Transformers and Data Augmentation for Aggressiveness Detection in Mexican Spanish

Mario Guzman-Silverio[a], Ángel Balderas-Paredes[a] and Adrián Pastor López-Monroy[a]

[a]*Mathematics Research Center (CIMAT), Jalisco S/N Valenciana, 36023 Guanajuato, GTO México.*

Abstract

In this paper we describe the system designed by the Mathematics Research Center (CIMAT) for participating at *MEX-A3T 2020*. In this work, we addressed the Aggressiveness Detection (AD) task by exploiting Bidirectional Encoder Representations from Transformers (BERT) and Data Augmentation. BERT fine-tuning has shown outstanding performance in a wide range of language tasks. However, according to recent research fine-tuning BERT on small size datasets (<10K instances) often results in unstable models. In other words, even when only the final layer is randomly initialized, distinct random seeds lead to substantially different results. In this paper, we use two strategies that are helpful in producing more stable classification models based on fine-tuned BERTs. The first strategy take advantage of ensembles, whereas a second strategy relies in data augmentation. The experimental evaluation showed that our proposals outperforms all baselines by a wide margin, and has the overall first place for Aggressiveness Detection in Mexican Spanish Tweets.

Keywords

Agressiveness Detection, Transformers, Data Augmentation, Text Classification

1. Introduction

Nowadays Internet users can easily share information in a number of social platforms. In this regard, the automatic analysis of textual information has been a popular research topic for the scientific community. This is especially true in applications that could prevent risks. Recently, some academic competitions have emerged as forums where researches evaluate their approaches for specific tasks by analyzing the way language is used by people, for example, the case of offensive language [1].

The MEX-A3T 2020 [2] forum tackles two tracks focused on digital text forensics: Fake News (FN), and Aggressiveness Detection (AD) in Mexican Spanish. The interest of this paper is only on AD, that aims to determine which tweets attempt to insult, offend, attack, or hurt other people. In this regard, AD could prevent damages and harmful scenarios like cyberbullying [3]. The models and strategies we propose here are suitable to represent and augment such aggressive tweets by using Transformers and Data Augmentation. For the FN track due to the

Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020)

EMAIL: mario.guzman@cimat.mx (M. Guzman-Silverio); angel.balderas@cimat.mx (Á. Balderas-Paredes); pastor.lopez@cimat.mx (A.P. López-Monroy)
URL: https://www.cimat.mx/es/adrián-pastor-lópez-monroy (A.P. López-Monroy)
ORCID: 0000-0003-1018-4221 (A.P. López-Monroy)

CEUR Workshop Proceedings (CEUR-WS.org)

different structure in documents, our approach was a **simple baseline**, only for our reference for future research: a Bag-of-*N*-grams at word and character level fed into an Support Vector Machines (SVM).

The AD task is commonly approached as a supervised classification problem [4, 5, 1]. The problem has been approached by using a number of strategies, inlcuding: regression models [6], user network-attributes [7], or distributional terms representations [8]. Recently, with the rise of deep learning neural models, some authors have been using Recurrent Neuronal Networks [9], and Convolutional Neural Networks [10].

One of the most successful approaches is the Bidirectional Encoder Representations from Transformers (BERT), which was first proposed in [11]. BERT has shown outstanding performance in a wide range of Natural Language Processing (NLP) tasks. For the case of offensive language detection and hate speech detection, BERT has been successfully used by some authors in different ways [12, 13]. For using BERT in general text classification the most common strategy to take advantage of fine tuning. For example, BERT can be pre-trained in general domains (e.g., Wikipedia) to model syntactic and semantic properties of language that are useful in other tasks. In simple words, pre-trained BERT models are fine-tuned to specific domains by substituting the output layers of the model, and re-training the rest of layers at specific pace. Notwithstanding the effectiveness of BERT in text classification, several works have pointed out its instability when fine-tuning BERT on small size datasets [14, 15, 16]. In simple words, even when only the final layer is randomly initialized, distinct random seeds lead to substantially different results. In this regard, considering that the AD corpus have less than 10K samples, we propose to build a classification strategy based on combining several BERT models trained with different seeds on different augmented datasets. By doing this we aim to get a model that have in average a solid performance but small variance. According to our evaluation, the use of ensemble methods with specific voting schemes and adversarial data augmentation can improve the effectiveness of BERT while maintaining lower variance in performance for the small and unbalance dataset for AD.

The remainder of this document is organized as follows: Section 2 presents the proposed strategies. Section 3 describes experimental settings. The experiments and results are presented in Section 4. Finally, Section 5 outlines the final conclusions and future work.

2. Stable Classification Strategies

This section describes our two proposed strategies to alleviate the instability of fine-tuning BERT on few sample and unbalanced datasets. These strategies are built on fine-tuned BERT models that only vary the random seed of an extra classification layer for domain adaptation. The first strategy is based on BERT ensembles and two well known voting schemes. The second strategy uses data augmentation to improve even more the effectiveness and stability. In our evaluation, we empirically show that both methodologies provide benefits for the AD and both ranked first place of the challenge.

2.1. BERT Ensemble model

There are many ways to combine the information of several models, but we are particularly interested in those based in ensemble theory. The idea of having an ensemble is that several models (possibly weak models) can make a strong one [17]. One of the key ideas in successfully build ensembles is that the prediction space should be diverse [18]. This is that individual models can have differences among decisions; we hypothesize those are the unstable individual BERTs. In general, when the latter conditions are met, it could be possible to obtain a stronger model with simple strategies. In our case we consider the following two straightforward voting schemes:

- **Majority Voting Scheme:** we predict the most voted class among the classifiers of the ensemble. In case of tie, we perform random prediction among the classes in question.
- **Weighted Voting Scheme:** we aggregate the confidence prediction for classes in each model of the ensemble to build a final weighted vote. This confidence prediction, in our case is the output of the last Softmax layer.

These strategies based on ensembles and simple voting schemes have resulted very convenient and have been explored with different base models in several domains [17, 9]. It is worth to mention that there other popular alternatives to combine. For example, the *Early Fusion* strategy consist in feeding a classifier by using the concatenation of weights in the penultimate layer of each model [18]. Other strategy could be an End-to-End network of the models. All those strategies are interesting, but definitely much more computationally expensive than using individual models and perform voting. For that reason we prefer these simple, yet effective, voting strategies over the others. In our experimental evaluation we will show their effectiveness to reduce the variance and improve the performance.

2.2. Data Augmentation

A common and effective technique in deep learning for image related tasks is that of data augmentation, in which the goal is to create a new training data by means of a transformation: sometimes simple like rotation, reflections and cropping, sometimes more complex techniques are used with better results [19].

Data augmentation for text based tasks is a very different scenario. The information in documents is sequential and the word, usually taken as the basic unit, has a syntactic and semantic meaning that depends on the context. Thus, changing the individual words or their order could result in noisy data that hurts the performance. This is specially true for approaches beyond the Bags-of-Words and inspired in language modeling like BERT where the order, context and structure of the text matters. Fortunately, there have been some advances demonstrating sightly improvements in some scenarios [20]. In this work, we have carefully adapted two methods, and proposed a new one to perform data augmentation. The explored data augmentation strategies are the following:

- **Easy Data Augmentation (EDA) [20]**: This is the most simple strategy and consists in generating new instances by modifying 20% of the original tweets. To this end, four basic operations are applied to randomly selected words in each tweet:

1. *Replace*: select a word and change it by a random synonym.
2. *Insert*: randomly choose a synonym of a word, and insert it randomly.
3. *Swap*: select two words and swap their positions.
4. *Delete*: remove a word from the sentence.

For Easy Data Augmentation, one extra tweet was created for each tweet.

- **Unsupervised Data Augmentation (UDA) [21]**: This implies the use of semisupervised learning; by augmenting each sentence of the original training set and using the kullback-leibler divergence to penalize the difference in the distributions of the logits. For Unsupervised Data Augmentation, four elements were created for each selected element from the input, EDA was used to create those new elements.

- **Adversarial Data Augmentation (ADA) [22]**: At each epoch of the training, an adversarial method is used to create a new input for the misclassified sentences. For adversarial data augmentation a implementation of TextFooler [22] for Spanish was used in which the purpose is to create a well classified input for a originally misclassified one.

It is worth to mention that previous described strategies, originally were designed for English language and therefore rely on dependent language tools. Thus, for each strategy we did several adaptations in order to exploit them in Spanish.

3. Datasets, Baselines and Experimental Settings

The MEX-A3T Team provided us the data set, which has 5222 no-aggressive and 2110 aggressive tweets. For the experiments in this paper we split the training data into 80% for training and 20% for test.

To compare the proposed strategies in this paper, we have trained two baselines that are commonly used in the literature:

- **N-grams ensemble with SVM**: We used a grid search to find a suitable number of unigrams, bigrams and trigrams at word level. We also find the number of 2-6-grams at character level. Those features were fed into an SVM to explore the C hyperparameter.
- **Bi-LSTM**: This baseline is a neural architecture with a Bi-LSTM and a classification layer. We use the pre-trained word2vec vectors in Spanish from Caro and Cuervo Institute - Linguistic Research Group [1]. We fix the learning rate in 1e-3, and we used the Adam Optimizer.

Regarding fine-tuning BERT, we set the hyper-parameters as the authors in [11] recommend. We use Adam with a learning rate of 1e-5 and a batch size of 32 for three epochs. We use a classification layer and as loss function the weighted Cross Entropy Loss by using the proportion of each class. In this process, we used a BETO pre-trained model in Spanish text [23] by using the widely known default implementation in [24].

[1] https:www.datos.gov.co

Table 1
Ensembles evaluation using F_1-Score for the aggressive class and standard deviation over one hundred runs. For the Ensembles rows we report two results: the left one for majority voting scheme, whereas the right one for weighted voting scheme.

Model	F_1-avg offensive	F_1-std offensive	F_1-avg non-offensive	F_1-avg macro
Ngrams-SVM	74.53	n/a	89.51	82.02
Bi-LSTM	73.00	n/a	87.33	80.17
Single BERT	79.31	0.836	91.88	85.60
Ensemble (5)	79.93|79.88	0.637|0.617	92.14|92.13	86.03|86.01
Ensemble (10)	79.68|80.01	0.541|0.516	92.17|92.19	85.93|86.10
Ensemble (20)	79.94|**80.17**	0.475|**0.446**	92.23|**92.26**	86.08|**86.22**

4. Evaluation

This section describes the experimental evaluation that shows the main findings of this work. Firstly in Section 4.1 we design a set of experiments to observe the benefits of building and ensemble of several BERT. Secondly in Section 4.2 we explore data augmentation strategies to improve even more their performance. In order to have comparable results in this section, we partitioned the trained dataset in the following way: 80% for training, and 20% for test. The metrics we use to compare the methods were the F_1 in each class, the macro F_1 and the standard deviation.

4.1. BERT Ensembles

The purpose of experiments in this section is to empirically show that BERT ensembles are helpful and more stable for classification. For the experiments in this section BERT was full fine-tuned (the 130 millions of parameters) two hundred times with different seeds that initialized the last layer to detect aggressiveness. This pool is then used to compute the averages over one hundred runs for each voting scheme. This means that, for *Single BERT* the average performance of one hundred individual BERTs is reported with the standard deviations. For the case of a *Ensemble* (n), n different BERTs were randomly taken for each of the one hundred runs; we report the average performance and the standard deviation.

In Table 1 we show the performance of BERT ensembles and other reference approaches. First of all, note that Single BERT clearly outperforms the two baseliness: 1) N-grams-SVM and 2) Bi-LSTM. Both of this approaches are very strong references, since we use well known strategies and heuristics to find suitable hyper-parameters (see Section 3). The margin of improvement of Single BERT over the two baselines also shows that on average, the fine tuning have been successfully done.

From Table 1 one can also note that Single BERT have a standard deviation of .836 in one hundred runs. However, when several BERTs are combined by using ensembles, the classification performance improves while the standard deviation decrease. In Table 1 each row *Ensemble* (n) (n-BERTs) has two values in each column metric. The left value was obtained by using a majority voting scheme, whereas the right value represents the performance when using weighted voting scheme. In Figure 1 the left plot shows the F_1 of the aggressive class as the

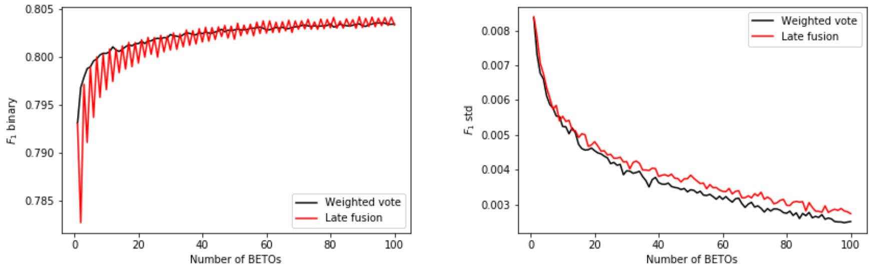

Figure 1: F_1 average by number of BERT in each kind of ensemble.

Table 2
Evaluation of trained models on validation data with F_1-score for the aggressive class and standard deviation over fifty runs.

Model	vanilla	eda	uda	adv
Single BERT	79.66 ± .67	79.29 ± .78	79.55 ± .57	79.66 ± .74
Ensemble (5)	80.47 ± .43	80.54 ± .48	80.21 ± .47	80.52 ± .50
Ensemble (10)	80.68 ± .34	80.73 ± .42	80.32 ± .38	80.73 ± .42
Ensemble (20)	**80.69** ± **.25**	**80.87** ± **.29**	**80.39** ± **.24**	**80.92** ± **.30**

number of BERTs in the ensemble increase up to one hundred. The red line is for majority vote, whereas the black line is for weighted vote. In a similar way, the right plot shows how the variance decrease as the size of the ensemble increase. The ensemble based on majority vote seems to have stable and higher performance until the number of BERTs is greater than sixty. Thus, the weighted voting scheme seems to be a better choice as the variance is consistently lower than the majority voting scheme. The *CIMAT-1* run reported by the organizers in the final ranking of the challenge is a simple *Ensemble* (20). In the following Section, we will show how data augmentation can be used to improve the performance.

4.2. Data Augmentation

In this section we evaluate the data augmentation strategies described in Section 2.2. The purpose of these experiments is to improve the classification performance of the previously evaluate ensemble approaches. In all experiments of this section we will use the weighted voting scheme for ensemble methods. Furthermore, as data augmentation strategies are computationally expensive in time and storage, the results of this section are based on fifty runs instead of one hundred.

In Table 2 experimental results shows that augmenting the data by using EDA and ADV improve the ensembles performance, while maintaining the standard deviation low compared to Single BERT. The UDA strategy actually seems to hurt the performance, but still having lower variance. The overall trend can be seen in the left plot of Figure 2, where ADV strategy seems to be the best choice to augment text. In Figure 3 we show the boxplot of fifty runs of the *Ensemble* (20) for different data augmentation strategies. Note that the blue bloxplot,

Table 3
Evaluation of models, trained without dropout, on validation data with F_1-score on aggressive category.

Model	vanilla	eda	uda	hard_adv
Single BERT	79.70 ± .74	79.28 ± .81	79.85 ± .71	79.06 ± .73
Ensemble (5)	80.43 ± .37	80.34 ± .64	**80.28** ± .40	80.12 ± .45
Ensemble (10)	80.45 ± .33	80.37 ± .52	80.26 ± **.30**	80.53 ± .35
Ensemble (20)	**80.46** ± **.25**	**80.53** ± **.37**	80.18 ± .24	**80.75** ± **.25**

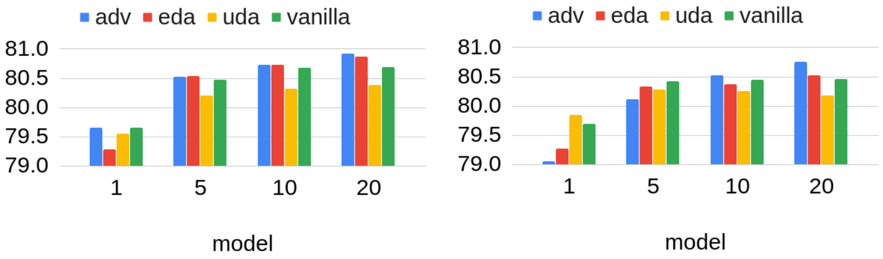

Figure 2: Histograms performance of Ensemble (20) with different Data Augmentation strategies. The left plot are fine-tuned BERTs with dropout, whereas the right plot are without dropout.

corresponding to the vanilla (no data augmentation) seems to have lower results than the red boxplot of the adversary data augmentation. Furthermore, the variance is still low in all data augmentation strategies.

Finally in Table 3 and the right plot of Figure 2, we show the experimental result of removing the dropout in last layer of the fine-tuning process of BERT. Reportedly, it is better to have dropout, but if the test data comes from very similar distribution that could not be necessary. For example, note that the vanilla strategy (no data augmentation) has very similar results with and without dropout (see the first row of both tables). However, in Table 3 note that data augmentation does not grow the performance at the same pace, and in some cases it hurts. Also note that the size of the ensemble helps, especially if the ADV strategy is used. Finally, note that UDA strategy improves the *Single BERT* model, which results in the best single model. These latter results suggest that while the UDA improves the behavior of the model by reinforcing positive examples every time, on the other hand, the use of adversaries might be doing it by learning a wider array of the dataset, specifically, the hard to learn examples. That would explain that the ensembles behave better as they go larger when using adversarial augmentation. In the final ranking of the challenge, *CIMAT-2* corresponds to an *Ensemble* (20) that used EDA data augmentation. As experiments in Table 2 and 3 show, this is not the best strategy to data augmentation since the performance is lower than ADV with and a similar or higher standard deviation. The experiments that use UDA and ADV was obtained once the challenge was finished.

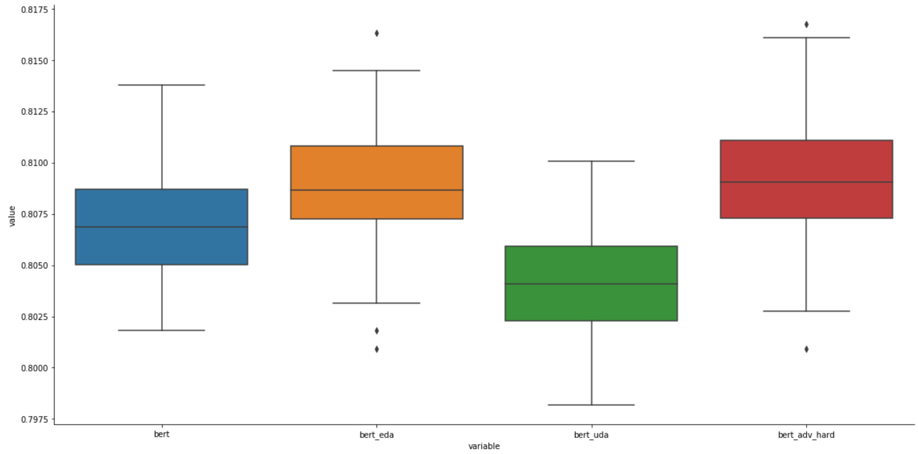

Figure 3: Boxplots on fifty runs of the Ensemble (20) by using different data augmentation strategies and evaluating the F_1 for the aggressive class.

5. Conclusions

We proposed strategies for AD that are based on ensembles of fine-tuned BERTs. The weighted voting scheme seems to be helpful for combining the decisions of several models. This means competitive performance and low variance of the model. The experiments show that there is space for the data augmentation paradigm in the tool set of the deep learning specialist. However, even if the results suggest that there is some improvement in the results when using data augmentation, they also imply certain trade offs. The results indicate that there is more to be gained when working with adversarial data augmentation on ensembles because the models seems to learn in a more heterogeneous way. But the cost of doing so is quite elevated because of the need of using costly methods for the adversarial examples' generation.

Acknowledgments

Guzman-Silverio and Balderas-Paredes appreciate CONACYT's support for scholarships 925934 and 928540 respectively.

References

[1] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. R. Pardo, P. Rosso, M. Sanguinetti, Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter, in: Proceedings of the 13th International Workshop on Semantic Evaluation, 2019, pp. 54–63.

[2] M. E. Aragón, H. Jarquín, M. Montes-y Gómez, H. J. Escalante, L. Villaseñor-Pineda, H. Gómez-Adorno, G. Bel-Enguix, J.-P. Posadas-Durán, Overview of mex-a3t at iberlef 2020: Fake news and aggressiveness analysis in mexican spanish, in: Notebook Papers of

2nd SEPLN Workshop on Iberian Languages Evaluation Forum (IberLEF), Malaga, Spain, September, 2020.

[3] N. Safi Samghabadi, A. P. López Monroy, T. Solorio, Detecting early signs of cyberbullying in social media, in: Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying, European Language Resources Association (ELRA), Marseille, France, 2020, pp. 144–149. URL: https://www.aclweb.org/anthology/2020.trac-1.23.

[4] M. E. Aragón, A. P. López-Monroy, Author profiling and aggressiveness detection in spanish tweets: Mex-a3t 2018., in: IberEval@ SEPLN, 2018, pp. 134–139.

[5] M. E. Aragón, M. Á. Álvarez-Carmona, M. Montes-y Gómez, H. J. Escalante, L. Villasenor-Pineda, D. Moctezuma, 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, 2019.

[6] L. P. Del Bosque, S. E. Garza, Aggressive Text Detection for Cyberbullying, in: A. Gelbukh, F. C. Espinoza, S. N. Galicia-Haro (Eds.), Human-Inspired Computing and Its Applications, Springer International Publishing, Cham, 2014, pp. 221–232.

[7] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, A. Vakali, Detecting Aggressors and Bullies on Twitter, in: Proceedings of the 26th International Conference on World Wide Web Companion, WWW '17 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 2017, pp. 767–768.

[8] H. J. Escalante, E. Villatoro-Tello, S. E. Garza, A. P. López-Monroy, M. Montes-y-Gómez, L. Villaseñor-Pineda, Early detection of deception and aggressiveness using profile-based representations, Expert Systems with Applications 89 (2017) 99 – 111.

[9] G. K. Pitsilis, H. Ramampiaro, H. Langseth, Effective hate-speech detection in twitter data using recurrent neural networks, Applied Intelligence 48 (2018) 4730–4742.

[10] Z. Zhang, D. Robinson, J. Tepper, Detecting hate speech on twitter using a convolution-gru based deep neural network, in: European semantic web conference, Springer, 2018, pp. 745–760.

[11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://www.aclweb.org/anthology/N19-1423. doi:10.18653/v1/N19-1423.

[12] A. Paraschiv, D.-C. Cercel, Upb at germeval-2019 task 2: Bert-based offensive language classification of german tweets, in: Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019). Erlangen, Germany: German Society for Computational Linguistics & Language Technology, 2019, pp. 396–402.

[13] M. Mozafari, R. Farahbakhsh, N. Crespi, A bert-based transfer learning approach for hate speech detection in online social media, in: International Conference on Complex Networks and Their Applications, Springer, 2019, pp. 928–940.

[14] J. Dodge, G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi, N. Smith, Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping, arXiv preprint arXiv:2002.06305 (2020).

[15] T. Zhang, F. Wu, A. Katiyar, K. Q. Weinberger, A. Yoav, Revisiting few-sample bert fine-tuning, arXiv preprint arXiv:2006.05987 (2020). URL: https://arxiv.org/pdf/2006.05987.pdf.

[16] M. Mosbach, M. Andriushchenk, D. Klakow, On the stability of fine-tuning bert:misconceptions, explanations, and strong baselines, arXiv preprint arXiv:2006.04884 (2020).

[17] S. Vega-Pons, J. Ruiz-Shulcloper, A survey of clustering ensemble algorithms, International Journal of Pattern Recognition and Artificial Intelligence 25 (2011) 337–372.

[18] L. Rokach, Ensemble-based classifiers, Artificial Intelligence Review 33 (2010) 1–39.

[19] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, Q. V. Le, Autoaugment: Learning augmentation strategies from data, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2019, pp. 113–123.

[20] J. Wei, K. Zou, EDA: Easy data augmentation techniques for boosting performance on text classification tasks, 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), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 6382–6388. URL: https://www.aclweb.org/anthology/D19-1670. doi:10.18653/v1/D19-1670.

[21] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, Q. V. Le, Unsupervised data augmentation, 2019. URL: http://arxiv.org/abs/1904.12848, cite arxiv:1904.12848.

[22] D. Jin, Z. Jin, J. T. Zhou, P. Szolovits, Is bert really robust? a strong baseline for natural language attack on text classification and entailment, 2019. arXiv:1907.11932.

[23] J. Cañete, G. Chaperon, R. Fuentes, J. Pérez, Spanish pre-trained bert model and evaluation data, in: to appear in PML4DC at ICLR 2020, 2020.

[24] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Brew, Huggingface's transformers: State-of-the-art natural language processing, ArXiv abs/1910.03771 (2019).