Sefamerve ARGE at SemEval-2021 Task 5: Toxic Spans Detection Using Segmentation Based 1-D Convolutional Neural Network Model

Selman Delil[*] **Birol Kuyumcu**[*] **Cüneyt Aksakallı** [*]

Sefamerve R&D Center
Istanbul, Turkey
{selman.delil, birol.kuyumcu, cuneyt.aksakalli}@sefamerve.com

Abstract

This paper describes our contribution to SemEval-2021 Task 5: Toxic Spans Detection. Our approach considers toxic spans detection as a segmentation problem. The system, Waw-unet, consists of a 1-D convolutional neural network adopted from U-Net architecture commonly applied for semantic segmentation. We customize existing architecture by adding a special network block considering for text segmentation, as an essential component of the model. We compared the model with two transformers-based systems RoBERTa and XLM-RoBERTa to see its performance against pre-trained language models. We obtained 0.6251 f1 score with Waw-unet while 0.6390 and 0.6601 with the compared models respectively.

1 Introduction

Unlike the text classification problems targeting to classify whole documents (Borkan et al., 2019; Schmidt and Wiegand, 2017; Pavlopoulos et al., 2019), toxic span detection is an NLP task focusing on capturing granular contents that make a text toxic. Proposed solutions may contribute to managing semi-automated moderations such as online discussions or news portals that are open to large participation and user comments. Therefore, the evaluation of systems that could accurately locate toxic spans within a text is considered a crucial step for this task (Pavlopoulos et al., 2021).

We adapt two solution approaches for the task. For the first approach, we consider toxic spans detection as a segmentation problem while in the second one we use transformers-based models. Our proposed model for the first approach uses character-based tokenized chunks as an input and outputs segmented text. The system uses a 1-dimensional (1-D) convolutional neural network adopted from U-Net architecture (Ronneberger et al., 2015) commonly applied for semantic segmentation. We previously studied this approach, as we call Waw-unet, on text parsing problems for unstructured postal addresses, and achieved remarkable results (Delil et al., 2020).

In our second approach, we consider toxic spans as a single label Named-Entity Recognition problem. We employ several different transformers-based models and obtained better scores with RoBERTa (Liu et al., 2019) and XLM-RoBERTa (Conneau et al., 2020) network architectures. To see the difference between our main system and transformers-based models, we compared model achievements, and obtained 0.6251 f1 score with Waw-unet while 0.6390 and 0.6601 with the other models respectively. Following the final rankings, our best score ranked 44th among 91 submissions[1].

2 Waw-unet Architecture

Waw-unet is a fully convolutional neural network architecture we designed by taking inspiration from U-Net architecture which was firstly developed for segmentation problems (Ronneberger et al., 2015). The U-Net network architecture is composed of two symmetric parts, which uses dimension reduction for the first half of the network, and then increases its dimension in the second half. In this architecture, the connections are taken from the convolutional layers on the encoding part, which also feeds each corresponding layer of the decoding part.

Similar to the pixel-based image segmentation, Waw-unet takes input samples, in our case text, and generates homogeneous masked regions for the targeted segment. However, unlike image processing which has multi-channel input, the network has 1-D input due to the single-dimensional nature of text

[1] Source code for our model is published on https://github.com/birolkuyumcu/wawunet_for_toxicspan

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 909–912
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

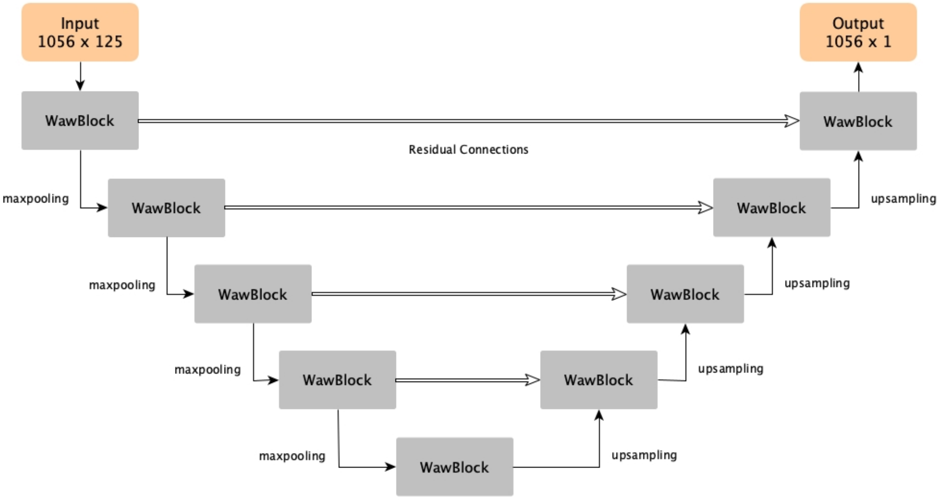


Figure 1: Waw-unet network architecture

data (Fig. 1). The outputs of the convolution layers are combined and passed through a 1-D convolution layer, and then we use batch normalization to accelerate training and prevent the objective function from getting stuck in local minima. After this stage, before the outputs send to the next block, if the output is in the encoder part their dimensions are reduced in half using max pooling, while doubled if it's in the decoder part by upsampling.

Although its architecture adapted from U-Net, we customize existing architecture by adding a special network block for text segmentation, as an essential component of the model. Waw-unet uses a special network block, which we call it Waw-block, to extract the attributes in the targeted text patterns. In our architecture, each waw-block contains three convolutional layers with different kernel sizes. Waw-unet learns input features through filter sizes of the 3, 5, and 7 as shown in Fig. 2.

2.1 Data Preparation

As we use a character-based system in our model, the total number of characters in the dataset and the maximum character length for each sample need to be determined. In the training dataset, the former was 1047, while the latter calculated as 125. Since our model has encoder-decoder architecture, to prevent matrix dimension problems, the input size has to be selected so that it can be divided by 2 until the end of the encoder part. Therefore, we defined max input size, the closest value as 1056, and the input matrix dimension as 1056 x 125. In accordance with the segmentation logic, the output character positions contain toxic spans masked as 1 and the other parts of the text masked as 0 (Fig. 3).

2.2 Model Training

The Tversky similarity index (TI) is used to calculate the loss function for training the network. It is an asymmetric similarity measure that is a generalization of Dice coefficient and Jaccard index (Tversky, 1977). To define Tversky loss function we use the following formulation:
TI : Tversky Index
TP : True Positive
FP : False Positive

$$TI = TP/(TP + a * FN + b * FP)$$

$$b = 1 - a$$

Here, we use $1 - TverskyIndex$ as the loss function. The parameters a and b are used to provide weight to the represented classes. In our case, we determine $a = 0.7$ to give weight to the false-negative classification so that the loss function is modified accordingly. Additionally, Dice similarity coefficient was used as a metric to judge the performance of the model training.

3 Transformers Models for NER

Toxic span detection can be adopted to NER problems by considering targeted toxic part of text as a predefined named-entity. We experiment with transformers models as an alternative for our model to see its performance. We use pre-trained models RoBERTa (Liu et al., 2019) and XLM-RoBERTa (Conneau et al., 2020) in our study utilising HuggingFace Trainer class.

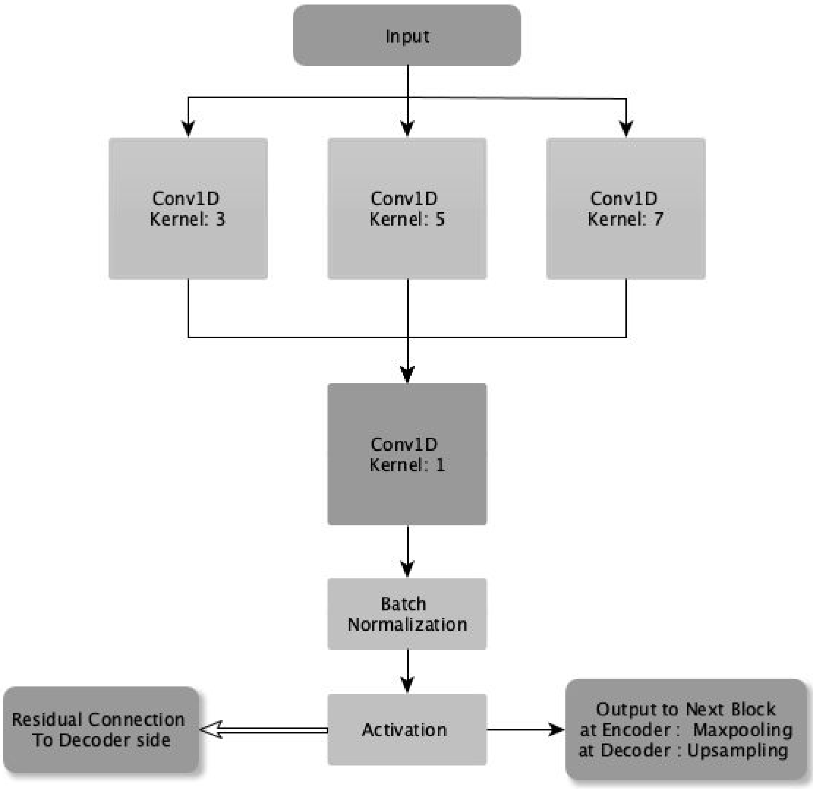
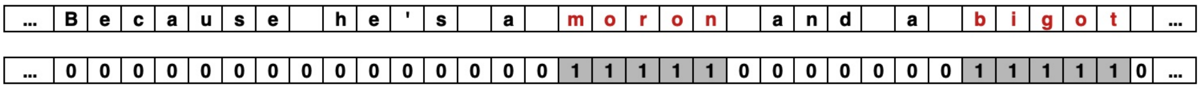


Figure 2: Waw-block architecture



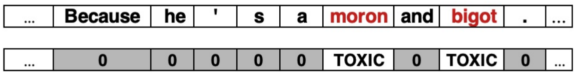
...	B	e	c	a	u	s	e		h	e	'	s		a		m	o	r	o	n		a	n	d		a		b	i	g	o	t		...
...	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	0	...

Figure 3: Waw-unet: Character-based masking

RoBERTa is an extension of pre-trained transformer model BERT with over more data and some configuration changes to the pre-training stages. The modifications include such as training longer and bigger batches, dynamically changing the masking pattern, and training on longer sequences (Liu et al., 2019). On the other hand, XLM-RoBERTa uses self-supervised training techniques designated to solve the cross-lingual understanding task. The model improves upon previous multilingual approaches by incorporating more training data and languages (Conneau et al., 2020).

To prepare toxic spans dataset for training, word labeling operation carried out by converting toxic spans into toxic words based on whether more than 50% of their characters labeled as toxic (Fig.4).

We use the Simple Transformers library (Rajapakse, 2019) to prepare our data for pre-trained models. Tokenized input containing the 3 columns—sentence_id, words, and labels. Each value in words has a corresponding label value. In this data format, the sentence_id determines which words belong to a given sentence (Fig. 5). We use the maximum sequence length of 128 for training and evaluation dataset.



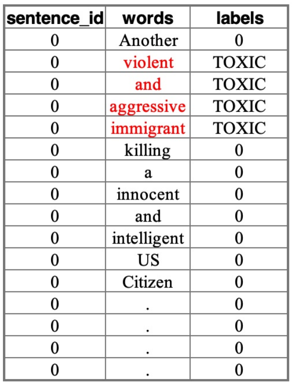
...	Because	he	'	s	a	moron	and	bigot	.	...
...	0	0	0	0	0	TOXIC	0	TOXIC	0	...

Figure 4: Transformers NER model: Word-based entity labeling

4 Results

We evaluate our system as well as transformers models' performance on the SemEval-2021 Task 5: Toxic Span Detection trail dataset and also report the evaluation result on the blind test dataset. We use standard train/test split of the official release dataset of the Task for experiments.

For our main system model, Waw-unet, we start model training with 300 epochs and utilize early



sentence_id	words	labels
0	Another	0
0	violent	TOXIC
0	and	TOXIC
0	aggressive	TOXIC
0	immigrant	TOXIC
0	killing	0
0	a	0
0	innocent	0
0	and	0
0	intelligent	0
0	US	0
0	Citizen	0
0	.	0
0	.	0
0	.	0
0	.	0

Figure 5: Transformers NER model: Input data format

stopping and learning pause using Keras's learning callbacks (Chollet et al., 2015). On the other hand, we trained transformers models with 8 batch size with 17 epochs. We gained the best score in 7th epoch on both pre-trained language models.

XLM-RoBERTa model gained best score 0.6601 while waw-unet and RoBERTa reached 0.6251 and 0.6390 respectively as shown in Table 1.

F1 For	Waw-Unet	RoBERTa	XLM-RoBERTa
Train	0.812	0.803	0.806
Trial	0.602	0.645	0.643
Test	0.625	0.639	0.660

Table 1: Model results

5 Conclusion

We framed the problem as a semantic segmentation task, and developed a unique approach to extract targeted spans from provided text data. Proposed system performs relatively well than expected against pre-trained transformers. Our models do not use any of the external dataset or automatic linguistic annotations, such as PoS or named entity tags. Overall, we showed that segmentation based systems can be used to address the toxic detection task. Our best submitted result was ranked 44th among 91 submissions, obtaining an average F1 score of 0.6601, 4.82 points behind the first ranked system.

For future studies, we're planning to work on unsupervised training of the Waw-unet architecture on large datasets to compete with the pre-trained general language models.

References

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.

François Chollet et al. 2015. Keras. `https://keras.io`.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale.

S. Delil, B. Kuyumcu, C. Aksakallı, and İ. S. Akçıra. 2020. Parsing address texts with deep learning method. In *2020 28th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

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.

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*.

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, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

T. C. Rajapakse. 2019. Simple transformers. `https://github.com/ThilinaRajapakse/simpletransformers`.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham. Springer International Publishing.

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, Valencia, Spain. Association for Computational Linguistics.

Amos Tversky. 1977. Features of similarity. *Psychological review*, 84(4):327.