

Vicomtech at DA-VINCIS: Detection of Aggressive and Violent Incidents from Social Media in Spanish

Pablo Turón[†], Naiara Perez[†], Aitor García-Pablos, Elena Zotova and Montse Cuadros

SNLT group at Vicomtech Foundation, Basque Research and Technology Alliance (BRTA), Mikeletegi Pasealekua 57, Donostia/San-Sebastián, 20009, Spain

Abstract

This paper describes the participation of the Vicomtech NLP team in the DA-VINCIS shared task. This shared task is focused on mentions of violent events in Spanish tweets, and proposes two subtasks: first, detecting whether a violent incident is mentioned in a tweet; and, second, determining which type of violent event is being mentioned. We participated in this shared task with multiple systems built on Transformer-based models, which we fine-tuned on different versions of the provided data. Among others, we explored the impact of automatic data augmentation and relabelling. Further, we tested masking keywords during training as a means to avoid the models from overfitting these recurrent expressions. Our systems ranked in 2nd place in both tasks, with F1-scores of 77.32 and 52.86 respectively.

Keywords

Deep Learning, Transformers, Text Classification, Spanish, Online Social Networks

1. Introduction

Twitter is an established research platform that has been widely exploited by the Natural Language Processing (NLP) community to research, for instance, into hate speech and fake news detection [1]. The goal of the DA-VINCIS challenge [2] is to encourage research on a novel topic—the detection of mentions of violent events in Spanish-language tweets.

The organisers proposed two subtasks: first, detecting whether a violent incident is mentioned or not in a tweet; and second, determining which type of violent event is being mentioned (if any). From a technical perspective, both subtasks are text classification problems, where each tweet must be automatically associated to one or more labels. These working notes describe the participation of Vicomtech’s NLP team in the challenge. They are organised as follows: Section 2 provides an overview of the related work; Section 3 presents the data of the task and the techniques we tried to maximise its exploitation; Section 4 describes our systems; Section 5 reports and analyses our results; finally, Section 6 provides concluding remarks.

Detailed information about the shared task (e.g., about related work, the evaluation framework or the results of other participants) can be found in the organisers’ overview article [2].

IberLEF 2022, September 2022, A Coruña, Spain.


[†]These authors contributed equally.

✉ pturon@vicomtech.org (P. Turón); nperez@vicomtech.org (N. Perez); agarciap@vicomtech.org (A. García-Pablos); ezotova@vicomtech.org (E. Zotova); mcuadros@vicomtech.org (M. Cuadros)

🆔 0000-0002-5563-1120 (P. Turón); 0000-0001-8648-0428 (N. Perez); 0000-0001-9882-7521 (A. García-Pablos); 0000-0002-8350-1331 (E. Zotova); 0000-0002-3620-1053 (M. Cuadros)



© 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. Related work

DA-VINCIS is one more of the many shared tasks organised within the Iberian Languages Evaluation Forum (IberLEF) that proposes the problem of classifying tweets in the Spanish language. Some of the closest related shared tasks, which could be gathered under the umbrella theme of “harmful content”, include AMI [3], EXIST [4], MeOffendEs [5] and, above all, the MEX-A3T series [6, 7, 8]. These shared tasks have covered topics such as misogyny, aggressiveness and offensive language detection and classification throughout the last few years.

Statistical machine learning has been successfully applied for years to resolve problems of document classification. With the surge of interest in deep learning, these techniques were gradually replaced by the use of continuous word embeddings [9, 10, 11] and Convolutional Neural Networks (CNN) [12]. More recently, NLP has entered a new era, brought about by the convergence of powerful techniques such as word embedding contextualisation [13, 14], the Transformer architecture [15], and the improvement of the pre-train/fine-tune paradigm [16].

The most popular of these models are BERT [16] and RoBERTa [17]. Much of the success of this type of models lies on their capability to transfer the knowledge captured during pre-training to a new language, domain or task. Nevertheless, it has been repeatedly shown that better results are obtained when the models are pre-trained on the target language and domain [18, 19, among others]. For this reason, the last years have witnessed an explosion of publications of domain- and/or language-specific pre-trained models [20].

Social networks like Twitter are considered application domains worthy of special treatment, because the language employed by users in these platforms differs vastly from that of the commonly used sources to train generic models (e.g., Wikipedia or Common Crawl). Among the many BERT-like models learned from Twitter content, we must mention the pioneers BERTweet [21] and Twitter-roBERTa-base [22]. As is the default in NLP, both are English-specific. To the best of our knowledge, there exists no similar model for Spanish.

Fortunately, the Spanish language boasts currently a handful of competitive pre-trained models learned from a variety of monolingual content sources: BETO [23], SpanBERTa [24], BERTIN [25], MarIA [26], and RigoBERTa [27]. BETO is a BERT-based model, while BERTIN, SpanBERTa and MARIA are RoBERTa-like models. RigoBERTa follows the more recent DeBERTa [28] architecture. Finally, we must also mention the multilingual models Multilingual BERT [29] and IXAmBERT [30], whose pre-training included data in Spanish. Although multilingual models are generally less competitive than the monolingual ones, they offer the advantage of being better suited for cross-lingual learning.

3. Data

The dataset provided by the organisers consists of a set of 3,362 tweets for training, and 50 tweets as trial data. Table 1 (columns labelled as OFF) shows the distribution of these examples over the task’s categories. Subtask 1 is concerned with the categories Non-violent (N) versus Violent (V), while Subtask 2 provides one or more specific labels—Accident (A), Homicide (H), Theft (T) and/or Kidnap (K)—to Violent (v) tweets. That is, Subtask 1 is a binary classification problem, while Subtask 2 is a multi-label problem of 5 categories.

Table 1

Training and trial data quantification by category. OFF: official data split; RES: our split; RES+: our split, augmented; SIL: our split, automatically relabelled; SIL+: our split, automatically relabelled and augmented.

	Train					Trial		
	OFF	RES	SIL	RES+	SIL+	OFF	RES	SIL
Non-violent (N)	1,798	1,551	1,501	1,551	1,501	27	172	169
Violent (V), of which	1,564	1,345	1,395	2,230	2,258	23	147	150
Accident (A)	1,125	950	1,011	982	1,039	12	105	114
Homicide (H)	260	231	224	703	680	5	23	20
Theft (T)	179	158	153	475	461	5	22	17
Kidnap (K)	45	43	40	192	187	2	3	3
<i>Total</i>	3,362	2,896	2,896	3,781	3,759	50	319	319

As can be seen, the dataset presents a sharp imbalance. Moreover, the categories Homicide, Theft, and Kidnap are hardly represented in the trial dataset. This, paired with the facts that

- a) the official metrics of the shared task are macro-averaged, whereby mistakes involving the minority categories are more severely penalised, and
- b) during initial experiments, we spotted a number of inconsistencies in the gold labels, motivated us to try a series of techniques in automatic data cleaning and augmentation, among others, which we describe in the forthcoming sections.

3.1. Data pre-processing

That the pre-processing of tweets can have a considerable impact on system performance is a proven fact (see [31] and the related work therein). In this work, we applied basic pre-processing techniques aimed at reducing noise: removal of emojis, URLs, and hashtags. The result is illustrated in the following example:

Before: 🚨🔥🚌 #InformacionVial (14/09) #ACTUALIZACIÓN #Carabobo Troncal 5. Autopista del Sur. El accidente se trató de una Encava que venía sentido #Valencia #Tocuyito y por desperfecto mecánico en un neumático, saltó la isla quedando en sentido contrario de... <https://t.co/rjL5EF2sZ> <https://t.co/g4emPTDjbY>

After: (14/09) Troncal 5. Autopista del Sur. El accidente se trató de una Encava que venía sentido y por desperfecto mecánico en un neumático, saltó la isla quedando en sentido contrario de...

Of note, preliminary experiments with a number of more sophisticated data pre-preprocessing combinations did not yield improved results. Among others, we tested replacing emojis with descriptive words (e.g., “bus” instead of “🚌”) and segmenting mid-sentence hashtags based on casing (e.g., “Informacion Vial” instead of “#InformacionVial”). We expected the latter to be particularly beneficial because important content words are sometimes presented as hashtags. However, results on the trial data did not confirm our hypothesis.

3.2. Data resampling

After pre-processing the examples, we dropped 197 duplicate tweets (5.77% of the entire dataset). Then, in light of the small number of examples given as trial data, we computed new train and trial splits from the whole randomised, normalised dataset, keeping 90% for training and 10% for evaluation purposes. These new splits (see Table 1, columns Res) were used as starting point of all our subsequent experiments, including those reported in these working notes. All references to train and/or trial data should be hereafter interpreted in this manner too.

3.3. Data relabelling

Error analyses of preliminary experiments revealed a number of inconsistencies in the gold annotations of the train and trial data. Naturally, a small error rate is to be expected in any manually annotated corpus. Nevertheless, we considered it worthwhile to attempt to handle the noisy instances, as they also affected the least represented categories.

Our approach consisted simply in relabelling the train and trial data per the votes of 5 systems learned from the noisy training data. Specifically, we established that a given gold label in the training data should be corrected if at least 4 of the 5 systems agreed to do so. In the case of the trial data, we required that all the systems agreed, due to the sensitivity of this data split. Table 7 in Appendix A contains a few examples of these corrections. We henceforth refer to the version of the dataset resulting from this step as the “silver” dataset.

The new distribution of instances over categories can be consulted in Table 1 (columns SIL). Table 8 in Appendix A shows the number of automatic corrections by source (gold) and destination (silver) category. In total, 138 (4.77%) and 31 (9.72%) instances of the train and trial data were relabelled, respectively. The success rate of these corrections is analysed in Section 5.3 Error analysis.

The 5 systems employed in this procedure will be duly introduced throughout Section 4 System description, as they are themselves part of our official submissions.

3.4. Data augmentation

Our last effort in relation to the task’s corpus was to address the major imbalance between categories by augmenting the training data of the underrepresented ones. Among the multiple techniques that we tested in preliminary experiments (e.g., back-translation [32]), the most significant improvement was achieved by translating the tweets to languages other than Spanish and feeding them to a multilingual model (to be described in Section 4).

We used the pre-trained OPUS-MT models [33], available through HuggingFace’s Python library Transformers [34], to obtain different representations of the same tweet by translating them automatically from Spanish to English and German. Here is a real example:

- Spanish:** A través de un comunicado, el Eln confirmó el secuestro de dos militares en Arauca.
- English:** Through a statement, Eln confirmed the kidnapping of two soldiers in Arauca.
- German:** Durch eine Erklärung bestätigte der Eln die Entführung zweier Militärs in Arauca.

The process was applied to the minority categories Homicide, Theft, and Kidnap. In order to promote generation diversity, we computed 5 translations per target language and tweet, of which we chose randomly 1 translation per target language in the case of Homicide and Theft tweets, and up to 2 translations per target language in the case of Kidnap tweets. That is, we tripled the Homicide and Theft categories, and almost quintupled the Kidnap category. The result is shown in Table 1, columns RES+ and SIL+ (notice that we augmented both the original noisy dataset and our silver dataset). We also tweaked certain hyperparameters of the generation function, which can be consulted in Appendix B, Table 9a.

4. System description

Since Subtask 1 is contained within Subtask 2—Subtask 1 could be described as a simplification of Subtask 2, with attention only to the category Non-violent (N) versus the rest—, all our systems aim at resolving Subtask 2. Their output is then post-processed to obtain the expected output for Subtask 1. From this perspective, we developed two main types of systems:

- **1-step system** (Figure 1a): this system consists of one multi-label text classification model that makes predictions for the five categories of Subtask 2 (namely, Non-violent [N], Accident [A], Theft [T], Kidnap [K], and Homicide [H]) in one pass.
- **2-step system** (Figure 1b): this system requires up to two passes per input to produce an output. The first step consists of a binary classifier that decides whether a violent incident is mentioned in a tweet or not. That is, it filters all Non-violent tweets. In the second step, a multi-label classification model assigns the finer-grained labels Accident, Theft, Kidnap, and/or Homicide to Violent tweets.

The classifiers that conform the systems are all Transformer neural networks [15]. We have developed multiple variants, which differ in aspects such as the base pre-trained language model or the data from which the classifiers were learned, among others. In addition, we have built a number of ensembles from the system variants, in order to assess whether the knowledge acquired by the different models complements each other.

In what follows, we present the core architecture of the classifiers (Section 4.1), and explain how the outputs of the different system types are handled (Section 4.2). Finally, we report the implementation details and training setup of the submitted system variants (Section 4.3).

4.1. Architecture

The three classification models introduced above follow the same Transformer-based standard architecture for text classification: a BERT encoder, whose output for the special token at the start of each sequence (e.g., BERT’s [CLS] or RoBERTa’s <s>) is pooled and fed to a dropout layer, followed by a dense linear layer that produces the logits for the target categories. The binary classifier is fine-tuned with the cross-entropy loss, while the multi-label classifiers use the binary cross-entropy with logits loss (BCEWithLogitsLoss).

In inference, the binary model predicts the most likely category from the logits (Figure 2b). In the case of the multi-label classifiers (Figures 2a and 2c), the sigmoid function is applied to

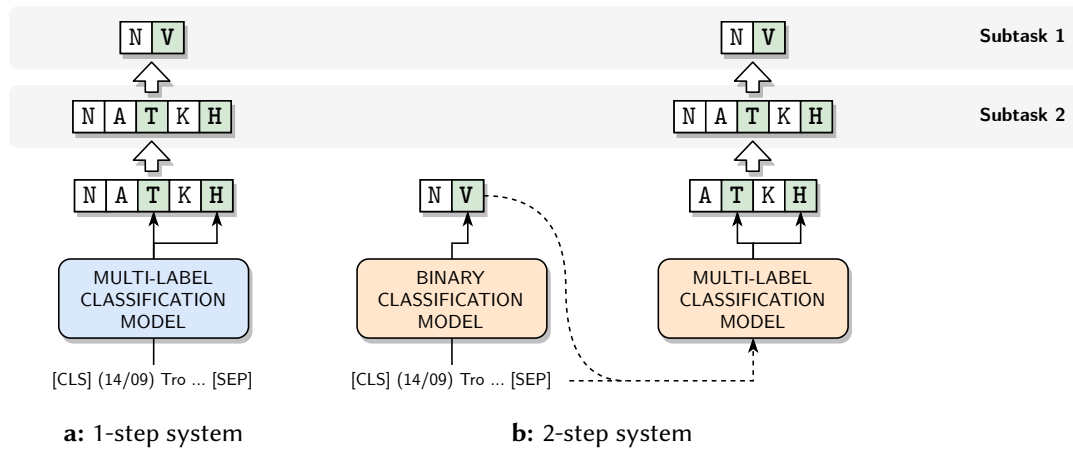


Figure 1: Inference flowchart diagrams

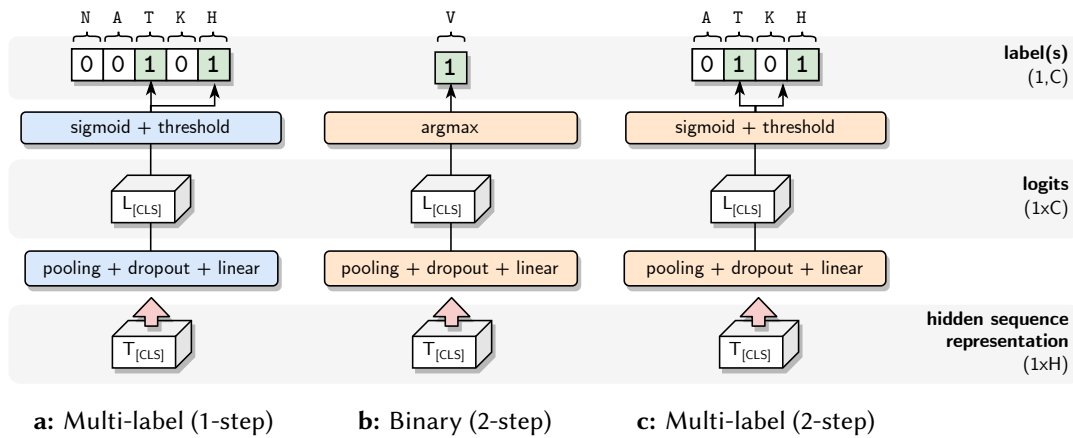


Figure 2: Diagram of the classifier models' output layers. All the trained models consist of a Transformer encoder followed by a binary or multi-label sequence classification head. 1-step systems (Figure 1a) consist of one multi-label model of the 5 categories of Subtask 2 (2a), while 2-step systems (Figure 1b) combine a binary model (2b) and a multi-label model of the 4 Violent subcategories (2c).

the logits, after which the categories whose probabilities exceed a given threshold, empirically set to 0.5, are taken as positive labels.

In all cases, the input tweets to be passed through the encoder are appropriately tokenised with the corresponding pre-trained tokeniser, wrapped between the special tokens that signal the start and end of the sequence, and padded to a maximum length of 256 tokens.

4.2. Output handling

The output of the systems had to be post-processed in order to submit the predictions as expected to the challenge. The post-processing consists of 2 phases, as illustrated in Figure 1:

Table 2
Prediction post-processing rules for Subtask 2

Prediction	Well-formed	Rule	Example	Systems to which the rule applies
N	Yes	n/a	$N \rightarrow N$	1-step and 2-step (binary model)
A, T, K, and/or H	Yes	n/a	$T, H \rightarrow T, H$	1-step and 2-step (multi-label model)
$N + A, T, K, \text{ and/or } H$	No	Discard N	$N, A \rightarrow A$	1-step
\emptyset	No	Add N	$\emptyset \rightarrow N$	1-step and 2-step (multi-label model)

First, we obtain well-formed predictions for Subtask 2. We define a well-formed prediction as that which consists of *a*) the label Non-violent (N)—and only that label—, or *b*) any combination of the other 4 categories for violent incidents. With this definition in mind, four possible scenarios arise, which we summarise in Table 2. As can be seen, the post-processing consists in fixing ill-formed outputs by adding or removing the label N as necessary.

Second, we compute the final predictions for Subtask 1 from the post-processed results for Subtask 2. We simply return the label Non-violent if that was the result for Subtask 2, and Violent otherwise.

4.3. Implementation details and training setup

Our participation in the shared task includes a series of variants of the above explained systems. In addition to the type of system itself—1-step or 2-step—, the variables that we considered in our final submissions (to be listed below) were the following four:

Hyperparameters In general, we used the default hyperparameters of the Transformers library, as given by the TrainingArguments class [35]. We did experiment with a few commonly tweaked hyperparameters, such as the batch size and the learning rate. Our final submissions include 2-step system variants trained with 2 different sets of hyperparameters, and a 1-step system trained with a third set of hyperparameters. These combinations can be consulted in Appendix B, Table 9b.

Keyword masking In order to prevent models from learning to represent each label by its keywords, we experimented with masking them randomly during training with a controlled masking probability rate. The masking consisted in replacing each token of a keyword with the mask token of the pre-trained tokeniser being used (e.g., BERT’s [MASK]). Consider the following example, where the keywords “asesinato” and “presunto” have been masked:

Original: Autoridades reportaron el asesinato de un presunto miembro de la [...]
Tokenised: Autoridad ##es reportar ##on el asesinato de un presun ##to miembro de la [...]
Masked: Autoridad ##es reportar ##on el [MASK] de un [MASK] [MASK] miembro de la [...]

The keywords were estimated using the Term Frequency–Inverse Document Frequency (TF-IDF) weights of the words in the normalised training dataset, after removing stopwords and converting the text to lowercase. Thus, we calculated the TF-IDF value of each word per violent

Table 3

Relation of systems and their training details: training and trial data, base pre-trained model, key mask (KM) rate, and hyperparameters (HP).

	Type	Binary model				Multi-label model			
		Data	Base	KM	HP	Data	Base	KM	HP
R1	1-step	-	-	-	-	RES	BETO	-	9bA
R2	2-step	RES	BETO	-	9bB	RES	BETO	0.60	9bB
R3	2-step	RES	BETO	-	9bC	RES	BETO	-	9bC
R4	2-step	RES	MarIA	-	9bB	RES	MarIA	-	9bB
R5	2-step	RES+	mBERT	-	9bB	RES+	mBERT	-	9bB
S1	1-step	-	-	-	-	SIL	BETO	-	9bA
S2	2-step	SIL	BETO	-	9bB	SIL	BETO	0.60	9bB
S3	2-step	SIL	BETO	-	9bC	SIL	BETO	-	9bC
S4	2-step	SIL	MarIA	-	9bB	SIL	MarIA	-	9bB
S5	2-step	SIL+	mBERT	-	9bB	SIL+	mBERT	-	9bB

event category against the rest of the data. Then, we selected the words whose TF-IDF value exceeded a given threshold (empirically set to 0.10) in any of the categories, obtaining a total of 25 keywords. Experiments with a range of masking probability rates pointed to 0.60 as the most beneficial rate.

Base pre-trained model The pre-trained language models employed in our experiments were the monolingual Spanish models BETO_{Base} Cased [23] and MarIA RoBERTa_{Base} [26]. We also used Multilingual BERT_{Base} Cased (mBERT) [29], which we introduced in our experiments to better leverage the augmented multilingual datasets.

Dataset As explained in Section 3, we prepared four versions of the training dataset, all of which were used to create system variants: we trained monolingual classifiers with the given Spanish tweets, and multilingual classifiers with the augmented datasets; furthermore, we trained an analogous variant with the silver data for each of the system variants trained on the original noisy data.

The final trained systems are summarised in Table 3. We developed a total of 10 systems that differ among each other in one or more of the above introduced variables. The systems R1-5 were the ones used to produce the silver dataset, as explained in Section 3.3.

Finally, we also computed ensemble results from various combinations of the 10 systems. The ensembles consist simply in the majority vote per each category of Subtask 2 (the output post-processing rules explained in Section 4.2 apply here as well). They are as follows:

- **E1:** an ensemble of R1-5 (the systems trained on the original noisy data)
- **E2:** an ensemble of S1-5 (the systems trained on the silver data)
- **E3:** an ensemble of R1, R2, R3, S4, and S5 (the best system, when compared pairwise on the trial results, of the analogous R and S variants)

This makes a total of 13 systems. As the organisers allowed for 5 runs per task, we submitted the predictions of the 3 ensembles to both tasks, and chose the best two standalone systems in the trial data (one original and one silver) to complete the submissions: R3 and S5 for Subtask 1, and R2 and S2 for Subtask 2.

All the systems were implemented in Python 3.9 with HuggingFace’s Transformers library [34] version 4.18.0. The models were trained during a maximum of 50 epochs, with one checkpoint saved per epoch. The criterion to choose the best checkpoint was the macro F1-score. Each model took ~30-60 minutes to train, depending on the batch size, in one NVIDIA GeForce RTX 2080 GPU with 11GB of memory.

5. Results

The results of the 13 systems developed for the shared task are shown in Table 4. As is standard in classification tasks, the results are measured in terms of precision (P), recall (R) and F1-score (F1). Following the official shared task definition, Subtask 1 is concerned with the metrics for the category Violent, while Subtask 2 shows the macro-average of all the categories.

We report results on the trial and test datasets. Since the gold labels of the test dataset have not been released at the time of writing these working notes, the test results only include the 5 officially submitted systems per task. In addition, we provide the results of the organiser’s baseline [2] and of the best participant as reference.

5.1. Subtask 1: Violent event identification

In the binary subtask we ranked in second position with an F1-score of 77.32, close to the winner, with a 2-step system of BETO models fine-tuned on the original noisy data and the hyperparameter set that included a bigger batch size and a lower learning rate (see column C in Table 9b). This system was also the best among the standalone systems in the trial evaluation, although it was surpassed by some of the ensemble results.

In general, we observe a clear disadvantage of the 1-step systems with respect to the 2-step systems in the trial data. This is to be expected, because the 1-step system was learnt directly from the data for Subtask 2, which is more complex.

No such general remark can be made with respect to the impact of the silver dataset: some systems worsen (namely, S1, S2, and R3), while others achieve slightly better results (S4 and S5) when compared to their analogous variants R1-5. Interestingly, the system that most benefits from the silver data is the one built on the augmented training data (S5), where the augmentation was performed after correcting the labels; that is, where we augmented the least represented categories based on the automatic labels. This system increases precision considerably while maintaining the recall metrics. It also achieves the best precision of the shared task in the trial data, but the lower recall renders it the worst system submitted to test. It is also noteworthy that the ensembles built partly or totally on S1-5 (i.e., E2 and E3) manage to surpass the ensemble E1, built on the votes of R1-5, on both the trial and test data.

Overall, we observe better recall than precision metrics in the trial data, and a reverse pattern in the test data. Having no access to the labelled test dataset at the moment of writing these

Table 4

Official results, including the organiser’s baseline (Base) and the results of the best participant (Best). The best results among our systems are highlighted in boldface.

	Subtask 1						Subtask 2					
	Trial (RES)			Test			Trial (RES)			Test		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
R1	79.14	74.83	76.92	-	-	-	68.81	58.94	62.95	-	-	-
R2	77.64	85.03	81.17	-	-	-	71.48	62.07	65.53	48.88	50.47	49.30
R3	77.78	85.71	81.55	81.28	73.73	77.32	62.65	64.08	62.92	-	-	-
R4	75.32	80.95	78.03	-	-	-	68.41	52.63	56.78	-	-	-
R5	71.35	86.39	78.15	-	-	-	56.64	57.69	55.31	-	-	-
S1	72.90	76.87	74.83	-	-	-	65.94	56.18	59.72	-	-	-
S2	79.73	80.27	80.00	-	-	-	72.32	59.97	64.63	48.33	51.82	48.79
S3	76.54	84.35	80.26	-	-	-	59.64	62.07	60.51	-	-	-
S4	85.71	73.47	79.12	-	-	-	74.54	51.89	58.05	-	-	-
S5	75.60	86.39	80.63	82.08	69.99	75.55	57.71	60.48	58.24	-	-	-
E1	78.12	85.03	81.43	81.12	73.05	76.88	74.23	63.00	67.39	50.68	55.58	52.21
E2	81.46	83.67	82.55	80.64	73.68	77.01	76.01	60.81	66.16	51.75	54.59	52.86
E3	81.05	84.35	82.67	80.32	74.26	77.17	74.38	61.38	66.33	50.35	52.86	51.31
Base	-	-	-	74.00	82.00	78.17	-	-	-	57.00	46.00	49.81
Best	-	-	-	80.32	75.04	77.59	-	-	-	55.00	56.42	55.43

working notes, we can only speculate that our systems have not been able to generalise well to the new instances of tweets with mentions of violent incidences.

Finally, it must be noted that none of the participants managed to reach the baseline.

5.2. Subtask 2: Violent event category recognition

In the case of multi-label subtask, the submission that achieved the highest F1-score (52.86) was an ensemble of all our systems trained with the silver data (E2), obtaining again the second place in the official shared task ranking. In this case, however, the advantage of the subtask’s winner with respect to our best result is more substantial. Overall, similar, general observations can be made as those of Subtask 1, but for three major differences:

First, that the 1-step systems (R1 and S1) are at least as competitive as the 2-step systems in this subtask’s trial data. They do not yield the best results, but do not lag behind either.

Second, that the trial results show a clear positive impact of the random keyword masking. This training strategy has resulted in a marked increase of precision metrics with respect to the other standalone systems, making R2 and S2 our the top standalone performers by a margin of 2-4 F1-score points.

Finally, that all the ensembles managed to beat the standalone systems R2 and S2, and the baseline too (whereas neither R2 nor S2 did). Further, the best ensemble predictions, E2, were achieved with the votes of the systems trained on the silver data (S1-5).

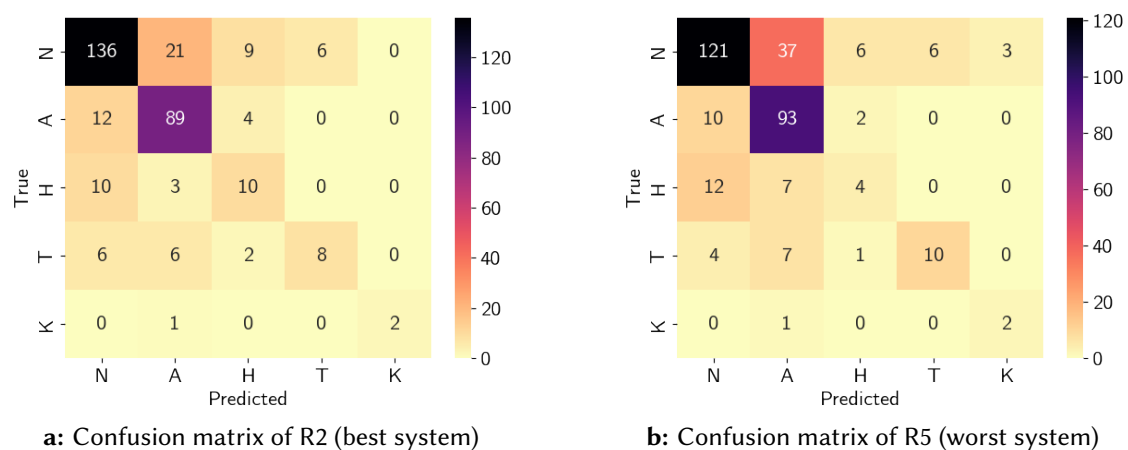


Figure 3: Confusion matrices on the trial (RES) dataset of Subtask 2

5.3. Error analysis

In what follows, we analyse the errors of the best and worst standalone system in Subtask 2. Since the gold labels of the test dataset have not been released at the time of writing these working notes, we analysed the predictions on the trial dataset. Figures 3a and 3b, show, respectively, the confusion matrices of the best (R2) and worst (R5) systems.

We observe in both cases that many of the errors involve the majority class Non-violent (N), both as false positive and false negative predictions. (This is actually the case of all the tested systems; see examples in Table 5.) The main difference between the two systems is that R2 commits less false positive errors in the Accident (A) category. Furthermore, R2 does not make false positive predictions of the category Kidnap (K), while R5 makes 3 errors of this type. While 3 errors out of 319 predictions could be thought to be marginal, the fact that Subtask 2 was measured in terms of macro-average metrics makes the evaluation extremely sensitive to even the smallest number of errors, if said errors involve an underrepresented category. In the case of R5, 3 false positive errors in the Kidnap category results in a precision of 40.00 points for said category. In turn, the macro-averaging translates it into a penalisation of 12.00 points in the total precision of the system, which explains virtually all the difference between R2 and R5.

We conclude the error analysis with a report on the silver dataset. We have examined manually all the automatically corrected instances to quantify *a*) how many of the relabelled instances were actually inconsistently labelled in the gold standard; and, *b*) how many of the relabelled instances have been assigned a correct label. We acknowledge that this exercise was carried out based solely on the definition of the task provided by the organisers, and that it is possible that some of our judgements contravene the annotation policy that the organisers employed to create the gold standard corpus.

The result of this analysis is shown in Table 6. The numbers are broken down by category (that is, by the original category of the relabelled instances). In total, 138 (4.77%) and 31 (9.72%) instances of the train and trial data were relabelled, respectively. The relabelled instances

Table 5

Trial examples where all the tested systems committed the same error




	Tweet	Gold	Pred
1	 (16:39) A partir de este momento las líneas 1 y 2 de buses  normalizan su servicio comercial entre Universidad de Medellín y Aranjuez, luego de operar parcialmente por un accidente de tránsito.	A	N
2	Acusan a mayor de la Policía por muerte de joven de 19 años en el paro. Audiencia preparatoria por homicidio de Santiago Murillo seguirá en noviembre. Le contamos los detalles 	H	N
3	#Nayarit es el 4to estado con menor incidencia del delito de #Secuestro en lo que va del sexenio, de 14 casos que se reportaron 10 se registraron en #Tepic . @Wallacelsabel @AntonioEcheG @MeganoticiasTEP @NayaritFiscalia	K	N
4	Tras denuncia recibida, en la @GNBMedianiaANZ fue detenido ciudadano por robo con arma blanca #14Sep #GNB #FANB #CEOFANB #GNB2DOCMDT-EDCR529 #AlertasContraEISabotaje #FANByPuebloVenceremos #GNBESpueblo @vladimirpadrino @libertad003 @GNB_Anzoategui	T	N

Table 6

Analysis of silver annotations: number of relabelled or changed examples (C) and, of those, accurately spotted gold errors (GE) and accurately made silver corrections (SC). Note that the percentage of C is computed over the entire dataset, while the percentage of GE and SC is computed over C.

	Train			Trial				
	C		GE/C	SC/C	C		GE/C	SC/C
	#	%	%	%	#	%	%	%
Non-violent (N)	94	6.06	100.00	100.00	12	6.98	100.00	100.00
Violent (V), <i>of which</i>	44	3.27	90.91	47.73	19	12.93	84.21	52.63
Accident (A)	27	2.84	92.59	44.44	10	9.52	80.00	30.00
Homicide (H)	8	3.46	87.59	37.50	5	10.64	80.00	60.00
Theft (T)	9	5.70	100.00	66.67	6	27.27	100.00	100.00
Kidnap (K)	3	6.98	66.67	66.67	0	0.00	-	-
<i>Total</i>	138	4.77	97.10	83.33	31	9.72	90.32	70.97

involved presumed gold errors respectively 97.10% and 90.32% of the time, and the assigned label was correct in 83.33% and 70.97% of the relabelled instances.

Note that, since the corrections on the train data were carried out with models trained on the train data itself, it is likely that the gold error rate is actually higher in that partition. With all, our attempt at automatically detecting and correcting the noisy data did not have a remarkable impact on the results, as explained in the previous section.

6. Conclusions and future work

In these working notes we have described the participation of the Vicomtech NLP team in the DA-VINCIS shared task. The organisers proposed a binary classification subtask (Subtask 1)

and a multi-label classification subtask (Subtask 2).

We have developed systems to resolve the two subtasks using the Transformer-based models BETO, MarIA, and Multilingual BERT. BETO [23] yielded a better performance overall.

The developed systems were of two types: one solves the tasks in a single inference step, while the other approaches the tasks in a 2-step fashion (the first classifier determines whether a tweet mentions a violent event; if so, the second classifier emits finer-grained violence categories). In our experiments, the 2-step systems outperformed the former consistently.

In addition, we have computed ensemble predictions following the majority vote algorithm. The ensembles outperformed the standalone systems in the trial evaluations of the two subtasks, but only in Subtask 2 when evaluated with the test data.

Furthermore, we have explored several data curation techniques. First, we have resampled the dataset in order to obtain a bigger trial split, so as to be able to choose the best checkpoints more reliably. Then, we have generated three additional dataset versions: *i*) a silver dataset, where some of the examples have been automatically relabelled, *ii*) a dataset automatically augmented with machine-translated examples, and *iii*) a dataset that combined the previous two techniques. The impact of these modifications to the training and trial data was not systematic. Notably, the ensemble of the systems trained with the silver dataset was the one to obtain the best scores in Subtask 2.

Finally, we have tested masking keywords during training as a means to avoid the models from overfitting these recurrent expressions. This strategy has proven remarkably beneficial in Subtask 2, where the systems trained thus surpassed all the others by a substantial margin.

Through all these experiments we have achieved the second place in both subtasks of the competition. In Subtask 1, our best system was the 2-step system built on BETO models and fine-tuned on gold labels. It has obtained an F1-score of 77.32 (0.27 points below the best participant’s model, and 0.85 below the baseline model). In Subtask 2, our best system consisted of an ensemble of 5 models fine-tuned with the silver data, which achieved a macro-average F1-score of 52.86 points (2.57 below the winner and 3.05 above the baseline).

In conclusion, the shared task has proposed an interesting and relevant problem, with many practical applications in real life. The main challenges we encountered had to do with the competition’s data being unbalanced and noisy, a very typical scenario when implementing real-world applications [36, 37]. The basic strategies that we implemented to address these problems did not prove remarkably fruitful. Thus, we believe that future work should explore more sophisticated methods for robust training, both through improved learning architectures and/or better data curation techniques.

Acknowledgments

This work was funded by the APPRAISE project – fAcilitating Public Private secuRity operAtors to mitigate terrorism Scenarios against soft targEts, with the support of the European Commission and the Horizon 2020 Program, under Grant Agreement No. 101021981.

References

- [1] D. Antonakaki, P. Fragopoulou, S. Ioannidis, A survey of Twitter research: Data model, graph structure, sentiment analysis and attacks, *Expert Systems with Applications* 164 (2021) 114006.
- [2] L. J. Arellano, H. J. Escalante, L. Villaseñor-Pineda, M. Montes y Gómez, F. Sanchez-Vega, Overview of DA-VINCIS at IberLEF 2022: Detection of aggressive and violent incidents from social media in Spanish, *Procesamiento del Lenguaje Natural* 69 (2022) TBP.
- [3] E. Fersini, P. Rosso, M. Anzovino, Overview of the task on automatic misogyny identification at IberEval 2018, in: *Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018)*, CEUR Workshop Proceedings, Sevilla, Spain, 2018, pp. 214–28.
- [4] F. Rodríguez-Sánchez, J. C. de Albornoz, L. Plaza, J. Gonzalo, P. Rosso, M. Comet, T. Donoso, Overview of EXIST 2021: sEXism Identification in Social neTworks, *Procesamiento del Lenguaje Natural* 67 (2021) 195–207.
- [5] F. M. Plaza-del Arco, M. Casavantes, H. J. Escalante, M. T. Martín-Valdivia, A. Montejo-Ráez, M. Montes-y Gómez, H. Jarquín-Vásquez, L. Villaseñor-Pineda, Overview of MeOffendEs at iberlef 2021: Offensive language detection in Spanish variants, *Procesamiento del Lenguaje Natural* 67 (2021) 183–194.
- [6] M. Á. Álvarez-Carmona, E. Guzmán-Falcón, H. J. Escalante, L. Villaseñor-Pineda, V. Reyes-Meza, A. Rico-Sulayes, Overview of MEX-A3T at IberEval 2018: Authorship and Aggressiveness Analysis in Mexican Spanish Tweets, in: *Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018)*, CEUR Workshop Proceedings, Sevilla, Spain, 2018, pp. 74–96.
- [7] M. E. Aragón, M. Álvarez-Carmona, M. Montes-y Gómez, L. Escalante, Hugo Jair Villaseñor-Pineda, D. Moctezuma, Overview of MEX-A3T at IberEval 2019: Authorship and Aggressiveness Analysis in Mexican Spanish, in: *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019) co-located with 35th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019)*, CEUR Workshop Proceedings, Bilbao, Spain, 2019, pp. 478–494.
- [8] M. E. Aragón, H. J. Jarquín-Vásquez, M. Montes-y Gómez, H. J. Escalante, L. V. Pineda, H. Gómez-Adorno, J. P. Posadas-Durán, G. Bel-Enguix, Overview of MEX-A3T at IberEval 2020: Fake news and Aggressiveness Analysis in Mexican Spanish, in: *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020) co-located with 36th Conference of the Spanish Society for Natural Language Processing (SEPLN 2020)*, CEUR Workshop Proceedings, Málaga, Spain, 2020, pp. 222–235.
- [9] T. Mikolov, W.-t. Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013)*, Association for Computational Linguistics, Atlanta, GA, USA, 2013, pp. 746–751.
- [10] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: *Proceedings of the 26th International*

- Conference on Neural Information Processing Systems (NIPS 2013) - Volume 2, Curran Associates Inc., Lake Tahoe, NV, USA, 2013, pp. 3111–3119.
- [11] J. Pennington, R. Socher, C. Manning, GloVe: Global vectors for word representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543.
- [12] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1746–1751.
- [13] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2018): Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, LA, USA, 2018, pp. 2227–2237.
- [14] A. Akbik, D. Blythe, R. Vollgraf, Contextual string embeddings for sequence labeling, in: Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018), Association for Computational Linguistics, Santa Fe, NM, USA, 2018, pp. 1638–1649.
- [15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS 2017), Curran Associates Inc., Long Beach, CA, USA, 2017, pp. 6000–6010.
- [16] 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 (NAACL 2019): Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, MN, USA, 2019, pp. 4171–4186.
- [17] 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 (2019). [arXiv:1907.11692](https://arxiv.org/abs/1907.11692).
- [18] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451.
- [19] R. Agerri, I. San Vicente, J. A. Campos, A. Barrena, X. Saralegi, A. Soroa, E. Agirre, Give your text representation models some love: the case for Basque, in: Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020), European Language Resources Association, Marseille, France, 2020, pp. 4781–4788.
- [20] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang, W. Han, M. Huang, Q. Jin, Y. Lan, Y. Liu, Z. Liu, Z. Lu, X. Qiu, R. Song, J. Tang, J.-R. Wen, J. Yuan, W. X. Zhao, J. Zhu, Pre-trained models: Past, present and future, *AI Open* 2 (2021) 225–250.
- [21] D. Q. Nguyen, T. Vu, A. Tuan Nguyen, BERTweet: A pre-trained language model for English tweets, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 9–14.

- [22] F. Barbieri, J. Camacho-Collados, L. Espinosa Anke, L. Neves, TweetEval: Unified benchmark and comparative evaluation for tweet classification, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020.
- [23] J. Cañete, G. Chaperon, R. Fuentes, J. Pérez, Spanish pre-trained BERT model and evaluation data, in: Proceedings of the Practical ML for Developing Countries Workshop (PML4DC 2020) at the 8th International Conference on Learning Representations (ICLR 2020), Addis Ababa, Ethiopia, 2020, pp. 1–9.
- [24] C. Tran, Pretrain RoBERTa for Spanish from scratch and perform NER on Spanish documents, 2020. URL: <https://github.com/chriskhanhtran/spanish-bert>, accessed: 2022-06-02.
- [25] J. de la Rosa, E. G. Ponferrada, M. Romero, P. Villegas, P. González de Prado Salas, M. Grandury, BERTIN: Efficient pre-training of a Spanish Language Model using perplexity sampling, *Procesamiento del Lenguaje Natural* 68 (2022) 13–23.
- [26] A. Gutiérrez-Fandiño, J. Armengol-Estapé, M. Pàmies, J. Llop-Palao, J. Silveira-Ocampo, C. Pio Carrino, C. Armentano-Oller, C. Rodríguez-Penagos, A. Gonzalez-Agirre, M. Villegas, MarIA: Spanish Language Models, *Procesamiento del Lenguaje Natural* 68 (2022) 39–60.
- [27] A. Vaca Serrano, G. García Subies, H. Montoro Zamorano, N. Aldama García, D. Samy, D. Bencur Sánchez, A. Moreno Sandoval, M. Guerrero Nieto, Á. Barbero Jiménez, RigoBERTa: A state-of-the-art language model for Spanish (2022). *arXiv:2205.1023*.
- [28] P. He, X. Liu, J. Gao, W. Chen, DeBERTa: decoding-enhanced BERT with disentangled attention, in: Proceedings the 9th International Conference on Learning Representations (ICLR 2021), Vienna, Austria, 2021, pp. 1–21.
- [29] J. Devlin, S. Petrov, Multilingual models, 2018. URL: <https://github.com/google-research/bert/blob/master/multilingual.md>, accessed: 2022-06-02.
- [30] A. Otegi, A. Agirre, J. A. Campos, A. Soroa, E. Agirre, Conversational Question Answering in low resource scenarios: A dataset and case study for Basque, in: Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020), European Language Resources Association, Marseille, France, 2020, pp. 436–442.
- [31] U. Naseem, I. Razzak, P. W. Eklund, A survey of pre-processing techniques to improve short-text quality: A case study on hate speech detection on Twitter, *Multimedia Tools and Applications* 80 (2021) 35239–35266.
- [32] R. Sennrich, B. Haddow, A. Birch, Improving Neural Machine Translation models with monolingual data, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 86–96.
- [33] J. Tiedemann, S. Thottingal, OPUS-MT – Building open translation services for the World, in: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation (EAMT), European Association for Machine Translation, Lisbon, Portugal, 2020, pp. 479–480.
- [34] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art Natural Language Processing (2019). *arXiv:1910.03771*.
- [35] HuggingFace, TrainingArguments, 2022. URL: <https://huggingface.co/docs/transformers/>

Table 7
Examples of automatically relabelled (silver) tweets

Tweet	Gold	Silver
Los colombianos pueden tener la tranquilidad que sus @GaulaMilitares se encuentran custodiando toda la geografía colombiana para prevenirlos de los delitos del secuestro y la extorsión. Denuncia en la #Línea147 a cualquier hora. Siempre habrá alguien que te ayudará.	A	N
▲ Un chofer de una pipa por evadir una Vaca se impactó con dos vehículos provocando la muerte de cuatro personas, la noche del miércoles ▲ El accidente ocurrió en la carretera #Monterrey - #Reynosa kilómetro 145. #GANDIAGANOTICIAS #RedesSociales #LasNoticias #Tamaulipas	N	A
#ELBRAVO #Noticias #Internacional Policía del Capitolio sancionará a 6 agentes por su conducta tras asalto al Congreso de EU > https://t.co/HAavTY43QH	T	N
Hoy comencé mi vida de nuevo 🕒😄, porque por fin pude levantarme después de mis cuatro operaciones en el Chicho Fábrega, son 4 meses de mi accidente en ese tiempo conocí a mis verdaderos amigos, pero quiero agradecer al DR Alvino De León por creer que podía levantarme. ❤️🇲🇽🔥	H	N
#Regiones El exfutbolista Macnelly Torres fue víctima de un atraco en Medellín. El robo ocurrió cuando se encontraba con unos amigos en su negocio en el barrio Robledo. #InseguridadEs #LaFmTeCuida @lafm https://t.co/hABbiQ3E4m	N	T
● En el poblado de #SanLorenzoTalmimilolpan, en #SanJuanTeotihuacan, #EstadoDeMexico, autoridades reportaron el asesinato a balazos de un presunto miembro de la @GN_MEXICO_ ●	N	H
No es meme. La cara de Bottas cuando le mostraron el accidente. Dijo: "lamentable", y se sonrió.	K	N
#Regiones Un hecho de intolerancia sucedió en el barrio Las Palmas, en Neiva. Allí se presentó el homicidio Edward Rodríguez, de 20 años, quien fue sorprendido dentro de su casa, por el papá de una menor de edad, que al parecer era su novia. @lafm #LaFmTeCuida #MiSeleccionHoy	A	H
Sucesos Empleado de la morgue robó prótesis mamarias del cadáver de una mujer ➡ #elsiglocomve https://t.co/yobNuJetuQ	H,T	T

main_classes/trainer#transformers.TrainingArguments, accessed: 2022-06-02.

- [36] J. M. Johnson, T. M. Khoshgoftaar, Survey on deep learning with class imbalance, Journal of Big Data 6 (2019) 1–54.
- [37] H. Song, M. Kim, D. Park, Y. Shin, J.-G. Lee, Learning from noisy labels with deep neural networks: A survey, IEEE Transactions on Neural Networks and Learning Systems (2022) 1–19.

A. Silver corrections

Table 7 contains examples of tweets that were automatically relabelled. See Section 3.3 for a detailed explanation of this procedure. As can be seen from the examples, not all the relabelled tweets were wrong originally, nor was the automatically reassigned label always correct.

Table 8
Breakdown of silver instances by the original gold category

a: Train			b: Trial		
Gold	Silver	#	Gold	Silver	#
Non-violent	Accident	87	Non-violent	Accident	12
Accident	Non-violent	25	Accident	Non-violent	8
Theft	Non-violent	8	Homicide	Non-violent	1
Homicide	Non-violent	5	Theft	Accident	4
Non-violent	Theft	5	Accident	Homicide	2
Non-violent	Homicide	2	Homicide	Accident	2
Kidnap	Non-violent	2	Homicide, Theft	Theft	1
Accident, Homicide	Non-violent	2	Homicide, Theft	Accident	1
Homicide, Kidnap	Non-violent	1			
Homicide, Theft	Non-violent	1			
Total		139	Total		31

Table 9
Hyperparameters for some of the automated process involved in different phases of the experimentation

a: Generation hyperparameters		b: Classification model hyperparameters			
Hyperparameter	Value	Hyperparameter	A	B	C
Number of beams	6	Batch size	32	8	16
Number of beam groups	3	Sequence length	256	256	256
Temperature	1.2	FP16	true	true	true
Repetition penalty	0.5	Learning rate	3e-05	5e-05	2e-05
No repeat n-gram size	3	Warm-up ratio	0.1	0.1	0.1
Diversity penalty	0.5	Weight decay	0.1	0.0	0.0
Top K	0	Seed	42	42	42

Eventually, however, the obtained silver dataset contained presumably less errors than the gold standard (see Section 5.3 Error analysis). Table 8 shows the breakdown of automatic corrections by source (gold) and destination (silver) category.

B. Hyperparamters

Table 9 contains the most relevant hyperparameters that we used during our experiments. Table 9a shows the hyperparameters for the generation of machine-translated examples (see Section 3.4 Data augmentation). Table 9b shows the three hyperparameter sets involved in the training of the developed tweet classification systems (see Section 4.3 Implementation details and training setup). Notice that we only report the hyperparameters whose values differ from the default given by Huggingfaces’s Python library Transformer (version 4.18.0). The default values can be consulted in their extensive documentation.