Cyber-aggression Detection using Cross Segment-and-Concatenate Multi-Task Learning from Text

Ahmed Husseini Orabi, Mahmoud Husseini Orabi, Diana Inkpen, Qianjia Huang
University of Ottawa
{ahuss045, mhuss092, diana.inkpen, qhuan035}@uottawa.ca

David Van Bruwaene
SafeToNet Canada
dvanbruwaene@safe-tonet.com

Abstract

In this paper, we propose a novel deep-learning architecture for text classification, named cross segment-and-concatenate multi-task learning (CSC-MTL). We use CSC-MTL to improve the performance of cyber-aggression detection from text. Our approach provides a robust shared feature representation for multi-task learning by detecting contrasts and similarities among polarity and neutral classes. We participated in the cyber-aggression shared task under the team name uOttawa. We report 59.74% F1 performance for the Facebook test set and 56.9% for the Twitter test set, for detecting aggression from text.

1 Introduction

Deep neural networks provide multiple structures that help learn abstract features from text, which is beneficial for NLP. However, deep neural networks are more likely to have overfitting issues more than traditional approaches. Multi-task learning (MTL) (Caruana, 1993) provides a solution to overcome such a limitation, due to its ability to provide transferable learning models (Baxter, 1997; Mou et al., 2016). We are interested in the use of MTL in NLP applications, particularly for detecting cyber-aggression in text — the trolling, aggression, and cyberbullying shared task.

Cyberbullying and cyber-aggression (Smith et al., 2008) have become a growing social problem. They have several definitions (Grigg, 2010; Smith et al., 2008; Tokunaga, 2010) due to their different forms and impacts on internet users. As opposed to traditional bullying and aggression, they can occur at any time, and they do not have restricted places; i.e. schools. Cyber-aggression basically describes the behavior of sending intentional offensive, derogatory, or harmful content (Grigg, 2010) to an individual or a group of people. Cyberbullying, as opposed to cyber-aggression, involves repetitions rather than intention. On the other hand, trolling has different definitions, some of which describe the behavior of disrupting or provoking people by posing inflammatory, malicious, or off-topic comments (Mihaylov & Nakov, 2015).

The trolling, aggression, and cyberbullying shared task (Bhatia & Maheshwari, 2018; Kumar, Ojha, Malmasi, & Zampieri, 2018) was used to determine the severity of cyber-aggression (Table 1) of Twitter and Facebook posts published by users. This task is challenging as it is new, and there is no prior data available for comparison. The leaderboard results have been anonymized during the compaction. The development sets are relatively small compared to the testing set. Thus, the models must be able to generalize, specially that the testing sets have a larger size as compared to the training sets.

In this paper, our contributions can be summarized as below.

- **Cross segment-and-concatenate multi-task learning (CSC-MTL):** we propose a novel approach that enables robust shared feature representation of multi-tasks by highlighting contrasts among polarity and neutral classes. We report the performance of our approach on binary and categorical problems and show that it leads to improved performance for both.
- **Multi-Convolutional Neural Network with Pooling (MultiCNNPooling) model:** we develop and evaluate the performance of our proposed method with MultiCNNPooling model for cyber-aggression intensity ordinal classification.

Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying, pages 159–165
Santa Fe, USA, August 25, 2018.

Table 1: Number of text messages in cyber-aggression shared task 2018.

Category	Train	Dev	Test		Total
			Facebook	Twitter	
Covertly aggressive	4240	1057	142	413	5852
Overtly aggressive	2708	711	144	361	3924
Non-aggressive	5051	1233	630	483	7397

2 Data Preprocessing

We use a social media processing tool to prepare the text and provide a fast and reliable tokenization. It helps process social media posts such as emoticons, emojis, hash tags, and user mentions, as well as tokenization and sentence encoding. This involves the steps below:

- The NGram tokenizer is used with a supplied vocabulary. Sentences are tokenized into tokens, which are used afterwards to encode text as a sequence of indices to be fed to the network.
- Accents and non-Latin characters are cleaned from text.
- Emoticons and emojis are identified and then replaced with meaningful text. For instance, replace :(by "sad".
- Hashtags and URLs are recognized and then replaced with unique text; i.e., <hashtag_start>depressed<hashtag_end> instead of #depressed.
- User reference mentions are identified and replaced with person entities; i.e., <person>.

3 Cross Segment-and-Concatenate MTL (CSC-MTL)

Multi-task learning aims to exploit the shared representations across multiple classification tasks, in order to improve learning efficiency and prediction accuracy. The main underlying mechanism of MTL promotes task regularization over model overfitting regularization, which helps to penalize all complexity systematically.

There are two basic ways to share hidden layers within MTL in deep learning, 1) we train each task independently and then freeze all models before starting joint-learning training (Caruana, 1993). Freezing prevents shared layers from being modified, which helps alleviate model overfitting, and 2) we use a regularized loss function such as the l_2 -norm loss function (Ng, 2004) to constrain shared layers. This enables simultaneous training of multiple tasks, and helps avoid performance degradation. For instance, the overall cost function is $c = lc_1 + (1 - l)c_2$ where l denotes the learning rate, and c_1 and c_2 refer to the simultaneous tasks being trained.

Our novel approach uses a cross segment-and-concatenate (CSC) layer to enable multitask learning. This layer finds the abstract representation of tasks by identifying and segmenting polarity classes on each. Sentiment analysis tasks provide good examples. They include positive, neutral, and negative classes; e.g. positive and negative sentiments are the polarity classes. CSC is used as a shared weight layer that can be easily integrated with multiple network models. CSC-MTL helps enforce constraints to regularize learning and solve problems such as class imbalance.

We demonstrate the effectiveness of our approach used on the cyber-aggression detection from text shared task.

3.1 Cross Segment-and-Concatenate (CSC) Layer

Consider an example (Figure 1) of multiple tasks $X = \{x_1, x_2\}$, which report on different classification problems. They are fed the same input of a sequence (S) of tokens, such as words. The first step is to segment each task using class binarization in a one-vs-all (OVA) manner and then manually identify the polarity and neutrality of the tasks' classes.

For an n-class problem, neutrality is defined as the negative classes, which in the case of a binary class problem, neutrality is defined as concatenation of $p_{x_1}||p_{x_2}$ representations, where p denotes the shared weight feature of a polarity class, and the operator || refers to the concentration. After that, we train each binarized class independently. Second, we provide a cross-concatenation layer to supervise and constrain the shared representation among tasks in a way that signifies the distinctive features between each polarity group and their neutral classes (Figure 1).

For two tasks x_1 and x_2, we define polarity $\{p_{x_1}, p_{x_2}\}$ and neutrality$\{n_{x_1}, n_{x_2}\}$ classes. Based on which, we learn the linear combinations $\{l_1 = p_{x_1}||p_{x_2}p_{x_1}, l_2 = p_{x_2}||p_{x_1}p_{x_2}\}$ of polarity classes. Then, we parameterize $\{l_1, l_2\}$ and concatenate them into neutrality classes $\{l_1 \,||\, n_{x_1}, l_2|| \, n_{x_2}\}$ in order to learn the polarity shared representation as compared to neutrality classes. l_{CSC} refers to the output of CSC, which is used to feed the next layers.

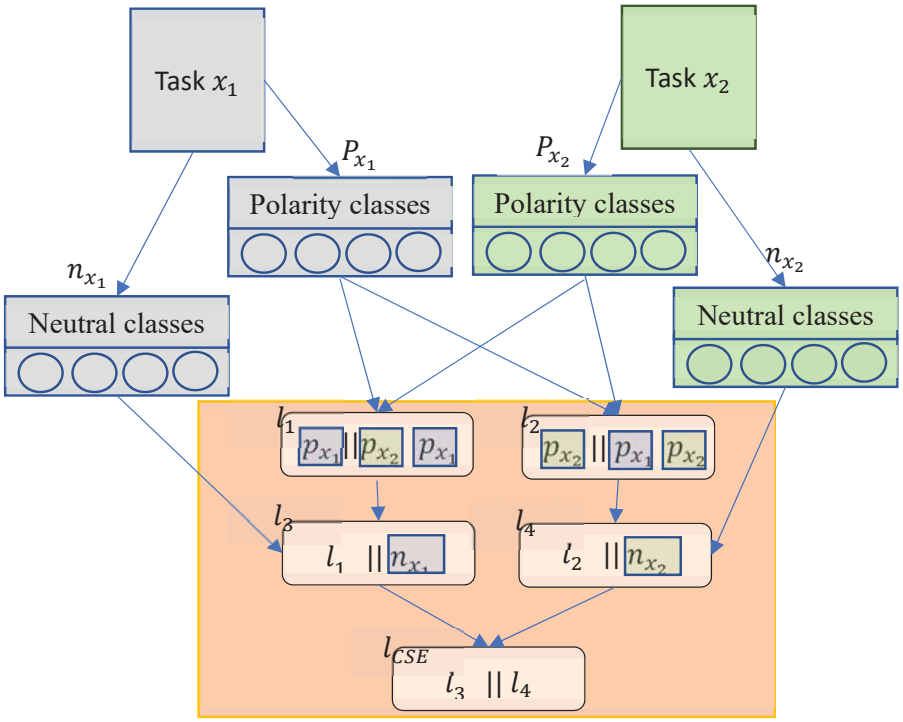


Figure 1: Shared representation learnt by linear combinations of tasks x_1 and x_2.

3.2 CSC Design

We evaluate our novel approach, CSC-MTL, using MultiCNNPooling model (explained in the next section). We perform, 1) *emotion analysis:* a categorical problem of analyzing emotions from text, so it will be binarized into 9 tasks (anger, fear, joy, sadness, surprise, anticipation, trust, disgust, and neutral), and 2) *cyber-aggression detection*: a problem involving two tasks, cyber-aggression and emotion analysis, which are used to detect aggression from text.

Emotion analysis: a categorical problem to detect emotion in text. We represent emotions using Plutchik's wheel of emotions (Plutchik, 2001), which provides emotional granularity by defining 8 primary bipolar emotions. Using this construct, we define our emotional polarity classes as the set of pairs {joy, sadness}, {anger, fear}, {trust, disgust}, and {surprise, anticipation}. We represent the neutral class as the singleton set {*neutral*} (no emotion).

Cyber-aggression detection: a categorical problem to detect aggression for given text. Polarity classes are defined as the set containing *covertly, overtly* and the *emotion CSC* inputs, while neutral classes contain one class, *non-aggressive*.

3.3 Model Description

A network input is a sequence s of tokens—such as words—where $S = [s_1, s_2, \ldots., s_t]$ and t denotes the timestep. S_i is a one-hot encoding of input tokens that have a fixed length (T), such that a sequence that exceeds this length is truncated in the pre-mode. For instance, a sentence such as "this is an example", which has a fixed length, thirteen, will be truncated to "is an example".

Word encoding: A word dictionary of fixed terms W is used to encode a sequence. It contains three constants that determine the start and end of the sequence in addition to the out of vocabulary (OOV) words. We normalize the variable text length using padding for short sequences and truncation for long sequences.
Word embedding: It is used to project words into a low-dimensional vector representation x_i, where $x_i \in R^W$ and W is the word weight embedding matrix. We use GloVe (Pennington, Socher, & Manning, 2014) pre-trained word embeddings. GloVe is pre-trained over social media posts, such as Twitter. For pre-trained embedding, we have an additional hyperparameter that is used to either freeze its weight matrix or allow for further training. We used cyber-aggression, and emotion corpora (Section 4.1) to build word embeddings of dimension 200; the resulting dictionary size is 60237 words.
Convolutional Neural Network (CNN): a convolution operation applies a sliding w-gram operation on a given input sequence $\{e_1, e_2, .., e_t\}$ with a length d. It results in a concatenated embedding vector $x_{i-f+1}, \ldots, x_i$of dimension of a filter length f. Thus, it generates $p_i \in \Re^d$ using weights $W \in \Re^{d \times wd}$ for a bias $b \in \Re^d$ and $p_i = \tanh(W_{x_i+b})$

3.4 Models

We describe our CNN-based neural network model (Figure 2) for single task training, which are used to evaluate the performance of our CSC-MTL approach. We build this model on top of the word-embeddings described in the previous section. The word embedding layer is followed by a dropout. Each model is followed by a vanilla layer that is fully-connected, has 200 hidden units, and uses a Rectified Linear Unit (ReLU) activation. Then, we apply a final dropout. The output layer is a fully-connected layer with one hidden unit, and it uses a sigmoid activation to produce an output. Each CSC input is followed by a gaussian noise layer of the value 0.3.
MultiCNNPooling (Figure 2): consists of 3 convolutions. Each convolution has 64 features, as well as filters of the lengths 3, 4, and 5. After that, we apply a max-pooling operation to extract the abstract information $\widehat{w_i^f} = \max(C_i^f)$. Finally, feature representations are concatenated into a single representation. Convolutional operations is helpful with max-pooling pooling to extract word features (Kalchbrenner, Grefenstette, & Blunsom, 2014).

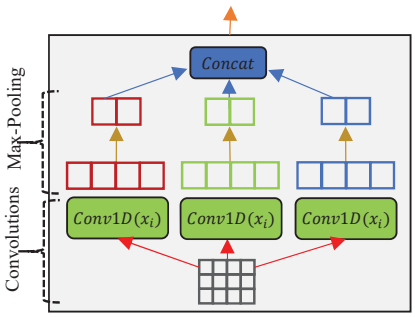


Figure 2: MultiCNNPooling network model.

3.5 Training

We use mini-batch gradient descent with a batch of the size 32 with back-propagation to reduce the loss error between the actual and predicted classes. For model training, Adam optimizer (Kingma & Ba, 2014) is used with a learning rate of 0.01. We also clip the gradient norm (Pascanu, Mikolov, & Bengio, 2012) at 7 to alleviate the risk of exploding gradient, in particular with recurrent model training.
Regularization: We apply dropout on neurons and recurrent connections to protect them from co-adaptation (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). We additionally perform weight decay regularization using L2 penalty (Cortes, Mohri, & Rostamizadeh, 2012). We set the gaussian noise level to 0.3 for each CSC input.
Hyperparameters: We use an embedding layer of a dimension size 200. We set a dropout of 0.2. Finally, we use L2 regularization of 0.0002 at the loss function.

4 Datasets

We use two datasets to perform the cyber-aggression shared task to validate our CSC-MTL approach with MultiCNNPooling structure on detecting aggression from text.

4.1 Emotion Analysis Dataset

For emotion analysis from text, we prepared a dataset from three datasets, CrowdFlower text emotion[1], blogs (Ghazi, Inkpen, & Szpakowicz, 2015), and tweets (Buechel & Hahn, 2017). The total number of instances of our dataset is 67,091. The final dataset consists of 8,638 neutral, 16,252 joy, 10,290 sadness, 5,219 anger, 11,971 fear, 4,601 trust, 2,398 disgust, 6,196 surprise, and 1,526 anticipation instances.

4.2 Cyber-aggression Shared Task Dataset

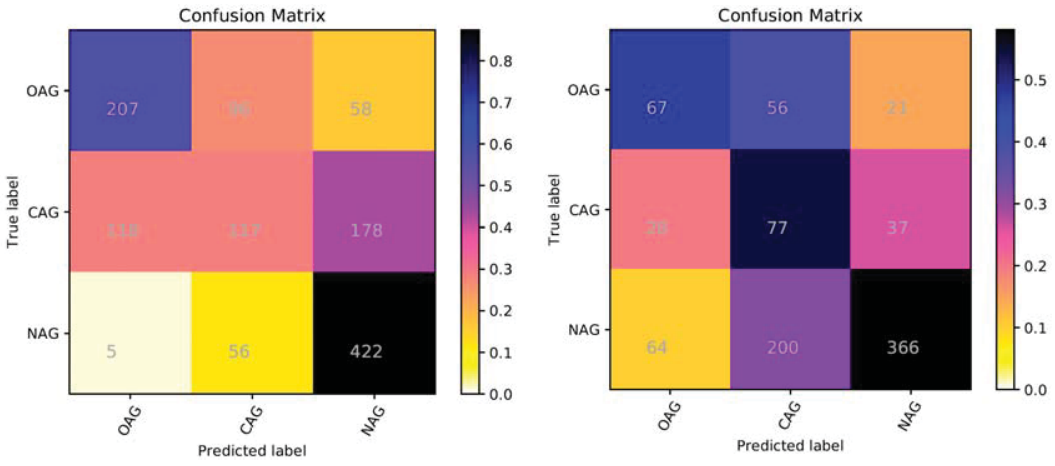
The trolling, aggression, and cyberbullying shared task consists of 17,173 interchanged messages. These messages are categorized intro three different classes, covertly aggressive, overtly aggressive, and non-aggressive, which are labeled 5852, 3924, 7397 times respectively. Facebook and Twitter are used as unseen datasets to test the generalization ability of our approach.

5 Evaluation and Results

Our model ranks 5/30 and 8/30 on the Twitter and the Facebook test set, respectively. We evaluate the performance of a categorical problem, cyber-aggression (Section 4.2) from text, using our CSC-MTL model. We perform a stratified split on the cyber-aggression training set, so that the development set is used as a held-out dataset. The split ratio is 80%, and it is used for training, while 20% is used for the validation set. Then, we use these trained models to evaluate on the unseen Facebook and Twitter test sets.

Dataset		Accuracy	F1 (weighted)
Twitter	Baseline	-	34.77%
	Our model	**59.35%**	**56.9%**
Facebook	Baseline	-	35.35%
	Our model	**55.68%**	**59.74%**

Table 2: Results for English Facebook and Twitter test sets



(a) Twitter (b) Facebook

Figure 3: Confusion matrix on the Twitter and on the Facebook test set.

[1] https://www.crowdflower.com/wp-content/uploads/2016/07/text_emotion.csv

Table 1 reports good standing results of our model for the generalization ability test. This indicates that regularization and hyperparameter tuning managed to control the overfitting issue. The MultiCNN model with static pretrained GloVe embedding reported a higher F1 as compared to the baseline. Figure 3 shows the confusion matrix for both tasks. The generalization ability test (Table 2) shows that our model reports competitive performance with the Facebook set (59.74%) over the Twitter set (56.9%).

6 Conclusion

In this paper, we presented a novel approach, named cross segment-and-concatenate multi-task learning (CSC-MTL). We used our approach for cyber-aggression detection from text. Our experiment showed that our CNN-based model with CSC settings had promising results for cyber-aggression detection. MultiCNN with our CSC-MTL reported a competitive F1 score for both the Twitter and Facebook test sets.

In future work, we will further test our approach on different NLP tasks. We will need to perform a systematic evaluation of our method with different hyper-parameters, as well as to test on common neural network structures such as recurrent neural network (RNN).

Acknowledgements

This research is funded by Natural Sciences and Engineering Research Council of Canada (NSERC), Ontario Centres of Excellence (OCE) and SafeToNet.

References

Jonathan Baxter. 1997. A Bayesian/Information Theoretic Model of Learning to Learn via Multiple Task Sampling. *Machine Learning*, *28*(1), 7–39. https://doi.org/10.1023/A:1007327622663

Sven Buechel, Udo Hahn. 2017. EmoBank: Studying the Impact of Annotation Perspective and Representation Format on Dimensional Emotion Analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers* (pp. 578–585). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.18653/v1/E17-2092

Richard A. Caruana. 1993. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Machine Learning Proceedings 1993* (pp. 41–48). Elsevier. https://doi.org/10.1016/B978-1-55860-307-3.50012-5

Corinna Cortes, Mehryar Mohri, Afshin Rostamizadeh. 2012. L2 Regularization for Learning Kernels. Retrieved from http://arxiv.org/abs/1205.2653

Diman Ghazi, Diana Inkpen, Stan Szpakowicz. 2015. Detecting Emotion Stimuli in Emotion-Bearing Sentences. In *Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing 2015* (pp. 152–165). https://doi.org/10.1007/978-3-319-18117-2_12

Nal Kalchbrenner, Edward Grefenstette, Phil Blunsom. 2014. A Convolutional Neural Network for Modelling Sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 655–665). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/v1/P14-1062

Diederik P. Kingma, Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. Retrieved from http://arxiv.org/abs/1412.6980

Ritesh Kumar Aishwarya N Reganti Akshit Bhatia, Tushar Maheshwari. 2018. Aggression-annotated Corpus of Hindi-English Code-mixed Data. In N. C. (Conference chair), K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, … T. Tokunaga (Eds.), *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Paris, France: European Language Resources Association (ELRA).

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, Marcos Zampieri. 2018. Benchmarking Aggression Identification in Social Media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbulling (TRAC)*.

Todor Mihaylov, Preslav Nakov. 2015. Hunting for Troll Comments in News Community. In *The 19th Conference on Computational Natural Language Learning. Proceedings of the Conference. Beijing, China*

(pp. 310–314).

Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, Zhi Jin. 2016. How Transferable are Neural Networks in NLP Applications? Retrieved from https://arxiv.org/abs/1603.06111

Andrew Y. Ng. 2004. Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In *Twenty-first international conference on Machine learning - ICML '04* (p. 78). New York, New York, USA: ACM Press. https://doi.org/10.1145/1015330.1015435

Razvan Pascanu, Tomas Mikolov, Yoshua Bengio. 2012. On the difficulty of training Recurrent Neural Networks. Retrieved from http://arxiv.org/abs/1211.5063

Jeffrey Pennington, Richard Socher, Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162

Robert Plutchik. 2001. The nature of emotions: Human emotions have deep evolutionary roots. *American Scientist*, *89*(4), 344–350. https://doi.org/10.1511/2001.4.344

Peter K. Smith, Jess Mahdavi, Manuel Carvalho, Sonja Fisher, Shanette Russell, Neil Tippett. 2008. Cyberbullying: its nature and impact in secondary school pupils. *Journal of Child Psychology and Psychiatry*, *49*(4), 376–385. https://doi.org/10.1111/j.1469-7610.2007.01846.x

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Robert S. Tokunaga. 2010. Following you home from school: A critical review and synthesis of research on cyberbullying victimization. *Computers in Human Behavior*, *26*(3), 277–287. https://doi.org/10.1016/j.chb.2009.11.014

Dorothy Wunmi Grigg. 2010. Cyber-Aggression: Definition and Concept of Cyberbullying. *Australian Journal of Guidance and Counselling*, *20*(02), 143–156. https://doi.org/10.1375/ajgc.20.2.143