

# NLPIR-UNED at CheckThat! 2023: Ensemble of Classifiers for Check-Worthiness Estimation

Juan R. Martinez-Rico<sup>1</sup>, Lourdes Araujo<sup>1,2</sup> and Juan Martinez-Romo<sup>1,2</sup>

<sup>1</sup>NLP & IR Group, Dpto. Lenguajes y Sistemas Informáticos, Universidad Nacional de Educación a Distancia (UNED), Madrid 28040, Spain

<sup>2</sup>Instituto Mixto de Investigación - Escuela Nacional de Sanidad (IMIENS)

## Abstract

This article outlines the NLPIR-UNED team's strategies for Task 1B in the CLEF 2023 CheckThat! Lab. The goal of this task is to determine if a text fragment from a tweet or a debate/speech is worth fact-checking. Our team devised three main approaches based on ensemble models for this binary classification task. For the English version of subtask 1B, which involves classifying text fragments from debates/speeches, we utilized an ensemble classifier composed of three transformer models that were fed different sentences from the debate/speech. On the other hand, for the Spanish version of subtask 1B, which requires classifying tweets, we have tried two more strategies, an ensemble classifier composed of three different transformer models in Spanish that receive the same tweet, and the one that we finally use: an ensemble classifier that combined a transformer model and two feed-forward neural networks (FFNN). The transformer model processes the tweet's text, while the two FFNNs receive as input TF-IDF vectors and LIWC features extracted from the text, respectively. With these approaches, our team achieved the fourth position in subtask 1B English and the same position for subtask 1B Spanish.

## Keywords

Check-worthiness Classification, Transformer Models, Ensemble of Classifiers

## 1. Introduction

Fake news is a growing problem that has been amplified by the rise of social media and the ease of spreading misinformation online. This phenomenon can have serious consequences, such as influencing political elections, spreading harmful health information, and causing social unrest. Traditional fact-checking methods are often slow and labor-intensive, making them ineffective at keeping up with the speed at which false information can spread. This has led to the development of automated methods to detect and combat fake news, such as machine learning algorithms that can quickly analyze large volumes of data and identify suspicious patterns. These automated methods have the potential to provide a more efficient and effective approach to combating fake news.

One of the fundamental tasks to perform if we want to detect fake news in news or message flows on a social network is the selection of the statements to check. This is precisely what task

---


CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

✉ jrmartinezrico@invi.uned.es (J. R. Martinez-Rico); lurdes@lsi.uned.es (L. Araujo); juaner@lsi.uned.es (J. Martinez-Romo)

🆔 0000-0003-1867-9739 (J. R. Martinez-Rico); 0000-0002-7657-4794 (L. Araujo); 0000-0002-6905-7051 (J. Martinez-Romo)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

1 of the CheckThat! Lab [1] aims to do. Our team has focused on variant B of this task, which contemplates only the use of textual information, and in the English and Spanish languages. For the English version the organizers provide a dataset generated from the transcript of a debate, while for the Spanish version the dataset is composed of a series of tweets. In both cases each instance is annotated with the values *Yes* if the sentence/tweet contains a factual statement and may be harmful, or *No* otherwise.

We have organized the rest of the article as follows: in Section 2 we make a brief review of the different approaches carried out in recent years to the task of estimating the check-worthiness of a statement, in Section 3 we explain our different approaches to this task, Section 4 discuss the results obtained, and Section 5 contains our conclusions and future work.

## 2. Related Work

The task of estimating check-worthiness of a sentence has had different approaches as the available models and tools have evolved. The approaches considered include word embeddings, bag-of-words representations, and heuristic rules to classify claims, and multilayer perceptron or support vector machine models [2], recurrent neural networks with attention, combining word2vec embeddings, part-of-speech tags, and syntactic dependencies [3], learning-to-rank approaches based on the MART algorithm, using word embeddings, named entities, part-of-speech tags, sentiment labels, and topics as features [4], k-nearest neighbors classifiers with character n-gram representations, considering linguistic lexicons and named entities [5], or support vector machines and random forests classifiers with information retrieval nutritional labels as representations [6].

Subsequently, more sophisticated representation models have been applied, such as training a feed-forward neural network with Standard Universal Sentence Encoder embeddings and using different variations of embeddings and training epochs [7].

Finally, with the generalization of transformer models as a basic tool in almost any task related to general language processing, most approaches have used this strategy [8], either using pre-trained models in generic documents [9, 10], in other languages [11], or in a specific domain such as health [12]. The approaches that have appeared more recently and have a superior performance than the solo transformer models, are the ensembles of classifiers [13]. These models typically include two or more different transformer models, or transformers pre-trained in different documents, or a combination of transformers and other types of classifiers such as feed-forward neural networks (FFNN). In our proposal, we use these two types of approaches: an ensemble with several transformers, and an ensemble composed of a transformer and two FFNNs that allows incorporating lexical and text analysis features.

## 3. Approaches to Check-Worthiness Estimation

In this 2023 edition of the CheckThat! Lab, the goal of subtask 1B is to determine if a given tweet is worth checking (binary classification), taking into account whether this tweet contains a factual statement that can be verified and whether it could be harmful. This task is offered in three languages: English, Spanish and Arabic. The organizers provide three different datasets

**Table 1**  
Size of provided datasets.

Language	train	dev	dev-test	test
English	16876	5625	1034	318
Spanish	7490	2500	5000	5000

for each language with which the models can be developed, in addition to the test dataset used for the competition. In Table 1 we can see the number of instances of each dataset for the languages in which our team has participated.

To tackle this subtask, our team has evaluated three strategies, all based on ensemble classifiers. The first of them is an ensemble classifier composed of a transformer model, a feed forward neural network (FFNN) whose inputs are TF-IDF vectors, and a FFNN whose inputs are text analysis indicators. Here, the objective is to complement the latent features that a transformer model is able to extract from plain text, with other types of features such as TF-IDF vectors extracted from that same text, and the features provided by the text analysis tool Linguistic Inquiry and Word Count (LIWC) [14].

The other two strategies are also ensemble classifiers but this time they contain three transformer models although used differently. In the subtask proposed for the English language, the dataset is composed of sentences extracted from a debate and in the first column appears an identifier that, after carefully examining the instances of training and test, we have assumed is the order in which the sentences appeared in the debate. That is why we wanted to explore the possibility of taking advantage of this information, making the ensemble classifier receive in its input 3 the instance to be evaluated, and in its inputs 2 and 1 the two instances immediately prior to the current one existing in the dataset, taking into account that there are gaps we assume generated when making the training-dev-test partition. Our hypothesis is that providing context information (previous sentences) to the sentence to be evaluated can be useful in determining the check-worthiness of that sentence.

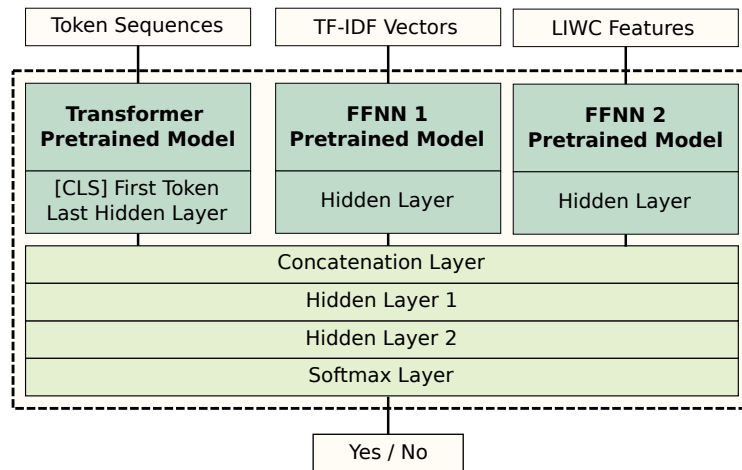
For the subtask proposed in Spanish the dataset is composed of tweets, which we cannot identify as related following any type of order. That is why we have chosen to use an ensemble composed of three different transformer models pre-trained all of them in Spanish, hoping that the different behaviors that each of them may have complement each other, achieving a superior performance to each of them separately.

We detail each of these approaches below. All used pre-trained transformer models have been downloaded from <https://huggingface.co/>.

### **3.1. Transformer-FFNN Ensemble**

#### **3.1.1. Method**

To check whether different types of input generated from the same text can complement each other and lead to greater efficiency in detecting whether that text deserves to be verified, we need to be able to handle these three types of input simultaneously for each instance of the dataset. Therefore, we have developed an ensemble model (Figure 1) composed of a transformer



**Figure 1:** Transformer-FFNN ensemble.

that processes the text as a sequence, a FFNN classifier that admits as input that text in the form of a TF-IDF vector, and a second FFNN classifier that has as inputs the discrete features generated by the LIWC text analysis tool (93 features for the English language and 90 features for the Spanish language). The hidden layers of the FFNNs and the first token of the last hidden layer of the transformer (classification token) are concatenated and form the first layer of the ensemble classifier. Behind this concatenation layer are two hidden layers and one output layer. It is also possible to disable one of the hidden layers by configuration. Before training the ensemble classifier, the transformer and the two FFNN models are trained separately on the same dataset and stored in binary files. These models are then loaded in evaluation mode in the ensemble classifier to prevent their parameters from being modified during ensemble training.

### 3.1.2. Training Strategy

To determine the configuration with the best performance, the system was configured to use deterministic algorithms and the tests were repeated for 10 different random seeds, obtaining the average of the precision, recall, and F1 measures. An early stopping mechanism has also been implemented. This mechanism stores the updated state of parameters after each epoch and stops training when there have been no improvements in the F1 measure over the *dev* dataset in the last  $n$  epochs (default value 2), then selecting the saved configuration with the best F1 measure during that interval. After performing a grid search for each component separately and for the ensemble classifier, the following hyperparameters have been selected for the English language:

- FFNN hidden layer size: 1000.
- FFNN seed value: 0.
- FFNN max. epochs: 250.
- FFNN (TF-IDF) activation function: *relu*.

- FFNN (LIWC) activation function: *sigmoid*.
- Transformer pre-trained model: *bert-base-uncased*.
- Transformer max. sequence length: 128.
- Transformer max. epochs: 10.
- Transformer seed value: 63.
- Ensemble activation function: *relu*.
- Ensemble hidden layers: 2.
- Ensemble dropout: 0.
- Ensemble max. epochs: 10.

And for the Spanish language the same hyperparameters have been used except for the following:

- FFNN seed value: 96.
- Transformer pre-trained model: *bertin-project/bertin-roberta-base-spanish* [17].
- Transformer seed value: 70.

## 3.2. Ensemble of Transformers

### 3.2.1. Method

In the English subtask, as discussed above, we wanted to take advantage of what appears to be a flow of sentences to provide context for the sentence to be evaluated. For this, we have developed an ensemble classifier (Figure 2) composed of three transformer models so that each one can receive a different sentence. During the processing of the *training* and *test* datasets, we look for the two instances that have the identifiers immediately before  $i - n$ ,  $i - n - m$  to the instance  $i$  to evaluate. These three sentences form the input of the ensemble classifier using the class value of the instance  $i$ . As a pre-trained model, we use the same in each of the three transformer components: *bert-base-uncased*.

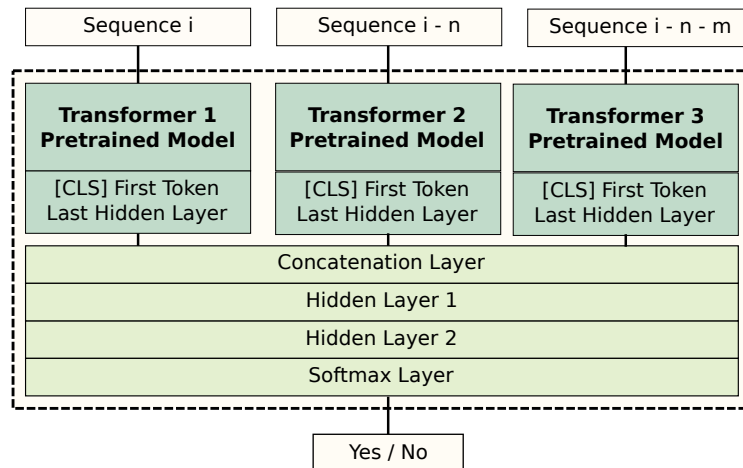
For the Spanish subtask, since the datasets contain tweets, we have assumed that they had no relationship between them by adopting the strategy of providing the same sentence (tweet) in the three inputs of the ensemble classifier, and use a different pre-trained model on each transformer component. With this, we hoped that the different pre-trainings that each one has had are somehow complementary and allow us to determine more precisely if the sentence to evaluate is worth checking. Specifically, we have used the following models:

- Transformer 1: *PlanTL-GOB-ES/roberta-large-bne* [15].
- Transformer 2: *dccuchile/bert-base-Spanish-wwm-cased* [16].
- Transformer 3: *bertin-project/bertin-roberta-base-spanish*.

### 3.2.2. Training Strategy

The hyperparameters used in both languages were as follows:

- Transformer max. sequence length: 128.



**Figure 2:** Ensemble of Transformers ( $i$ : current instance,  $i-n$ : a previous instance,  $i-n-m$ : another instance prior to  $i-n$ )

- Transformer max. epochs: 10.
- Transformer seed value: 63.
- Ensemble activation function: *relu*.
- Ensemble hidden layers: 2.
- Ensemble dropout: 0.
- Ensemble max. epochs: 10.

## 4. Results

This section describes the results obtained with the three strategies described. The F1 score on the positive class is the evaluation measure used by the organizers in the competition.

### 4.1. English Subtask

The organizers provided three datasets for both languages: *training*, *dev-test* and *test*. Since the values of F1 measure obtained for the English subtask with the *test* dataset were very high even for the  $n$ -gram baseline, we also performed the evaluation joining the *dev-test* and *test* datasets. In all cases the training has been done using only the training dataset. Table 2 shows the averaged results of the models after 10 runs with different random seeds.

The last column shows the F1 measure calculated on the *test* dataset, along with the reference values provided by the baselines. As we can see, the FFNN with input in the form of TF-IDF vectors is not able to overcome the baseline  $n$ -gram. On the other hand, the transformer model *bert-base-uncased* outperforms the two ensemble classifiers, indicating that, at least in this *test* dataset, the combination of models and inputs does not bring any improvement to the classifier's performance. In principle, ensemble models would be expected to outperform the solo transformer model as they should be able to select the best information present in each

**Table 2**

English Subtask: Average results on *dev-test + test* and *test* datasets. The primary submission appears in bold.

Model	Precision	Recall	F1	F1 (test)
Majority Baseline				0.000
Random Baseline				0.220
Ngram Baseline				0.821
FFNN	0.712	0.556	0.624	0.803
Transformer bert-base-uncased	0.770	0.783	0.777	0.946
Ensemble Transformer + LIWC FFNN + TF-IDF FFNN	0.779	0.791	0.785	0.937
<b>Ensemble of Transformers</b>	0.759	0.803	0.780	<b>0.937</b>

**Table 3**

English Subtask: Results on competition *test* dataset.

Submission	Accuracy	Precision	Recall	F1
<b>Primary (Ensemble of Transformers)</b>	0.909	0.954	0.769	<b>0.851</b>
Contrastive (Transformer + LIWC FFNN + TF-IDF FFNN)	0.937	0.978	0.833	0.900

of their three inputs to evaluate a given sentence. It is possible that the context search that we intended to use by looking for sentences before the current one is not working in this *test* dataset because there is too much distance between them. Remember that we have assumed that the identifier of each sentence is the order it has within the debate. Thus, having partitioned the debate by randomly extracting sentences to create the *train*, *test* and *dev-test* datasets, the sentences are no longer consecutive in these datasets. Regarding the other type of ensemble classifier, we can also assume that in this *test* dataset the inputs of TF-IDF vectors and LIWC features do not provide enough information to improve the behavior of the ensemble model.

If we look at the evaluation carried out on the *dev-test + test* dataset, we see that in this case the two ensemble models surpass to the transformer model alone, although by a small margin. For the ensemble model that searches for previous sentences, performance may be improving as two datasets have been rejoined, making the distances between the current sentence and the two immediately preceding sentences smaller and contributing to this context information more effectively.

For the main submission, we selected the ensemble model that seeks context information in two previous sentences (shown in Table 2 as *Ensemble of Transformers*). Although it does not have the highest average F1 score, the differences with the other models are minimal, and we wanted to see how this approach performs in the competition. With this configuration we have achieved the fourth best result among the eleven participating teams with an F1 measure of 0.851 (Table 3), being 0.898 the value obtained by the team classified in first position and 0.462 the one obtained by the baseline.

**Table 4**

Spanish Subtask: Average results on *dev-test + test* and *test* datasets. The primary submission appears in bold.

Model	Precision.	Recall	F1	F1 (test)
Majority Baseline				0.000
Random Baseline				0.133
Ngram Baseline				0.511
FFNN	0.635	0.458	0.532	0.548
Transformer bertin-roberta-base-spanish	0.708	0.614	0.658	0.664
<b>Ensemble Transformer + LIWC FFNN + TF-IDF FFNN</b>	0.654	0.706	0.679	<b>0.684</b>
Ensemble of Transformers	0.675	0.707	0.690	0.692

## 4.2. Spanish Subtask

Table 4 shows the results obtained for the subtask in Spanish. These results are also averaged for ten random seeds in two different evaluation datasets: the *test* dataset (last column) and the union of the *test* and *dev-test + test* datasets (rest of columns).

Here, because the datasets contain tweets, we were unable to apply context information search. Instead, we have configured the ensemble composed of three different transformer models. The first thing to note, looking at the results, is that in both datasets of evaluation the ensemble models clearly surpass the transformer working alone, the FFNN, and the baselines. This is the expected behavior, and that in the English subtask has not been seen.

Analyzing the two ensemble models, we see that the one composed of three different transformer models (*roberta-large*, *roberta-base*, *bert-base*) also pre-trained with different data in Spanish, get to obtain the best F1 measure both in the *test* dataset (F1 = 0.692) and in the *test + dev-test* dataset (F1 = 0.690), so we deduce that the latent features extracted by each of them are complementary and help to improve the ensemble classifier as a whole. The ensemble model that uses a single transformer and two FFNNs obtains somewhat lower results (F1 = 0.684 in the *test* dataset), but surpasses the transformer alone. This indicates that the features based on text analysis extracted by the LIWC tool, along with the TF-IDF vectors extracted from the tweet text, also complement in some way the latent features extracted by the transformer model.

For the main submission of this subtask in Spanish, we have considered the two ensemble models. Again, in this case the performance differences between the two have been small so we have chosen to send the results of the model that uses a single transformer, shown in Table 4 as *Ensemble Transformer-FFNN*, thus sending a totally different configuration in each version of subtask 1B. With this model, we have managed to place ourselves in the fourth position with an F1 measure of 0.589 (Table 5), being 0.641 the F1 measure obtained by the winning team and 0.172 the one obtained by the reference baseline.

## 5. Conclusions and Future Work

To tackle the task of estimating the check-worthiness of a sentence or tweet, in this edition of the CheckThat! Lab our team has evaluated several strategies that involve the use of ensemble



**Table 5**Spanish Subtask: Results on competition *test* dataset.

Submission	Accuracy	Precision	Recall	F1
<b>Primary (Transformer + LIWC FFNN + TF-IDF FFNN)</b>	0.923	0.643	0.544	<b>0.589</b>
Contrastive (Ensemble of Transformers)	0.930	0.699	0.542	0.611

classifiers.

One of them, has been based on the use of an ensemble classifier containing a transformer model, a feed-forward neural network (FFNN) with TF-IDF vectors at the input, and a second FFNN with features extracted by the text analysis tool LIWC. This model, as we expected, has been able to surpass the solo transformer models in the Spanish subtask and is the one we have used to make the main submission in this language, obtaining the fourth best result (F1 = 0.589) among the seven participants. In this same subtask we have also evaluated an ensemble model that contained three different transformer models in Spanish, obtaining similar results.

In subtask in English language, the differences between the results obtained by the ensemble models and the transformer models alone have been much smaller, the latter surpassing the ensemble in the *test* dataset. The main difference between both subtasks is the content of the datasets: sentences of a debate in the subtask in English, and tweets in the subtask in Spanish. This, in principle should not be the reason for this similarity in performance since ensemble models usually perform better than solo transformers. Still, we have selected the ensemble classifier composed of three transformer models fed with the current sentence and two previous sentences to provide context information. With this configuration, our team has obtained the fourth best F1 measure (0.851) among the eleven participating teams. We think that this model can give good results when, for example, we are analyzing a text to identify the sentences that are worth checking, because in this case we could select the two sentences immediately prior to the current one, unlike what happens with the datasets of this subtask where not all the sentences of the debate are available.

In the future, we intend to further explore alternative methods of integrating diverse models into an ensemble classifier, thereby expanding the range of features utilized in identifying sentences that need to be verified.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation within the DOTT-HEALTH Project (MCI/AEI/FEDER, UE) under Grant PID2019-106942RB-C32 and OBSER-MENH Project (MCIN/AEI/10.13039/501100011033 and NextGenerationEU/PRTR) under Grant TED2021-130398B-C21 as well as project RAICES (IMIENS 2022).

## References

- [1] F. and Alam, A. Barrón-Cedeño, G. S. Cheema, S. Hakimov, M. Hasanain, C. Li, R. Míguez, H. Mubarak, G. K. Shahi, W. Zaghouani, P. Nakov, Overview of the CLEF-2023 CheckThat!

Lab Task 1 on Check-Worthiness in Multimodal and Multigenre Content, in: Working Notes of CLEF 2023—Conference and Labs of the Evaluation Forum, CLEF '2023, Thessaloniki, Greece, 2023.

- [2] C. Zuo, A. Karakas, R. Banerjee, A hybrid recognition system for check-worthy claims using heuristics and supervised learning, in: CEUR workshop proceedings, volume 2125, 2018.
- [3] C. Hansen, C. Hansen, J. G. Simonsen, C. Lioma, The Copenhagen Team Participation in the Check-Worthiness Task of the Competition of Automatic Identification and Verification of Claims in Political Debates of the CLEF-2018 CheckThat! Lab (2018) 8.
- [4] K. Yasser, M. Kutlu, T. Elsayed, bigIR at CLEF 2018: Detection and Verification of Check-Worthy Political Claims (2018) 10.
- [5] B. Ghanem, M. Montes-y Gomez, F. Rangel, P. Rosso, UPV-INAOE-Autoritas - Check That: Preliminary Approach for Checking Worthiness of Claims (2018) 6.
- [6] R. Agez, C. Bosc, C. Lespagnol, J. Mothe, N. Petitcol, IIRIT at CheckThat! 2018, Cappellato et al.[5] (2018).
- [7] L. Favano, M. J. Carman, P. L. Lanzi, TheEarthIsFlat's Submission to CLEF'19 CheckThat! Challenge (2019) 12.
- [8] E. Williams, P. Rodrigues, V. Novak, Accenture at CheckThat! 2020: If you say so: Post-hoc fact-checking of claims using transformer-based models, Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum (2020) 12. URL: [http://ceur-ws.org/Vol-2696/paper\\_226.pdf](http://ceur-ws.org/Vol-2696/paper_226.pdf).
- [9] 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), 2019, pp. 4171–4186.
- [10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv:1907.11692 [cs] (2019). URL: <http://arxiv.org/abs/1907.11692>.
- [11] W. Antoun, F. Baly, H. Hajj, Arabert: Transformer-based model for arabic language understanding, arXiv preprint arXiv:2003.00104 (2020).
- [12] D. Q. Nguyen, T. Vu, A. T. Nguyen, BERTweet: A pre-trained language model for English Tweets, arXiv preprint arXiv:2005.10200 (2020).
- [13] N. Buliga, M. Raschip, Zorros at CheckThat! 2022: Ensemble Model for Identifying Relevant Claims in Tweets (2022).
- [14] J. W. Pennebaker, R. L. Boyd, K. Jordan, K. Blackburn, The development and psychometric properties of LIWC2015, Technical Report, 2015.
- [15] A. G. Fandiño, J. A. Estapé, M. Pàmies, J. L. Palao, J. S. Ocampo, C. P. Carrino, C. A. Oller, C. R. Penagos, A. G. Agirre, M. Villegas, MarIA: Spanish Language Models, *Procesamiento del Lenguaje Natural* 68 (2022). URL: <https://upcommons.upc.edu/handle/2117/367156#.YyMTB4X9A-0.mendeley>. doi:10.26342/2022-68-3, publisher: Sociedad Española para el Procesamiento del Lenguaje Natural.
- [16] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, Spanish Pre-Trained BERT Model and Evaluation Data, in: PML4DC at ICLR 2020, 2020.
- [17] J. D. I. R. y. E. G. P. y. M. R. y. P. V. y. P. G. d. P. S. y. M. Grandury, BERTIN: Efficient

Pre-Training of a Spanish Language Model using Perplexity Sampling, *Procesamiento del Lenguaje Natural* 68 (2022) 13–23. URL: <http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/6403>.