

# Using Transformers on Noisy vs. Clean Data for Paraphrase Identification in Mexican Spanish

Antonio Tamayo<sup>1</sup>, Diego A. Burgos<sup>2</sup> and Alexander Gelbukh<sup>1</sup>

<sup>1</sup>*Instituto Politécnico Nacional (IPN), Centro de Investigación en Computación (CIC), Av. Juan de Dios Batiz, s/n, 07320, Mexico City, Mexico*

<sup>2</sup>*Wake Forest University, 1834 Wake Forest Road, Winston-Salem, NC 27109, Winston Salem, USA*

## Abstract

Paraphrase identification is relevant for plagiarism detection, question answering, and machine translation among others. In this work, we report a transfer learning approach using transformers to tackle paraphrase identification on noisy vs. clean data in Spanish as our contribution to the PAR-MEX 2022 shared task. We carried out fine-tuning as well as hyperparameters tuning on BERTIN, a model pre-trained on the Spanish portion of a massive multilingual web corpus. We achieved the best performance in the competition (F1 = 0.94) by fine-tuning BERTIN on noisy data and using it to identify paraphrase on clean data.

## Keywords

Paraphrase identification, Language models, Transformers, Transfer learning.

## 1. Introduction

The process of identifying whether a pair of sentences express the same meaning is known as paraphrase identification [1,2]. This is a challenging task because paraphrasing uses a myriad of syntactic and semantic mechanisms to re-express a message into a different form while keeping as close as possible to the original meaning. This makes automatic recognition of paraphrasing a high-level task. A type of text similarity measurement task, paraphrase identification also contributes to more complex systems such as plagiarism detection, question answering, language generation, and dialogue systems.


In this paper, we report our contribution to PAR-MEX 2022 [3], a shared task for sentence-level paraphrase identification in Mexican Spanish. The participants were asked to identify the relation between two sentences of a sentence pair and to classify them as P (paraphrase) or NP (non-paraphrase). For the task, we followed a two-folded transfer learning approach, namely, a) the original language model we used was trained on non-paraphrase data and b) the model was fine-tuned and tested on both noisy and clean paraphrase data. We report the results of our experiments with BERTIN [4], a transformer-based model for Spanish. Interestingly, the model we fine-tuned on noisy data yields the best classification results (F1 = 0.9424) on the clean dataset.

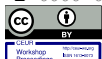
The paper is structured as follows. Section two describes the dataset, preprocessing, resources, and models, as well as the overall methodology followed for the task. Section three reports our results and discussion on both noisy and clean data including the best overall result for the task. Section four summarizes related work on paraphrase identification and Section five concludes.

---

IberLEF2022, September 2022, A Coruña, Spain

 atamayoh2019@cic.ipn.mx (A. Tamayo); burgosda@wfu.edu (D.A. Burgos); gelbukh@gelbukh.com (A. Gelbukh)

 0000-0002-5984-7463 (A. Tamayo); 0000-0002-5784-3952 (D.A. Burgos); 0000-0001-7845-9039 (A. Gelbukh)



© 2022 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)

## 2. Methodology

### 2.1. Dataset

The dataset for this task consists of sentence pairs taken from texts about molecular cuisine, sushi, tequila, kebab, vegan food, food-truck, and food on Mexican ofrenda. Each sentence pair is made up of the original sentence plus a second sentence, which may or may not be a paraphrase of the original sentence. Low- and high-level paraphrases were created following the methodology in [5]. Table 1 shows an example of both classes, which we translated into English for illustrative purposes. Italics in brackets show the original sentence in Spanish.

**Table 1**

A sentence pair sample from the dataset

Sentence 1	Sentence 2	Label
The more delicate flavors can be preserved with a little science. <i>[Los sabores más delicados pueden preservarse con un poco de ciencia.]</i>	The media popularize, more than the hospitality industry, expressions such as molecular gastronomy or molecular cuisine. <i>[Los medios popularizan, más que la hostelería, expresiones como gastronomía molecular o cocina molecular.]</i>	NP
Tequila is a term of controlled origin and designates the agave distillate made in certain regions of Mexico. <i>[El nombre tequila es un término de origen controlado y que designa el destilado de agave elaborado en regiones determinadas de México.]</i>	Tequila is a very specific name given to the agave distillate made in different areas of Mexico. <i>[El nombre tequila es muy específico y se le da al destilado de agave elaborado en distintas zonas de México.]</i>	P

We ran our experiments on two datasets of similar nature that we received from PAR-MEX organizers, namely a noisy dataset and a clean dataset. While originally PAR-MEX 22 did not intend to distribute a noisy dataset, a labeling error led the organizers to replace the original (noisy) dataset with the fixed (clean) dataset. As much work had been done on the noisy dataset already, we leveraged on this setback and, as we describe later on, we fine-tuned and tested our model on both noisy and clean datasets, hence the need to describe them below. Should the reader be interested in this data, both the noisy and the clean partitions that we used for our local experiments can be downloaded at [https://drive.google.com/drive/folders/1\\_lmjtkeajie3YcbsA6OKxg0xOWgWP9nI?usp=sharing](https://drive.google.com/drive/folders/1_lmjtkeajie3YcbsA6OKxg0xOWgWP9nI?usp=sharing).

#### 2.1.1. Noisy data

According to [6], the quality of a dataset can usually be characterized by two information sources: (1) attributes and (2) class labels. Noise can be defined as anything that obscures the relationship between attribute description and the assigned class [7]. In this dataset, the label assigned to 245 sentence pairs does not correctly classify the set of attributes of the labeled sentence pair. Therefore, we are dealing with a type of external noise, that is, errors in the assigned class labels. Table 2 shows the way the organizers partitioned this dataset.

**Table 2**

PAR-MEX partitions of the noisy dataset

Dataset	Total sentence-pairs	Paraphrase	Non-paraphrase
Training	7601 (100%)	1501 (20%)	6100 (80%)
Validation	100 (100%)	23 (23%)	77 (77%)
Test	2842 (100%)	565 (19.88%)	2277 (80.12%)
<b>Total</b>	<b>10543 (100%)</b>	<b>2089 (19.81%)</b>	<b>8454 (80.19%)</b>

### 2.1.2. Clean data

According to the organizers, the clean dataset came without the 245 mislabeled sentence pairs in the noisy dataset. Therefore, we assume that all the assigned labels in this dataset correctly classify the sentence pairs in it and that there is no noise. Table 3 shows the way the organizers partitioned this dataset.

**Table 3**

PAR-MEX partitions of the clean dataset

Dataset	Total sentence-pairs	Paraphrase	Non-paraphrase
Training	7382 (100%)	1282 (17.37%)	6100 (82.63%)
Validation	97 (100%)	20 (20.62%)	77 (79.38%)
Test	2819 (100%)	542 (19.23%)	2277 (80.77%)
<b>Total</b>	<b>10298 (100%)</b>	<b>1844 (17.91%)</b>	<b>8454 (82.09%)</b>

### 2.1.3. Data partitions

For the PAR-MEX task, the organizers let the participants download data for training, development, and evaluation purposes, but the participants needed to upload their predictions to a Codalab platform (<https://codalab.lisn.upsaclay.fr/competitions/2345>) in order to obtain official development and evaluation results. Because there was a limited number of development and evaluation submissions per team, we ran local experiments before submitting our best predictions to Codalab. For the sake of clarity hereafter, we define names and acronyms for data partitions according to whether they are our local data partitions or whether they are the organizers' partitions on Codalab (see below). Class labels are hidden for all the data partitions.

- *Local validation (LocVal)*. This refers to a local data partition that we defined for validation.
- *Local testing (LocTest)*. This refers to a local data partition that we defined for testing.
- *PAR-MEX development (PMDvN)*. This refers to a data partition that the organizers defined for development on Codalab on the noisy dataset.
- *PAR-MEX evaluation (PMEvalN)*. This refers to a data partition that the organizers defined for evaluation on Codalab on the noisy dataset.
- *PAR-MEX development (PMDvC)*. This refers to a data partition that the organizers defined for development on Codalab on the clean dataset.
- *PAR-MEX evaluation (PMEvalC)*. This refers to a data partition that the organizers defined for evaluation on Codalab on the clean dataset.

## 2.2. Pre-processing

We followed a transfer learning approach to tackle the paraphrasing task. For this purpose, we used the BERTIN language model. While BERTIN is not pre-trained for a text similarity task, it allows two sentences as input by changing the head for predictions to treat it as a binary classification problem rather than as a fill-mask task. This is the exact scenario we are faced with in the paraphrasing task.

Therefore, we concatenated both sentences in the sentence pair and the model deals with the problem as a sequence classification one.

### 2.3. Pre-trained model

BERTIN is a language model that was pre-trained for a fill-mask task on texts written uniquely in Spanish. It is a BERT-related [8] model based on the RoBERTa [9] model using the Spanish portion of the mc4 corpus [10]. This corpus was collected via web-scraping and contains about 416 million documents and 235 billion words, which takes approximately 1TB of uncompressed data.

Besides being the first publicly available RoBERTa-related model trained for Spanish, BERTIN features a data-centric technique called perplexity sampling which allows for reducing the needed amount of data and steps to train the model.

BERTIN uses the same setup and hyperparameters as in [9] but training requires only half of the steps. In this work, we used the same configuration of BERTIN, but we adjusted some hyperparameters during the fine-tuning process. The configuration of our fine-tuned model and the validation method used during its training process are shown later.

We chose BERTIN rather than other language models for two basic reasons. First, the organizers of PAR-MEX reported a baseline based on BERT in its Spanish version called BETO [11] using different random seeds for 5 iterations. In order to outperform the baseline, we thought of either a multilingual model based on RoBERTa or a RoBERTa-based model pre-trained from scratch only on a Spanish corpus. We found the latter feature in BERTIN, as we already described above. Secondly, BERTIN was trained on a dataset collected from websites, which we thought could be semantically closer to PAR-MEX dataset and therefore improve the model's performance.

### 2.4. BERTIN fine-tuning

For the fine-tuning process, the parameters of the model prediction head were initialized randomly. The head was modified to tackle the paraphrasing task as a binary classification problem.

Then, we conducted several experiments with various hyperparameter combinations during the fine-tuning process. We found out that the learning rate and the epochs are the most relevant hyperparameters for the paraphrasing task. As it will be shown later, our best result was achieved by setting them as follows: learning rate ( $lr$ ) =  $5e-7$ ; epochs = 7.

Hyperparameter tuning can be done easily using brute force with an extensive grid search, but it is not convenient as it is very time consuming. Therefore, we started from a default configuration ( $lr=5e-5$ ; epochs=3) and tuned it by decreasing the learning rate and, at the same time, gradually increasing the number of epochs. This approach led us to find the best hyperparameter configuration with reasonable effort and time. The code to replicate these experiments can be download at <https://colab.research.google.com/drive/1b9tdmvKo45OZaLiclZ7NKD9ALDmHiRa0?usp=sharing>.

### 2.5. Validation method

We used a random partition of the noisy training dataset into training (5700 samples), validation (1425 samples), and test (476 samples) sets. It was done iteratively five times, and the results that we present below in Table 4 are an average of the results obtained in these iterations. The same validation process applies to the clean training dataset, where the sample distribution was the following: training (5536), validation (1384), and test (462). It is worth mentioning that our validation method, with five iterations, emulated the organizer's methodology for the baseline in order to have comparable results.

## 2.6. Technical resources

We fine-tuned BERTIN using the Transformers library through the checkpoint available at the Hugging Face platform (<https://huggingface.co/bertin-project/bertin-roberta-base-spanish>).

The fine-tuning process was carried out using Google Colab Pro (<https://colab.research.google.com>) with a GPU Tesla P100 with 27.3 gigabytes of available RAM. The average time consumed during the fine-tuning process for each iteration of the validation method was approximately ten minutes.

## 3. Results and discussion

Table 4 shows the results for the best model’s hyperparameter configuration found, namely, epochs = 7 and lr = 5e-7. As we mentioned above, these results are averages of F1, Recall (R), and Precision (P) for the 5 iterations of the validation method. For both the development and the evaluation phases of the competition, we achieved the best scores with our model fine-tuned on the noisy dataset. The acronyms B\_FT\_C and B\_FT\_N in the Model column stand for BERTIN fine-tuned on the clean dataset and BERTIN fine-tuned on the Noisy dataset respectively. The table also includes the baseline reported by the organizers of the competition. By the time we wanted to test B\_FT\_C on the organization’s development and the evaluation noisy partitions, the data had been replaced with the clean dataset, hence the missing values in the table.

**Table 4**  
Fine-tuned BERTIN on clean vs noisy data

Model	LocVal			LocTest			PMDvN	PMEvalN	PMDvC	PMEvalC
	P	R	F1	P	R	F1	F1	F1	F1	F1
Baseline	-	-	-	-	-	-	-	-	-	0.70
B_FT_C	0.94	0.90	0.92	0.91	0.89	0.90	n/a	n/a	0.95	0.87
B_FT_N	0.96	0.79	0.87	0.96	0.80	0.87	0.93	0.92	1	<b>0.94</b>

Our results outperformed not only the competition’s baseline but they were also the best results for the task on both the noisy and clean data. An F1 score of 0.94 on clean data and of 0.92 on noisy data is impressive for high-level tasks such as paraphrase identification. The methodology used to build the dataset [5] informs about the complexity of the task, even for human beings. Likewise, Spanish is a language with traditionally fewer resources than English, which is why the dataset developed by the organizers is a very valuable contribution to the field.

It is interesting to note that the model fine-tuned on noisy data (B\_FT\_N) achieved the best results on both the noisy and clean PAR-MEX evaluation partitions. It is also remarkable that B\_FT\_N correctly classified 100% of the samples of the clean development dataset (PMDvC). We would have liked to fine-tune the model on the clean data with a different number of epochs, which might have achieved better F1 results, but we were not able to test this hypothesis because the competition had a limited number of submissions in the development and evaluation phases. While seven epochs worked great on the noisy dataset, this configuration may have caused the model to be overfitted on the clean local dataset. This is the reason why B\_FT\_C performed well on our local test (LocTest = 0.90), but not as well on the organization’s evaluation partition (PMEvalC = 0.87) as Table 4 shows.

Let us also add a remark with regards to the inconsistency between our local F1 scores and the organization F1 scores (see F1s in Table 4). While we used the standard F1 metrics for our local runs, it seems that there was some kind of discrepancy with the organization metrics, which we were unable to figure out. This is why no matter how good or bad our local results were, we had to use one of the limited online submissions each time in order to see how the model was doing. It must be said, though, that the iterative results achieved by our model in the development and evaluation phases reflect a regular, predictable behavior of the model, mainly on the clean dataset.

All in all, the hyperparameters tuned, that is, the number of epochs and the learning rate, proved to work great with the paraphrase identification problem. Any variation we tried to this configuration led

to drastic, poorer results. Our results suggest the suitability of a RoBERTa-based model, such as BERTIN, over the BERT model used for the baseline of the paraphrasing task.

## 4. Related works

Works to tackle the paraphrasing task may be grouped in three main approaches, namely: a) similarity-based methods, b) machine learning methods, and c) deep learning methods. Both a) and b) imply feature engineering through syntactic analysis such as part-of-speech tagging, n-grams, etc. as well as word representations, usually as pre-trained word embeddings to measure the distance between a pair of documents [12,13]. Some works have tried interesting approaches such as lexical and semantic level text canonicalization [14]. In [15], one of the few relevant works for paraphrase detection in Spanish, the authors used a corpus like the one used in this work, and they proved that different metrics to measure semantic similarity to identify paraphrasing can give different results. Their best approach achieved 0.84 F1 for identifying paraphrase relationships. From another perspective, there are several approaches based on neural networks and different deep learning architectures to address paraphrasing identification [16]. In [17], an interesting application of convolutional neural networks is shown to obtain a set of rich granularity features for paraphrase detection. Likewise, in [18], it can be seen an application of bidirectional recurrent neural networks to detect paraphrasing in the Arabic language. More recently, some works with modern language models applied to paraphrasing detection have been published. [19] report the performance of the most powerful language models such as BERT, RoBERTa, XLNet [20], and ALBERT [21] under a data-augmentation strategy to tackle the task we deal with in this article.

## 5. Conclusions

In this work, we report a methodology and a system to tackle the paraphrase identification problem using transfer learning with BERTIN, which is a transformer model based on RoBERTa and pretrained from scratch on a big corpus in Spanish for the fill-mask task. Our model achieved the best result in the PAR-MEX 2022 shared task (F1 = 0.94). The model was fine-tuned by carefully choosing a hyperparameter combination and testing it on both a noisy and a clean dataset. The paraphrase identification task was treated as a sequence classification problem, which was possible by changing the model's head of prediction during the fine-tuning process.

Due to technical and logistic issues, we were no able to test as many configurations as we would have liked to shed some additional light on the fact that our model fine-tuned on noisy data outperforms another one fine-tuned on clean data. However, this raises the question about the role of data quality for paraphrase identification and, perhaps for other tasks, and the line remains open.

Despite the lack of interpretability of the features of deep-learning models, this work shows the power of a language model like BERTIN together with transfer learning, a careful hyperparameter tuning, and simple preprocessing to identify paraphrasing relationships in Spanish.

## 6. Acknowledgements

This work was done with partial support from the Mexican Government through the grant A1-S-47854 of CONACYT, Mexico, grants 20220852 and 20220859 of the Secretaría de Investigación y Posgrado of the Instituto Politécnico Nacional, Mexico. The authors thank the CONACYT for the computing resources brought to them through the Plataforma de Aprendizaje Profundo para Tecnologías del Lenguaje of the Laboratorio de Supercómputo of the INAOE, Mexico and acknowledge the support of Microsoft through the Microsoft Latin America PhD Award.

## 7. References

- [1] Dolan B, Brockett C. Automatically constructing a corpus of sentential paraphrases. in Third International Workshop on Paraphrasing (IWP2005) 2005 Jan 1.
- [2] Xu W, Callison-Burch C, Dolan WB. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015) 2015 Jun (pp. 1-11).
- [3] Bel-Enguix G, Gomez-Adorno H, Sierra G, Torres-Moreno JJM, Ortiz-Barajas JG, and Vásquez J. Overview of PAR-MEX at Iberlef 2022: Paraphrase Detection in Spanish Shared Task. *Procesamiento del Lenguaje Natural*, 2022, 69.
- [4] De la Rosa J, Ponferrada EG, Romero M, Villegas P, de Prado Salas PG, Grandury M. BERTIN: Efficient Pre-Training of a Spanish Language Model using Perplexity Sampling. *Procesamiento del Lenguaje Natural*. 2022 Mar 15; 68:13-23.
- [5] Torres-Moreno JM, Sierra G, Peinl P. A German Corpus for Similarity Detection Tasks. *Int. J. Comput. Linguistics Appl.* 2014 Jul;5(2):9-24.
- [6] Zhu X, Wu X. Class noise vs. attribute noise: A quantitative study. *Artificial intelligence review*. 2004 Nov;22(3):177-210.
- [7] Hickey RJ. Noise modelling and evaluating learning from examples. *Artificial Intelligence*. 1996 Apr 1;82(1-2):157-79.
- [8] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. 2018 Oct 11.
- [9] Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*. 2019 Jul 26.
- [10] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*. 2019 Oct 23.
- [11] Canete J, Chaperon G, Fuentes R, Ho JH, Kang H, Pérez J. Spanish pre-trained bert model and evaluation data. *Pml4dc at iclr*. 2020; 2020:2020.
- [12] Yalcin K, Cicekli I, Ercan G. An external plagiarism detection system based on part-of-speech (POS) tag n-grams and word embedding. *Expert Systems with Applications*. 2022 Jul 1; 197:116677.
- [13] Kozareva Z, Montoyo A. Paraphrase identification on the basis of supervised machine learning techniques. In *International conference on natural language processing (in Finland)* 2006 Aug 23 (pp. 524-533). Springer, Berlin, Heidelberg.
- [14] Zhang Y, Patrick J. Paraphrase identification by text canonicalization. In *Proceedings of the Australasian Language Technology Workshop 2005* 2005 Dec (pp. 160-166).
- [15] Gómez-Adorno H, Bel-Enguix G, Sierra G, Torres-Moreno JM, Martínez R, Serrano P. Evaluation of Similarity Measures in a Benchmark for Spanish Paraphrasing Detection. In *Mexican International Conference on Artificial Intelligence 2020* Oct 12 (pp. 214-223). Springer, Cham.
- [16] Lan W, Xu W. Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *Proceedings of the 27th International Conference on Computational Linguistics 2018* Aug (pp. 3890-3902).
- [17] Yin W, Schütze H. Convolutional neural network for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies 2015* (pp. 901-911).
- [18] Mahmoud A, Zrigui M. BLSTM-API: Bi-LSTM recurrent neural network-based approach for Arabic paraphrase identification. *Arabian Journal for Science and Engineering*. 2021 Apr;46(4):4163-74.
- [19] Corbeil JP, Ghadivel HA. Bet: A backtranslation approach for easy data augmentation in transformer-based paraphrase identification context. *arXiv preprint arXiv:2009.12452*. 2020 Sep 25.
- [20] Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*. 2019;32.

[21] Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942. 2019 Sep 26.