Aggressiveness Detection through Deep Learning Approaches.

Victor Nina-Alcocer, José-Ángel González, Lluís-F. Hurtado, and Ferran Pla

VRAIN: Valencian Research Institute for Artificial Intelligence.
Universitat Politècnica de València
Camí de Vera s/n
46022 València, Spain
vicnial@inf.upv.es
{jogonba2,lhurtado,fpla}@dsic.upv.es

Abstract. This paper presents a description of our participation in task "MEX-A3T: Authorship and aggressiveness analysis in Twitter: case study in Mexican Spanish" at Iberian Languages Evaluation Forum (IberLEF) 2019. This work is focused on studying the detection of aggressiveness on Spanish tweets. For a finer-grained study of the task, we analyzed three different approaches: the first two approaches consider the design of architectures using convolutional and recurrent neural networks. Meanwhile, the third approach is focused on pay attention to certain important parts of a sentence using self-attention networks.

Keywords: Twitter · Aggressiveness · Convolutional Neural Networks · Recurrent Neural Networks · Self-Attention Networks.

1 Introduction

The shared task "MEX-A3T: Authorship and aggressiveness analysis in Twitter: case study in Mexican Spanish"[3] considers that aggressive comments on social networks can represent a certain type of threat or risk for users. Therefore, they proposed a main subtask to tackle this social issue: *the detection of aggressive comments on Mexican Spanish tweets*. This subtask is focused in determining whether a tweet is aggressive or not.

Considering the systems proposed in the last edition of MEX-A3T(2018)[1], we saw that the classical machine-learning algorithms [4] reached good results. Nevertheless, the main goal of this paper is to explore deep learning approaches using convolutional [6], recurrent [8] and self-attention networks [11] (CNNs, RNNs, and SANs) to see whether these paradigms can contribute in some way to reach competitive results.

This paper is organized into four sections: the first one presents an introduction to this study. The second section provides a description of the proposed systems. The third one explains the experiments conducted and the results achieved by our models and comments the official results. And the last section shows some conclusions.

2 Systems Description

As it is known, CNNs allow us to process a sentence using different n-grams at a time to recognize some patterns or information that can be useful for text classification. Also, LSTM and GRU [8] are mostly used for capturing long-term dependencies, learning the dependencies in lengths of considerably large text. Moreover, some research studies demonstrated that *attention mechanism* approach achieved good results for machine translation problem [11]. In our case, we will focus on discovering whether self-attention networks behave satisfactorily for the text classification task, knowing that it does not have recurrence or convolutional components.

This paper considers three main approaches: the first one takes into account a bunch of architectures which use CNNs as the base layer to handle n-grams. Basically, this architecture is composed of an input embedding layer, followed by an spatial dropout and a convolutional layer with a set up of 256 filters and kernel size of 2, 3 or 4. Next, a GlobalMaxPooling1D is defined (or a Maxpooling to add a LSTM or GRU layers) and finally, a dense layer with softmax is used to generate the results.

The second approach takes into account a stack of different layers: firstly, three CNNs are defined, the first CNN considers bigrams, the second one considers trigrams, and the last one considers quadrigrams. All these CNNs are flattened for feeding a dense layer (of 256 nodes). Finally a last dense layer with softmax is used to obtain the predictions.

The third approach considers the use of self-attention networks [2]. We used a similar Transformer architecture proposed by [11] for the machine translation problem. Nevertheless, we just kept part of this architecture because we are interested only in the first stage which pays close attention to some *important* parts of the input (tweet or sentence) [7]. Basically, this architecture has embeddings as inputs, next a SpatialDropout1D is applied and posteriorly attention is considered over the input (sentence-level). Then the encoder is followed by a GlobalMaxPooling1D and finally, a layer normalization and a dense layer with softmax was set up.

3 Experiments and results

In this section all the experiments carried out in this task are commented as well as the results obtained.

For Aggressive detection, the organizers of MEX-A3T provided a training set of 7700 tweets written in Spanish. (labeled with two possible classes: 0 =

“non-aggressive“(**NOA**), 1 = “aggressive“(**AGG**)). Regarding the distribution of the classes we can observe that these are not balanced (the NOA class (65%) and the AGG class (35%)). In order to evaluate the performance of the proposed systems, 3156 unlabeled tweets were provided as a test set.

We applied the following text-preprocessing to all the tweets that are the input to our systems: we remove all URLs, numbers, users, times, dates, emails, percents [5]. However, we normalize hashtags (i.e., “#ChileSinMundial“ is converted to “chile sin mundial“) [9] and emojis with its Unicode Common Locale Data Repository (CLDR) version, elongated words, repetitions, emphasis, and censored words. All the proposed systems in this paper used the text-preprocessing already mentioned and two different word embeddings: Twitter87 embeddings trained over 87 millions of tweets and MexE embeddings provided by the organizers [10].

Furthermore, 5-fold cross-validation sets were used to avoid over-fitting in all the experiments. The official evaluation metric to evaluate the systems was F1_score for aggressiveness (f1_aggr). However, we show in Table 1 and Table 2 some additional metrics, accuracy (acc.) and macro_f1 to have a better interpretation of the results.

Table 1. Architectures applied to the training dataset.

	Twitter87			**MexE**			
System	**f1_aggr**	**macro_f1**	**acc.**	**f1_aggr**	**macro_f1**	**acc.**	
1. cnn_b	**0.7401**	**0.8025**	**0.8223**	0.7259	0.7939	0.817	(Run2)
2. cnn_t	0.7239	0.7941	0.8187	0.7269	0.7932	0.8148	
3. cnn_q	0.7275	0.7953	0.8181	0.6895	0.7726	0.8035	
4. cnn_b_gru	0.6535	0.7385	0.7665	0.7397	0.7813	0.8057	
5. cnn_t_gru	0.6851	0.7475	0.7632	0.7278	0.7854	0.801	
6. cnn_q_gru	0.6545	0.7362	0.7655	0.724	0.7812	0.7965	
7. cnn_b_lstm	0.6672	0.7446	0.7688	0.6665	0.7432	0.7662	
8. cnn_t_lstm	0.6863	0.7482	0.7644	0.6741	0.7515	0.7761	
9. cnn_q_lstm	0.6957	0.7595	0.7768	0.6924	0.7552	0.7717	
10. stacked	0.7116	0.7815	0.8042	**0.7399**	**0.7935**	**0.8158**	(Run3)
11. attention	**0.8411**	**0.8733**	**0.8803**	**0.7611**	**0.8039**	**0.8132**	(Run1)

Table 1 shows the results obtained by all the systems in the tunning phase. As we commented in Section 2, the first approach allowed us to consider many different architectures using a CNN as base system. In the architectures named cnn_b, ccn_t, cnn_q we considered a kernel size of 2, 3 or 4 to process bigrams, trigrams and quadrigrams respectively. In the cases of cnn_b_lstm, cnn_t_lstm, cnn_q_lstm, cnn_b_gru, cnn_t_gru and cnn_q_gru we added LSTM or GRU layer for capturing long-term dependencies. After carrying out several experiments with this first approach, we realized that considering bigrams on CNNs (**cnn_b**) and the Twitter87 embeddings, the system achieved competitive results compared to other architectures that follow this first focus (see Table 1, rows 1 to 9).

As we mentioned in Section 2. The second approach takes into account a variety of layers. the main idea behind this approach is to vary the number of inputs with its respective CNNs. Also, we want to vary the number of nodes of the next dense layer which process the outputs of the previous CNNs. Firstly, we used only a CNN to process bigrams, then we added a dense layer of 128 nodes plus another dense layer with two nodes with softmax. The next tested architecture kept the same structure, however, we added another input and its CNN to process trigrams. And for the last tested architecture, we added another CNN to process the quadrigrams. Summing up, the system called **stacked** (row 10 in Table 1) which reached good results use MexE embeddings and it is structured by three inputs and its respective CNNs, a dense layer of 256 nodes and a last dense layer of two nodes.

To work and make experiments with the third approach based on SANs, firstly, we defined the architecture commented on Section 2. After defining the system, we set up some parameters to see the importance and impact that those have over the performance reached by the system. For instance, one interesting fact was the definition of the Dropout in the inputs to sentence-level on the encoder, we tried on a range of 0.2 to 0.8, and we noticed that a dropout of 0.7 achieved the best performance in this particular case, meaning that hiding a huge part of the sentence helps a lot to reach good results. Another important consideration, it was taken into account the unbalanced training dataset, to face this fact we set up the experiments to treat every instance of the class NOA as two instances of the class AGG (minority class), it means that we are assigning higher values to AGG instances for having a balanced training dataset. Therefore, having defined our system named **attention** (see row 11 in Table 1) and the parameters tuned, all this combined with the Twitter87 embeddings allow us to reach the highest results for f1_agg.

3.1 Official Results

Table 2 shows the official results published by the organizers of MEX-A3T-2019. For the aggressiveness detection subtask, we submitted two runs, both submissions obtained worst results than the baselines proposed by the organizers. The Run1 (**attention**) was the submission which reached the best result, this system uses SANs as explained on Section 2. Meanwhile, Run2 (**cnn_b**) achieved low results, all of them below bag-of-words (BoW) [12] baseline. Additionally, we can highlight Run3 (**stacked**) because it was the only architecture that achieved good results with (MexE) the embeddings provided by the organizers. Unfortunately, this Run3 did not achieve good results as Run1 or Run2. Moreover, it is worth mentioning that on MEX-A3T(2018) edition, some of the best results were reached by systems which used classic machine-learning algorithms. We assume that the two baselines proposed on the official ranking use some of these classic algorithms and trigrams or bag-of-words as features.

Some interesting fact that we noticed was the performance of our third approach based on SANs, this system worked really good on the training dataset. But it did not reach good results on the test dataset. We assume that this issue is due to the fact that the content of the test data is too different from the training set. It means that our system has trained and learned some patterns that would not be founded on the test data.

Table 2. Official results of Aggressiveness Detection.

Team	**f1_aggr**	**macro_f1**	**acc.**
Baseline (Trigrams)	0,4300	0,6080	0,6688
Our_System_Run1	0,4081	0,4897	0,5029
Baseline (BoW)	0,3690	0,5760	0,6777
Our_System_Run2	0,2921	0,5122	0,6115

4 Conclusions

In this work we have presented our participation in the MEX-A3T-2019 shared task focused on aggressiveness detection. We proposed three approaches: the first two approaches consider the use of convolutional and recurrent neural networks to compute n-grams and long-term dependencies in tweets. The third approach considers the use of self-attention networks to pay close attention to some words of the tweet. Based on the experiments and the results obtained in this paper, we noticed that the architectures proposed on the first two approaches reached slightly similar results. But making a comparison between the first two approaches and the last approach, we see good results reached by self-attention networks. Therefore, we have observed that a deeper study on SANs can help us to improve our results.

Acknowledgments

This work has been partially supported by the Spanish MINECO and FEDER founds under project AMIC (TIN2017-85854-C4-2-R) . Work of José-Ángel González is also financed by Universitat Politècnica de València under grant PAID-01-17.

References

1. Álvarez-Carmona, M., Guzmán-Falcón, E., Montes-y Gómez, M., Escalante, H.J., Villaseñor-Pineda, L., Reyes-Meza, V., Rico-Sulayes, A.: Overview of MEX-A3T at IberEval 2018: Authorship and aggressiveness analysis in Mexican Spanish tweets. In: CEUR Workshop Proceedings. vol. 2150, pp. 74–96 (2018), https://pan.webis.de/clef18/pan18-web/index.html

2. Ambartsoumian, A., Popowich, F.: Self-attention: A better building block for sentiment analysis neural network classifiers. CoRR **abs/1812.07860** (2018), http://arxiv.org/abs/1812.07860
3. Aragón, M.E., Álvarez-Carmona, M.Á., Montes-y Gómez, M., Escalante, H.J., Villaseñor-Pineda, L., Moctezuma, D.: Overview of mex-a3t at iberlef 2019: Authorship and aggressiveness analysis in mexican spanish tweets. In: Notebook Papers of 1st SEPLN Workshop on Iberian Languages Evaluation Forum (IberLEF), Bilbao, Spain, September (2019)
4. Dey, A.: Machine Learning Algorithms: A Review. Tech. rep., www.ijcsit.com
5. Ibrohim, M.O., Budi, I.: A Dataset and Preliminaries Study for Abusive Language Detection in Indonesian Social Media. Procedia Computer Science **135**, 222–229 (2018). https://doi.org/10.1016/j.procs.2018.08.169, https://linkinghub.elsevier.com/retrieve/pii/S1877050918314583
6. Jacovi, A., Shalom, O.S., Goldberg, Y.: Understanding convolutional neural networks for text classification. CoRR **abs/1809.08037** (2018), http://arxiv.org/abs/1809.08037
7. Letarte, G., Paradis, F., Gigù, P., Laviolette, F.: Importance of Self-Attention for Sentiment Analysis. Tech. rep. (2018), https://www.aclweb.org/anthology/W18-5429
8. Lu, Y., Salem, F.M.: Simplified gating in long short-term memory (LSTM) recurrent neural networks. CoRR **abs/1701.03441** (2017), http://arxiv.org/abs/1701.03441
9. Mathur, P., Ratn Shah, R., Sawhney, R., Mahata, D.: Detecting Offensive Tweets in Hindi-English Code-Switched Language. Tech. rep. (2018)
10. MEX-A3T: MEX-A3T: Authorship and aggressiveness analysis in Twitter case study in Mexican Spanish 2019 (2019), https://sites.google.com/view/mex-a3t/home
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need (2017), https://arxiv.org/pdf/1706.03762.pdf http://arxiv.org/abs/1706.03762
12. Zhang, Y., Jin, R., Zhou, Z.H.: Understanding bag-of-words model: A statistical framework. International Journal of Machine Learning and Cybernetics **1**(1-4), 43–52 (dec 2010). https://doi.org/10.1007/s13042-010-0001-0, http://link.springer.com/10.1007/s13042-010-0001-0