BERT and SHAP for Humor Analysis based on Human Annotation

Guillem García Subies, David Betancur Sánchez, Alejandro Vaca

Instituto de Ingeniería del Conocimiento, Francisco Tomás y Valiente st. 11, EPS, B Building, 5th floor UAM Cantoblanco. 28049 Madrid, Spain
guillem.garcia@iic.uam.es david.betancur@iic.uam.es
alejandro.vaca@iic.uam.es

Abstract. This paper describes a system created for the HAHA 2021 shared task, framed within the IberLEF 2021 workshop [6]. We present an approach mainly based in fine-tuned and hyperparameter-optimized BERT models for binary, multi class and multi label classification. Our models far outperform the baselines and achieve results close to to the state-of-the-art. We also present a SHAP-values based model to explain predictions on what is humorous and what is not.

Keywords: humour Detection · BERT · Transformers · multiclass · multilabel · hyperparameter optimization

1 Introduction

Humor research has been done historically from different domains such as linguistics, history, literature and psychology. Machine learning and computational linguistics are some tools that have been implemented on certain studies [2,11,14] but there is still a lot to tackle.

This article shows the process of using a BERT [7] model in Spanish to predict some of the present tasks such as humor prediction, humor mechanism and humor target. For all the tasks, fine-tuning was performed for binary, multiclass and multilabel problems. Additionally, for the binary task, a hyperparameter optimization was performed.

After predictions were performed, some explicability models were used to show the true portions of the text that was giving the humor to give us some insights on why some of them produce laughter.

2 Related Work

Most of the work in humor detection is dated before the appearance of Transformer [16] models. After this milestone, the state-of-the-art models started to be based in Transformers.

IberLEF 2021, September 2021, Málaga, Spain.

In a generic way, Sun et al. [15] propose very interesting techniques for text classification. However these are more focused in longer text. Some of the conclusions they achieve are that he top layer of BERT is more useful for text classification, within-task and in-domain further pre-training can significantly boost its performance and that a preceding multi-task fine-tuning is also helpful to the single-task fine-tuning, but its benefit is smaller than further pre-training.

Specifically, for short and humorous texts, ColBERT [1] is a novel approach based on sentence embeddings that achieves the best state-of-the-art results for English data.

For the Spanish language, we can underline the results of the 2019 HAHA shared task [5] where the best results were BERT-based models.

Some features have been evaluated to detect humor in texts. For example, in [2], they used features such as Animal presence, Keywords and a binary variable that establishes if a tweet is a dialog to detect humor. This feature engineering, though helpful for ML models, is based on heuristics rather than on a deep understanding of language so that we can truly understand when something is funny. A language model with pure texts as input, on the other hand, may need to understand the actual meaning of words in the concrete context of that input, thus they may be able to get a deeper and more complete understanding of the role language plays on humor construction.

3 Tasks Description

The main corpus consists of 24000 texts in Spanish for training and 6000 for evaluation. For each text we predicted if it was humorous or not, the mechanism of the joke and the target or targets the joke had.

The first task, Humor Detection consisted on determining if a tweet is a joke or not (intended humor by the author or not). The performance of this task was measured using the F1 score of the 'humorous' class.

The second task consisted on Funniness Score Prediction. It consists on giving a rating from 1 to 5 for how funny is a tweet. Here we did not made a competitive model.

The third task was Humor Mechanism Classification, which consisted on predicting the mechanism by which the tweet conveys humor from a set of classes such as irony, wordplay or exaggeration. In this task, only one class per tweet was allowed. The performance of this task was measured using the Macro-F1 score.

The fourth and last task was Humor Target Classification, which consisted on predicting the target of the joke (what it is making fun of) from a set of classes such as racist jokes, sexist jokes, etc. The performance of this task was measured using the f1-macro score.

As we can see in Table 1, the dataset is unbalanced. That is the main reason why the F-measure is used as the ranking metric.

Class		**Nº Samples Task1**	**Nº Samples Task3**
non-humor		14747	14747
wordplay	humor	9253	701
reference			578
exaggeration			476
unmasking			441
misunderstanding			416
absurd			566
irony			371
analogy			319
embarrassment			301
parody			255
stereotype			230
insults			146

Table 1. Distribution of Samples

In the table below, we can see some illustrative examples of the data and their labels:

La realidad es dura pero se tiene que afrontar.	non-humor
#20CosasQueHacerAntesDeMorir: Enseñarles la diferencia entre: -Hay de haber -Ahí de lugar -Ay de exclamar - Ai se eu te pego.	reference
Te quiero pero #YoTan Twitter y tú tan Facebook.	analogy
Cambié mi contraseña de Twitter por "incorrecta", si se me olvida, twitter me la recordará: Su contraseña es "incorrecta". Soy una genio	irony
WhatsApp cayó varias veces en 2015 y vos todavía no caes que nadie te soporta.	insults
Soy virgen, lo juro por mis dos hijos!	absurd
—Bienvenido a los X-Men, ¿cuál es tu poder? —Creo regresaré con mi Ex —Muy bien, te llamaremos "Bestia".	parody
–¿Tiene pastillas para la diarrea? –No. –Ok, deme un rollo de papel higiénico :(	embarrassment
—Hola linda, ¿Por qué tan sola? —Es que me vine a tirar un pedo.	unmasking
—¿Y Thomas? —No, yo no tomo. —No, ¿Que si Thomas vino? —No me gusta el vino. —¡No! ¡¿Que si llegó Thomas?! —No, no tomaré ni aunque llegues.	misunderstanding
Teníamos una farmacia pero la cerramos porque no teníamos mas remedio. #fb	wordplay
Si yo fuera presidente haria pintar la casa de gobierno de celeste , porque soy varon #chistes #humor	stereotype
Doctor, ¿cuanto me queda de vida? - Diez... - Diez qué? - Diez, nueve, ocho, siete... #humortico	exaggeration

Table 2. Examples of the different classes

4 Models

4.1 Data Preprocessing

We performed a simple texts preprocessing where we substituted some expressions with a more normalized form:

- Every URL was replaced with the token "[URL]" so we do not get strange tokens when the tokenizer tries to process a URL. Furthermore, no semantic information about humor can be inferred from a URL, the only information relevant for the model is that there is a URL in that token.
- The hashtag characters ("#") were deleted ("#example" → "example") because the base language models we will use, are trained in generic text and might not understand their meaning. Furthermore, most of hashtags are used the same way as normal words.
- We replaced every username with the generic token "[USER]" because the exact name of a user does not really add any information about the humor. The only relevant feature is knowing if someone was mentioned or not, but not who.
- Finally we normalized every laugh ("jasjajajajj" → "haha") so we minimize the noise of the misspellings, common in social networks.

4.2 Baselines

The competition owners provided some baselines to compare our models with. The baselines consisted on the following models:

- task 1: Naive Bayes with tfidf features. (0.6493 F1 over the dev corpus)
- task 2: SVM regression with tfidf features (0.6532 RMSE over the dev corpus)
- task 3: Naive Bayes with tfidf features (0.1038 macro-F1 over the dev corpus)
- task 4: Assign label X if the tweet contains one of the "top" words for label X on the training corpus (top words were selected as the 50th to 60th most frequent words for the label) (0.0595 F1 over the dev corpus)

4.3 Language Models

For our main language models we selected BETO [4], a BERT model trained with the Spanish Unannotated Corpora (SUC) [3] that has proven to be much better than the multilingual BERT model. The fine-tuning was performed distinctly for each task, varying the last layer of the model architecture to make binary (task1), regression (task2), multiclass (task3) and multilabel (task4) predictions. For Task1, Task2 and Task3, the default loss was used. For Task4 a custom loss was included in the Trainer class from Transformers library [17] to handle multilabel data. BCEWithLogitsLoss from pytorch was used as the custom loss for this task. This loss consist on calculating the binary cross entropy for each label and then averaging the values.

In addition, for the fine-tuning process, on Task1 we carried out a Grid-search optimization over the main parameters of the neural network: learning rate, batch size and dropout rate. The search was performed with a 5-fold stratified cross-validation with the following grid: Learning rate, $(1e-6, 1e-5, 3e-5, 5e-5, 1e-4)$; batch size, $(8, 16, 32)$ and dropout rate, $(0.08, 0.1, 0.12)$. The best parameters for both models were: learning rate, $1e-5$; batch size, 16 and dropout rate, 0.1.

On the task 2, 3 and 4, the default hyperparameters were used.

On task4 for final predictions we computed the f1 metric through different thresholds on the validation set in order to convert the logits to classes. The threshold values evaluated were 0.2, 0.3 and 0.4. The best one on validation was 0.2 but on test the best result was on a 0.4 threshold.

Finally, the epochs performed for the fine tuning of the models were 5 for task1, 2 for task2, 7 for task3 and 8 for task4.

For explainability, shap values were calculated for each token of the sentences. Random sentences were evaluated for insights look-up.

5 Experiments and Results

5.1 Experimental Setup

We trained all the models with a NVIDIA Tesla P100-PCIE-16GB GPU and a Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz CPU with 500GB of RAM memory.

The software we used was Python3.8, transformers 4.5.1 [17], pytorch 1.8.1 [12], shap 0.39.0 [10] and scikit-learn 0.24.1 [13].

5.2 Results

In the Table 3, under Task1, we can see the results for our models in the test set of the first task, where we obtained the second place.

Task 1		**Task 2**		**Task 3**		**Task 4**	
Model	**Accuracy**	**Model**	**Accuracy**	**Model**	**Accuracy**	**Model**	**Accuracy**
BETO	0.87	BETO	0.68	BETO	0.25	BETO	0.31
baseline	0.66	baseline	0.66	baseline	0.1	baseline	0.05
Winner	0.885	Winner	0.62	Winner	0.33	Winner	0.42

Table 3. Results for tasks (value is Accuracy)

In the Table 3, under Task2, we can see the results for our models in the test set of the second task.

For the third task, the results dropped in terms of F1, this for the difficulty of a multiclass model. In the Table 3, under Task3, we can look at them in more depth.

Finally, for the task 4 results varied a lot. The baseline was very poor and the winner was very far on top of the other competitors. Our model did good compared to the baseline, but there is a long way to reach the winner. Results are shown on Table3, under Task4.

For the explicability shap model we visualized the sentences, highlighting the important parts that led the model to make a "humor" prediction. We found 3 main cases:

- case 1: Some jokes have a very specific format, such as dialogues between characters. For example on Fig 1 the "–" that characterize a dialog, is very important on a prediction for a humor sentence. As mentioned before in the Related Work section, Castro et al. [2] also detect these features that specify dialog in a tweet, so it is definitely a important matter on detecting humor. This, however, is a weak heuristic because the text being in conversation form is not inherently funny, it's simply a typical text form people use for expressing humor in Twitter, therefore it doesn't mean the models are really understanding how humor is constructed in general; this aspect of texts has nothing to do with the language used, the humor techniques used (such as irony, sarcasm, etc.), but with the structure of the text.
- case 2: Some jokes are not exactly on the train set, but some are very similar, so the model "overfits" (highly weight words just for being in train and not because they are humorous) under this jokes and gives high values for predictions. For example in Fig 2 seems like there is an overfitting. We searched on the train set and found this tweet:
  —Mi amor ¿me compras un teléfono? —¿Y el otro? —El otro me va a comprar un iPad —¡ME REFERÍA AL OTRO TELÉFONO! — :decepcionado: AY!!
  Both texts are very similar, so the model overfits.
- case 3: Finally there is the case that we think represents the best kind of prediction. Where the model understands relations between words and set the context as a humorous one. One example can be seen on Fig 3

Fig. 1. Case for format joke

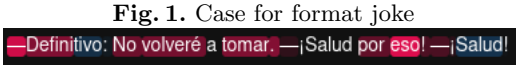
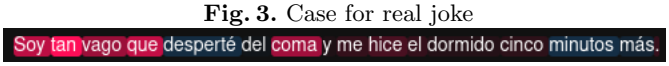


Fig. 2. Case for repeated joke

Fig. 3. Case for real joke



6 Conclusions and Future Work

Through this shared task, we have seen that NLP can be of great help in detecting and classifying humorous and non-humorous texts and there is still a long way to go. As was explained before, when analyzing which parts of the text the models use for deciding whether the text is humorous or not are based on heuristics such that whether or not the tweet represents a conversation. This shows that there is still much work to do until language models are able to understand the inherent semantics of the text so well that it can really understand the aspects of the texts, independently of the text form, that causes laughter. However, humor is an expression of high-level intelligence, expressed in sophisticated communication techniques, therefore only understanding the text meaning is probably not enough for many cases.

The results obtained by our systems are very promising given their great performance and their simplicity. Furthermore, the use of explicability models can really help get some insight on models behaviour for this kind of data. All this is very significant and could lead to much better results when combined with other improvements from the state-of-the-art.

We believe that our results could improve a lot using specific language models trained with corpora from social networks like TWilBert [8] for Spanish tweets. Finally, we have proven that good hyperparameters are also key for a good neural network so a better search, like the Population Based Training, though computationally expensive, [9], would further improve the model.

Acknowledgments

This work has been partially funded by the Instituto de Ingeniería del Conocimiento (IIC) and the hardware used was also provided by the IIC.

References

1. Annamoradnejad, I., Zoghi, G.: Colbert: Using bert sentence embedding for humor detection (2021)
2. Castro, S., Cubero, M., Garat, D., Moncecchi, G.: Is this a joke? detecting humor in spanish tweets (11 2016). https://doi.org/10.1007/978-3-319-47955-2_12
3. Cañete, J.: Compilation of large spanish unannotated corpora (May 2019). https://doi.org/10.5281/zenodo.3247731, https://doi.org/10.5281/zenodo.3247731
4. Cañete, J., Chaperon, G., Fuentes, R., Pérez, J.: Spanish pre-trained bert model and evaluation data. In: to appear in PML4DC at ICLR 2020 (2020)

5. Chiruzzo, L., Castro, S., Etcheverry, M., Garat, D., Prada, J.J., Rosá, A.: Overview of haha at iberlef 2019: Humor analysis based on human annotation. In: IberLEF@ SEPLN. pp. 132–144 (2019)
6. Chiruzzo, L., Castro, S., Góngora, S., Rosá, A., Meaney, J.A., Mihalcea, R.: Overview of HAHA at IberLEF 2021: Detecting, Rating and Analyzing Humor in Spanish. Procesamiento del Lenguaje Natural **67**(0) (2021)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2018)
8. Ángel González, J., Hurtado, L.F., Pla, F.: Twilbert: Pre-trained deep bidirectional transformers for spanish twitter. Neurocomputing (2020). https://doi.org/https://doi.org/10.1016/j.neucom.2020.09.078, http://www.sciencedirect.com/science/article/pii/S0925231220316180
9. Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W.M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., Fernando, C., Kavukcuoglu, K.: Population based training of neural networks (2017)
10. Lundberg, S.M., Lee, S.I.: Shap: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 4765–4774. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf
11. Mihalcea, R., Strapparava, C.: Learning to laugh (automatically): Computational models for humor recognition. Computational Intelligence **22**, 126–142 (05 2006). https://doi.org/10.1111/j.1467-8640.2006.00278.x
12. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
13. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
14. Sjöbergh, J., Araki, K.: Recognizing humor without recognizing meaning (07 2007). https://doi.org/10.1007/978-3-540-73400-0_59
15. Sun, C., Qiu, X., Xu, Y., Huang, X.: How to fine-tune bert for text classification? (2020)
16. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2017)
17. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Online (Oct 2020), https://www.aclweb.org/anthology/2020.emnlp-demos.6