text column, and some of the given data are shown in the following table.

spans	text
[7, 8, 9, 10, 11, 12]	Pretty **damned** eloquent ... :)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]	**He might fire you to the moon**, but you already have a head full of cheese!
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 98, 99, 100, 101, 102, 103, 104, 105, 106]	**Nauseating** and **disgusting**. Thank goodness the First Amendment permits people to demonstrate their **stupidity**.
[]	Not if they shoot you first...

Table 1: Examples for the given data.

The competitors receive two separate training and test sets from organizers. In the training set, there are 7,939 records, and in the test set, there are 2,000 records. Furthermore, as mentioned in the data annotating process, one text that possibly has multiple toxic spans is highlighted. Figure 1 and Figure 2 illustrate the distribution of spans in the training and the test sets.

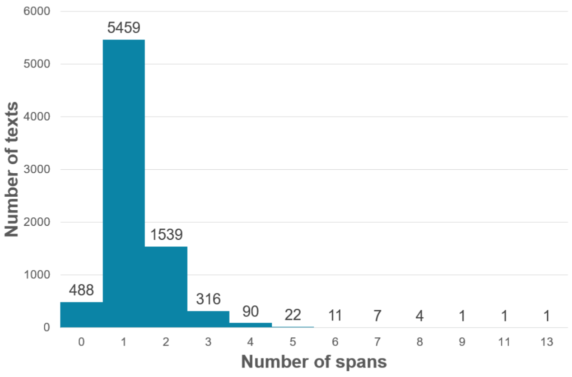


Figure 1: Distribution of spans in the training set.

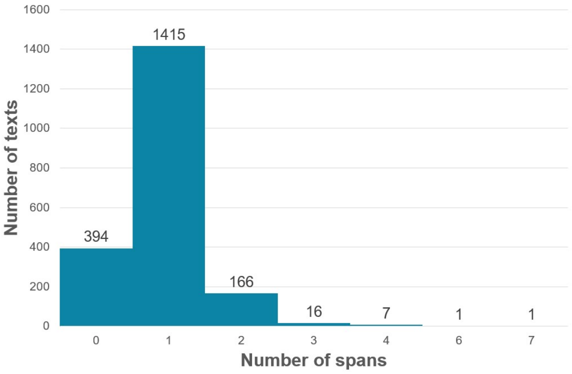


Figure 2: Distribution of spans in the test set.

For more details, according to Figure 1 and Figure 2, there is a significant number of single spans in each post, and it accounts for nearly 68.8% and 70.8% in the training set and the test set, respectively. It is also interesting to notice that the number of zero spans is not tiny, and the proportion of it in the training set is less than in the test set, more specifically, 19.7% in the test set and 6.15% in the training set.

Moreover, we also calculated the Jaccard score of text and spans in the given dataset for more in-depth analysis. The Jaccard score, also known as the Jaccard index or Jaccard similarity coefficient, was developed by Paul Jaccard (Jaccard, 1912) and it is a statistic used for measuring the similarity and diversity of sample sets as follows.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (1)$$

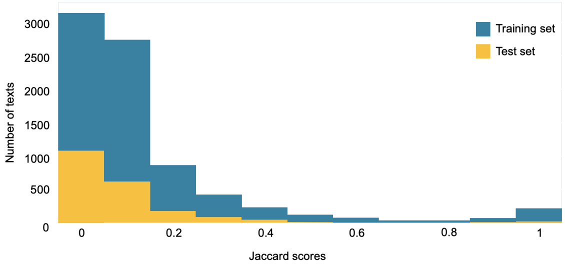


Figure 3: Histogram of Jaccard score of each record in the training set and the test set.

The histogram in Figure 3 illustrates that most of the data points have Jaccard scores in the range of 0 to 0.35, and the peak is at 0 to 0.05, which means toxic character offsets are just a fraction in each post even there are records annotated all characters of the post are toxic. There are 16 records in the test set and 212 records in the training set with Jaccard scores at 0.95 to 1.0. For that reason, just the toxic part(s) of the comments needs to be censored rather than the whole comment as in the traditional method.

4 Systems

In this paper, we propose two systems for the toxic spans detection task with NER and QA approaches. The first system is the QA approach based on RoBERTa and the second system is the NER approach based on spaCy's library.

4.1 Question-Answering Approach Based on RoBERTa

With the QA approach, we use RoBERTa combining with ToxicBERT as the basis for the system. RoBERTa (Liu et al., 2019) is a transfer learning model and it is a replication study of BERT (Devlin et al., 2019). Unlike BERT, to improve the training performance, RoBERTa eliminates the Next Sentence Prediction (NSP) task of the pre-trained model BERT. ToxicBERT (Hanu and Unitary team, 2020) is also a transfer learning model, and it uses BERT as the main model for classifying toxicity. ToxicBERT has an outstanding performance for the task of Jigsaw Unintended Bias in Toxicity Classification [1] on Kaggle, which uses the same dataset with SemEval-2021 Task 5, with 93.64% F1-score. We use two models for our QA approach system, and the overview of the system with training and testing phases is described in Figure 4.

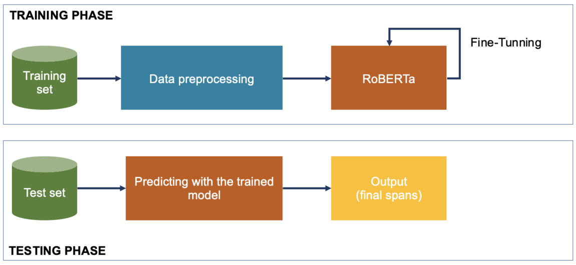
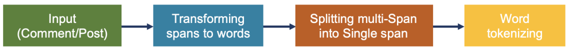


Figure 4: Training and testing phases of toxic spans detection with RoBERTa based system.



Input (Comment/Post)
Transforming spans to words
Splitting multi-Span into Single span
Word tokenizing

Figure 5: Data preprocessing of RoBERTa based system for toxic spans detection.

Firstly, we preprocess the training set with techniques to get the right format for the RoBERTa model, mentioned in Figure 5. The model we used only approve one spans, but several examples have more than one in the training set, and we called it "multi-span". Hence, we split multi-span (*) into single spans (**) (***) as below.

- Plain text:

  (*) This bitch is so fucking idiot.

- After splitting:

  (**) This **bitch** is so.

  (***) This is so **fucking idiot**.

After splitting texts, we tokenize the dataset with a subword model as Byte-Pair Encoding (BPE) (Sennrich et al., 2016). Then, we feed the data into a pre-trained RoBERTa model and fine-tune it with suitable parameters. We analyze the length of the texts in the dataset and set max_length=512 and epochs=5 for the model. After searching for extensive hyper-parameters, we set the learning_rate and drop_out equal to 3e-5 and 0.1, respectively. We also train the model with 5-fold cross-validation. After the training phase, the trained RoBERTa model is used for predicting new toxic spans.

[1] https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification

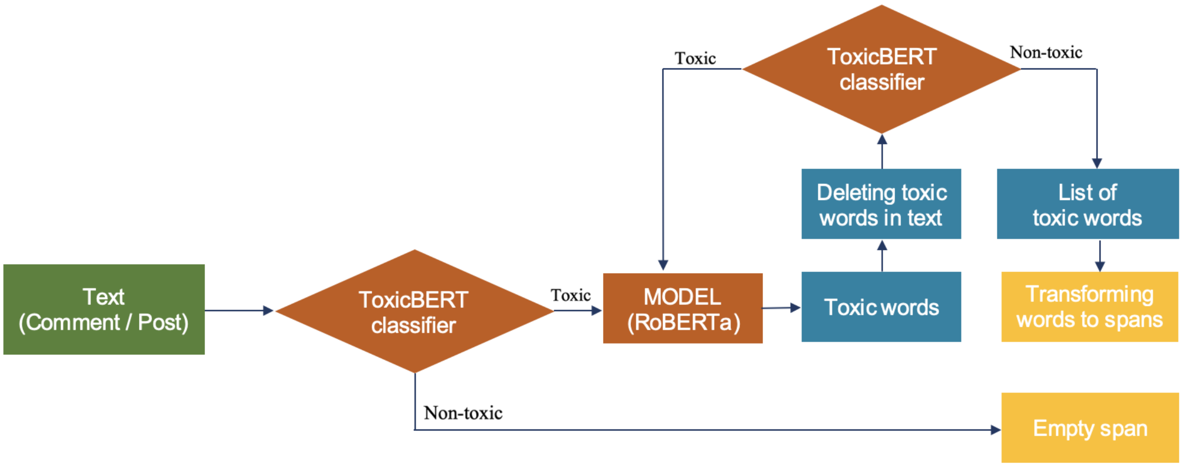


Figure 6: Predicting toxic spans with trained model by QA approach.

In the testing phase, besides using RoBERTa, we use another transfer learning model is ToxicBERT (Hanu and Unitary team, 2020) for identifying toxic comments. With ToxicBERT, we classify the input text into toxic or non-toxic labels before predicting spans. If the result is non-toxic, we stop the prediction, and the result is an empty spans. If it is toxic, we feed the text into the RoBERTa model to predict toxic words. After having the spans, to ensure that the text still has toxic words, we remove the predicted toxic word(s) from the processing text and then recheck its toxicity by ToxicBERT and re-predict its remaining toxic words (if any).

Because final results are words, we transform them into spans for the requirement of this task.

4.2 NER Approach Based on spaCy's Library

In this approach, we tag all the characters spans with text as TOXIC to train the model, and we predict all TOXIC tags in the text set of texts.

For solving this, we choose version 2.2.5 of spaCy's NER Model (Honnibal and Montani, 2017) because of its exceptionally efficient statistical system in both speed and accuracy for this named-entity recognition. Apart from default entities such as location, person, organization, and so on, spaCy also enables training the model with new entities by updating it with newer examples.

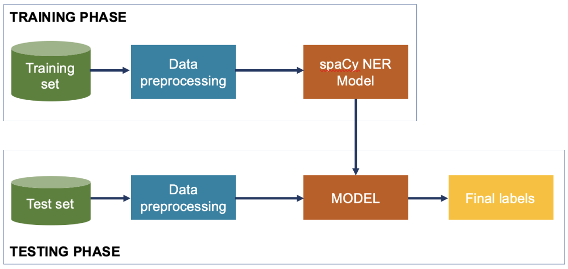
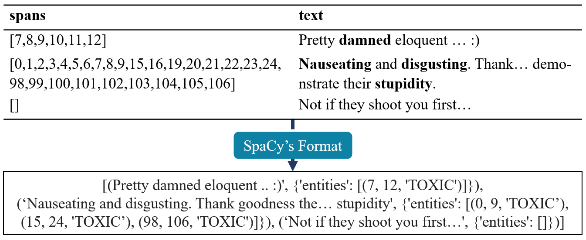


Figure 7: Training and testing phases of toxic spans detection with spaCy based system.

The above Figure 7 shows the process of our spaCy based system. Both training and test sets have to be tokenized before feeding them into the spaCy NER model or being predicted by the TOXIC entities. For more details, in the training phase, the input data have to be in the right format for the spaCy NER model as in the following Figure 8.



spans	text
[7,8,9,10,11,12]	Pretty **damned** eloquent … :)
[0,1,2,3,4,5,6,7,8,9,15,16,19,20,21,22,23,24, 98,99,100,101,102,103,104,105,106]	**Nauseating** and **disgusting**. Thank… demonstrate their **stupidity**.
[]	Not if they shoot you first…

[(Pretty damned eloquent .. :)', {'entities': [(7, 12, 'TOXIC')]}), ('Nauseating and disgusting. Thank goodness the… stupidity', {'entities': [(0, 9, 'TOXIC'), (15, 24, 'TOXIC'), (98, 106, 'TOXIC')]}), ('Not if they shoot you first…', {'entities': []})]

Figure 8: Process of re-formatting data for spaCy based system.

SpaCy has not published the architecture of their models yet, but they do have a brief explanation about how their models work, especially the NER model, through a four-step formula: embed, encode, attend, and predict.

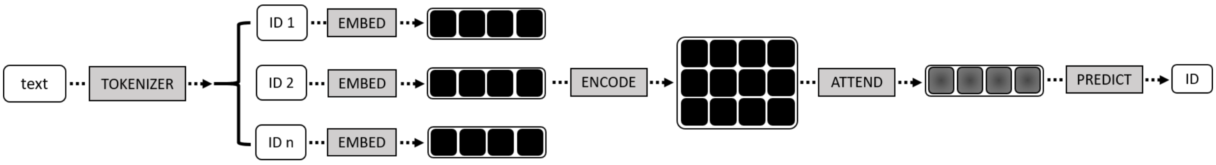


Figure 9: Diagram of how spaCy's models work.

As in Figure 9, spaCy's model is fed with unique numerical values (ID) which address a token of a corpus or a class of the NLP task (named entity class). In the first embed stage, word similarities are revealed by extracting hash, which is collected by extracting word features as the lower case, the prefix, the suffix, and the shape. The encode stage is fed with a sequence of word vectors from the previous stage to calculate a representation which is named sentence matrix. In the sentence matrix, the meaning of each token in the context of neighboring tokens is represented in each row, and this is done by using a bidirectional RNN (Schuster and Paliwal, 1997). The output matrix from the second stage is injected into the Attention Layer of the CNN after summarized by a query of vectors. Finally, to predict the toxic class, a softmax function is utilized. After the model is trained, the CNN model is now used for the NER task to extract the toxic class.

The given toxic spans dataset is fed into spaCy's library for training with a suitable format. During the contest, my team was using spaCy's library for a small model for English (en_core_web_sm) at version 2.2.5, and we tried different parameters to get the optimal result. When training, the dataset is shuffled and passed through spaCy's training algorithm in batches with an increment of batch sizes from 4.0 to 32.0 and step of 1.001. Moreover, the drop rate is consistently at 0.5, and most of the experiments loop 45 times.

5 Experiments

After building two such systems, we start to experiment on the test set, and the following subsections dicuss our results.

5.1 Evaluation Metrics

Before going through experimental results, we first discuss the evaluation metrics used in this SemEval-2021 Task 5.

In this task, all of the responding systems from participants are evaluated by **F1** score (Da San Martino et al., 2019). Assuming the system S_i returns $C^t_{S_i}$, which is a toxic character offsets of the post. Let G^t be the character offsets of the ground truth annotation of **t**. In the following formulas, the **F1** score of system S_i is computed regarding ground truth **G** of post **t** ($|\cdot|$ indicates set cardinality).

$$F_1^t(S_i, G) = \frac{2 \cdot P^t(S_i, G) \cdot R^t(S_i, G)}{P^t(S_i, G) + R^t(S_i, G)} \tag{2}$$

$$P_1^t(S_i, G) = \frac{\left|C^t_{S_i} \cap C^t_G\right|}{\left|C^t_{S_i}\right|} \tag{3}$$

$$R_1^t(S_i, G) = \frac{\left|C^t_{S_i} \cap C^t_G\right|}{\left|C^t_G\right|} \tag{4}$$

If S^t_G is empty for posts **t**, we set $F_1^t(S_i, G)$ = 1 and if $S^t_{A_i}$ is empty, $F_1^t(S_i, G)$ = 0. Finally, we calculate average of $F_1^t(S_i, G)$ of all over the posts **t** if test set to get a sigle **F1** score of the system S_i.

5.2 Experimental Results

The results of our systems compared with other teams' are shown in Table 2.

Rank	**Team name**		**F1-score**
1	HITSZ-HLT		70.83
2	S-NLP		70.77
33	lz1904		67.00
34	**UIT-E10dot3**	**spaCy**	**66.99**
		RoBERTa	52.12
			17.00±1.

Table 2: The results of our systems compared with other teams by F1-score (%).

During the SemEval-2021 Task 5, with the spaCy base system, we achieved rank 34 out of 91 teams, and in the table above, we have shown our result with the spaCy based system and the RoBERTa based system in comparison with rank 1, 2, 33 and random baseline of this task. The F1-score of our best system is 66.99%, 3.84% lower than the first rank team, and 49.09% higher than the baseline model.

5.3 Result Analyses

After analyzing our most effective system based on spaCy's library, we spot crucial errors in predicting and datasets by comparing predicted spans to gold spans. Several records in the given data are standing alone without the context that leads to confusing or multi-meaning. Moreover, comments are using slang(s) or idiom(s), causing null output for our system. We also realize a lack of consistency or highlighting non-toxic spans when annotating data about the datasets. Likewise, several words in the text have spelling mistakes that intentionally also impair our system performance. Evidence for those errors are in Table 3, Appendix.

6 Conclusion and Future Work

In this paper, we introduced two proposed systems for toxic spans detection based on named entity and question-answering approaches. We obtained the highest results with the SpaCy's library based system with the F1-score of **66.99%** and ranked 34 out of 91 teams in SemEval-2021 Task 5.

In future, we plan to improve our systems by implementing various SOTA models for toxic spans detection. With the built systems, we can create friendly online conversations and make social media forums safer for users.

Acknowledgement

We would like to express our gratitude for the helpful reviews from reviewers.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Nozza Debora, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, Manuela Sanguinetti, et al. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *13th International Workshop on Semantic Evaluation*, pages 54–63. Association for Computational Linguistics.

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.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeno, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis of propaganda in news article. 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)*, pages 5640–5650.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Laura Hanu and Unitary team. 2020. Detoxify. Github. https://github.com/unitaryai/detoxify.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7(1):411–420.

Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50.

João Augusto Leite, Diego Silva, Kalina Bontcheva, and Carolina Scarton. 2020. Toxic language detection in social media for Brazilian Portuguese: New dataset and multilingual analysis. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 914–924, Suzhou, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Shruthi Mohan, Apala Guha, Michael Harris, Fred Popowich, Ashley Schuster, and Chris Priebe. 2017. The impact of toxic language on the health of reddit communities. In *Canadian Conference on Artificial Intelligence*, pages 51–56. Springer.

Jihyung Moon, Won Ik Cho, and Junbum Lee. 2020. BEEP! Korean corpus of online news comments for toxic speech detection. In *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*, pages 25–31, Online. Association for Computational Linguistics.

Luan Thanh Nguyen, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2021. Constructive and toxic speech detection for open-domain social media comments in vietnamese.

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*.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sergey Smetanin. 2020. Toxic comments detection in russian. In *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2020"*.

Lynne Tirrell. 2018. Toxic speech: Inoculations and antidotes. *The Southern Journal of Philosophy*, 56:116–144.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

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). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

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).

No.	Predicted by spaCy base system	Ground Truth
	Requiring context (♠)	
1	No, my poor benighted correspondent, your reductionism reveals an archaic mindset. Intersex, questioning, nongendered, and a vast variety of others have rights too!	No, my poor **benighted** correspondent, your reductionism reveals an archaic mindset. Intersex, questioning, nongendered, and a vast variety of others have rights too!
	Slang(s) / Idiom(s)	
2	Lord have Mercy on us, Trump is running amok.	Lord have Mercy on us, Trump is running **amok**.
3	No, dumdum-you dont get it. The business does not get to decide who he serves and who he does not. If he makes cakes for one, he must make them for ALL	No, **dumdum**-you dont get it. The business does not get to decide who he serves and who he does not. If he makes cakes for one, he must make them for ALL
4	Ugh, the worst, burn them at the stake.	Ugh, the worst, **burn them at the stake**.
	Lacking of consistency or highlighting non-toxic spans when annotating (♣)	
5	There is even a website where sore loser Democrats can pretend that Hillary won. La-la-land indeed! #liberaltears #salt #schadenfreude	There is even a website where sore **loser** Democrats can pretend that Hillary won. La-la-land indeed! #liberaltears #salt #schadenfreude
6	Hey Repubes, I bet you think that "wall" is going to be built on Mexicos dollar too huh? What a bunch of gullible **losers**!!!!	Hey Repubes, I bet you think that "wall" is going to be built on Mexicos dollar too huh? What a bunch of gullible losers!!!!
7	<Trump was a draft dodger during Vietnam.>The president is a coward and a hypocrite who not only dodged the draft and avoided any and every opportunity to serve... No one has more contempt for the flag than does Donald J. Trump. No one.	<Trump was a draft dodger during **Vietnam.>** The president is a coward and a hypocrite who not only dodged the draft and avoided any and every opportunity to serve... No one has more contempt for the flag than does Donald J. Trump. No one.
	Spelling mistake	
8	FUKC YOU @QuinHillyer YOU ARE THE ONE NEED TO GO, YOU ANTI AMERICA FAKE MOTHER FUKCER	**FUKC YOU** @QuinHillyer YOU ARE THE ONE NEED TO GO, YOU ANTI AMERICA FAKE **MOTHER FUKCER**
9	Hey Repubes, I bet you think that "wall" is going to be built on Mexicos dollar too huh? What a bunch of gullible **losers**!!!!	Hey Repubes, I bet you think that "wall" is going to be built on Mexicos dollar too huh? What a bunch of gullible losers!!!!

Table 3: Examples for result analyses.

(♠) The table shows that with this example, our system predicts that there are no toxic spans; meanwhile, in ground truth, the word **benighted** is highlighted to be toxic. We assume that because of lacking the context of the text and the word **benighted** also has multi-meaning.

(♣) The table shows that in example No. 5, the word **loser** is annotated to be toxic when in example No. 6, also having the plural form of word **loser** but not to be highlighted. Meanwhile, in example No. 7, the spans **Vietnam.>** is highlighted even if it does not have toxicity.