Detection of Violent Events in Social Media: DA-VINCIS 2023

Braulio Hernández-Minutti[1], Jesus-Alejandro Olivares-Padilla[1],
Ricardo Valerio-Carrera[1] and Omar Juárez Gambino[1,*]

[1]*Instituto Politécnico Nacional (IPN) - Escuela Superior de Cómputo (ESCOM), J.D. Batiz e/ M.O. de Mendizabal s/n, Mexico City, 07738, Mexico*

Abstract

In this paper, we describe the participation of the ESCOM NLP group in the DA-VINCIS 2023 task. The task propose to detect violent events in tweets. Two tracks were defined: identification of violent events and recognition of categories of violent events. We trained machine learning methods and proposed an ensemble schema which boost the performance. Our best model ranked 11th and 9th in the first and second tasks.

Keywords

Violent event detection, Multimodal information, Machine Learning,

1. Introduction

Violence is perceived as a severe problem in society. Some consequences of violence are depression, anxiety, and post-traumatic stress disorder. Detecting these events helps to prevent the terrible effects mentioned above.

Social networks allow users to communicate quickly and effectively. The media have exploited these features to publish news and increase their audience. Twitter has become the media's favorite for posting news; nearly 85% of trending topics are reported to be headlines or persistent news stories [1]. Therefore, it has become relevant to detect violent events that are reported through this media.

Violent behaviour in social networks has been studied from different aspects. In [2], authors proposed some methods to identify violent radicals from non-violent considering user's profiles. Hate speech is another facet of social media violence. Transfer learning has been used for automatic detection of these events [3]. In [4], violent events reported on Twitter were used to train methods for automatic detection. Pretrained transformers, ensembles, and multi-task learning were the main approaches to tackling the task.

Considering the impact of violence and the importance of its early detection, a task was co-located at the IberLEF 2023 evaluation forum [5]. DA-VINCIS 2023 [6] proposes that participants create solutions that automate detecting violent events on Twitter.

IberLEF 2023, September 2023, Jaén, Spain

[*] Corresponding author.

bhernandezm1902@alumno.ipn.mx (B. Hernández-Minutti); jolivaresp1600@alumno.ipn.mx (J. Olivares-Padilla); rvalerioc1600@alumno.ipn.mx (R. Valerio-Carrera); jjuarezg@ipn.mx (O. J. Gambino)

CEUR Workshop Proceedings (CEUR-WS.org)

In the 2023 edition, two task were proposed. Violent event identification (binary classification) and violent event category recognition (multilabel classification). Text and images were provided for both tasks to consider single or multimodal information.

In this paper, we describe our participation in the DA-VINCIS task. The rest of this paper is organized as follows. Section 2 describes the task and the corpus. Section 3 describes the method we used. Section 4 explains the performed experiments and the obtained results. Section 5 shows our conclusions and future work.

2. Task and corpus description

The DA-VINCIS task aims to determine whether or not a tweet describes a violent incident by analyzing its textual and visual information. Two tracks were featured:

1. Identification of violent events: This involves determining whether a given tweet is associated with a violent incident or not. It is a binary classification task.
2. Recognition of categories of violent events: Recognizing the crime category to which a given tweet belongs. This is a multilabel classification task.

To accomplish the task, a significant corpus was collected from Twitter. A total of 4,731 tweets were collected and annotated, which were distributed among the different phases of the competition, for both training and testing. This included 3,578 tweets for the various training phases and 1,153 for testing. Each tweet includes textual information and an associated image that complements the tweet's context. This detail allows contestants to build models that interpret not only the textual content but also the visual content of the tweet, adding a layer of information. In the experiments to be described in Section 4, only textual information will be used, so the use of images is proposed for future work.

Tweets were labeled with the following categories:

- Accident: An eventual event or action resulting in unintentional harm to people or things.
- Murder: Deprivation of life.
- Robbery: Voluntary appropriation or destruction of another person's property without the right or consent of the person who can legally dispose of them.
- None of the above: Selected when no crime is reported in the tweet. It is worth noting that tweets under this category were also collected using keywords associated with violent events.

All categories will be utilized for the second track, while for the first track, only two categories will be considered, namely "none-of-the-above" versus the rest, i.e., violent events.

The challenge was conducted on the CodaLab platform. The shared task was divided into two stages:

- Development phase. Participants were provided with labeled training data, including 2,996 tweets and labeled validation data comprising 582 tweets. During this phase, which lasted approximately two months, participants could submit predictions for the validation set and receive immediate feedback on the CodaLab site.

- Final phase. Participants were given the same labeled training data set as in the previous development phase and unlabeled test data consisting of 1,153 tweets. They were allowed to upload up to five submissions per day, ten in total during the entire competition. Performance on the test set was used to rank participants. No immediate feedback was provided on the CodaLab site during this phase.

For track 1, the evaluation measures considered were recall, precision, and F1 score, specifically for the violent-incident class. For track 2, macro average recall, precision, and F1 score were considered. In both cases, the primary evaluation measure was the F1 score.

3. Method

Before applying machine learning techniques to text data, it is essential to perform adequate preprocessing. This process aims to clean and transform the text into a structured form, facilitating extracting relevant features and improving the results' quality.

3.1. Tokenization and Lemmatization

The first stage of data preprocessing in text analysis is tokenization. This technique involves dividing the text into smaller units known as tokens. Tokens can be individual words or even short phrases, depending on the desired level of granularity.

Once the tokens have been obtained, lemmatization is applied. This phase aims to reduce words to their base form, also known as the lemma. For example, the words "running," "runs," and "ran" would be lemmatized to their base form, "run." This helps reduce vocabulary variability and ensures that different forms of the same word are treated as a single entity.

3.2. Removal of Stop Words

Another important step in data preprocessing is the removal of stop words. Stop words are common words that do not provide significant information for analysis and can be safely removed without affecting the overall meaning of the text. The removed stop words include articles, prepositions, conjunctions, and common pronouns in Spanish.

3.3. Text representation

Once the data preprocessing is completed, selecting relevant features, appropriately representing the text, and applying machine learning algorithms to obtain robust predictive models are essential.

Regarding text representation, three different vector representation techniques were used: frequency-based representation, which converts the text into a numerical matrix; binary vector representation; and TF-IDF (Term Frequency-Inverse Document Frequency). In binary vector representation, 1 is assigned if a word is present in the text and 0 otherwise. On the other hand, TF-IDF assigns a weight to each word based on its frequency in the text and the overall corpus.

3.4. Data Partitioning and Model Optimization

Next, the data were divided into training and development sets, with 90% used for training and the remaining 10% for testing. Subsequently, the machine learning models to be evaluated in this study were selected. The considered models were Logistic Regression, Support Vector Machines, Naive Bayes, Multilayer Perceptron, and XGBoost. Then, an exhaustive hyperparameter search was performed using the grid search technique combined with cross-validation. This strategy allowed exploring different combinations of hyperparameters and identifying those that optimized the performance of each model.

3.5. Model Evaluation and Ensemble Approach

Finally, once the models were fine-tuned, their performance was evaluated using evaluation metrics such as precision, recall, and F1 score, with the latter being the primary metric of interest in this case. Additionally, an ensemble approach was implemented, combining multiple models using the soft voting technique.

In the ensemble approach, the soft voting technique was adopted. This method involves calculating the probability of belonging to each class for each model and conducting a weighted vote to make the final decision. By combining the predictions of multiple models, the ensemble approach aims to leverage the strengths and diversity of each model, leading to improved overall performance and robustness. For track 1, a voting technique was utilized, where the final prediction was based on the collective decision of multiple models. However, for track 2, a multioutput classifier was employed due to the multilabel classification expected.

4. Experiments and results

As mentioned in the previous section, experiments were conducted with three different text vector representations and various machine learning algorithms. A comprehensive search for the best parameters and performance was performed for each vector representation using grid search and 5-fold cross-validation. Additionally, a soft voting ensemble approach was applied. In the case of track 2, a multi-output classifier was also utilized to accommodate the specific characteristics of the expected results. All experiments were done using the machine learning library [7]. The results of these experiments for each track are explained below.

4.1. Violent event identification

This track involves binary detection of violent incidents in tweets, the preprocessing steps explained in the sub-section 3.1 were applied. After several experiments, it was found that the best text representation was the frequency representation. The following are the parameters for each model:

- Logistic Regression: multiclass=’ovr’, C=1, maxiter=100, penalty=’l2’, solver=’newton-cg’
- Support Vector Machine:C=10, degree=2, gamma=’scale’, kernel=’poly’, probability=True
- Naive Bayes: alpha=1

- Multilayer Perceptron: hiddenlayersizes=(10, 10), maxiter=1000, randomstate=42
- XGBClassifier:booster='gbtree', objective='binary:logistic', randomstate=0, eta=0.3, samplingmethod='uniform'

As can be seen in Table 1, Support Vector Machines was the best classifier. However, the ensemble of these methods outperformed the independent classifier results.

Algorithm	**Metrics**		
	precision	recall	fscore
LR	0.90	0.91	0.90
SVM	0.91	0.91	0.91
NB	0.90	0.91	0.90
MLP	0.89	0.90	0.89
XGB	0.91	0.90	0.90
Ensemble	0.93	0.93	0.93

Table 1
Results of all models in the development set

4.2. Violent event category recognition

For the second track, which is the multilabel detection of violent events in a tweet, the same preprocessing as in track 1 was applied. Subsequently, the multioutput classifier was implemented, which is used when dealing with a classification problem with multiple labels or output tasks. Unlike traditional classifiers that focus on a single label or task, the multioutput classifier has the ability to predict multiple labels simultaneously. Each output represents a different classification label or task, which is particularly useful in situations where there is interdependence between the tasks or when predicting multiple related features or labels. Another issue that was considered was the class unbalance. In particular, the classes in this task were highly unbalanced. Therefore, the parameter class_weight='balanced' was assigned when creating the machine learning models. This parameter allows handling the unbalance in the class's weights inversely proportional to their respective frequencies. Next, the grid search was conducted, and it was found that the best representation of the text was the binary representation. The following parameters were obtained for each model:

- Logistic Regression: warmstart=False, multiclass='auto', classweight='balanced', C= 5000, maxiter= 50000, penalty= 'l2', solver= 'lbfgs'
- Support Vector Machine: C= 0.5, shrinking=False, classweight= 'balanced', gamma= 'scale', kernel= 'rbf', probability=True
- Naive Bayes: alpha=1
- Multilayer Perceptron: hiddenlayersizes=(1000, 500), maxiter=10000, randomstate=0, learningrate="constant", solver= 'lbfgs'
- XGBClassifier: booster='gbtree', objective='reg:logistic', randomstate=0, eta=0.3, samplingmethod='uniform'

In particular, for this subtask, the logistic regression model achieved the best results on the development set (10% of the training corpus). The results of all models are shown in Table 2.

Algorithm	**Metrics**		
	precision	recall	fscore
LR	0.80	0.82	0.81
SVM	0.84	0.72	0.75
NB	0.49	0.48	0.48
MLP	0.86	0.68	0.74
XGB	0.83	0.73	0.77
Ensemble	0.85	0.82	0.83

Table 2
Results of all models in the development set

However, it can be observed that some models obtained results below expectations. Therefore, in the soft voting ensemble, it was decided not to include Naive Bayes and Multi layer Perceptron. The voting algorithms included in the ensemble are Logistic Regression, Support Vector Machine, and XGB. The results are shown in the last row of Table 2.

4.3. Results of models in the final phase file

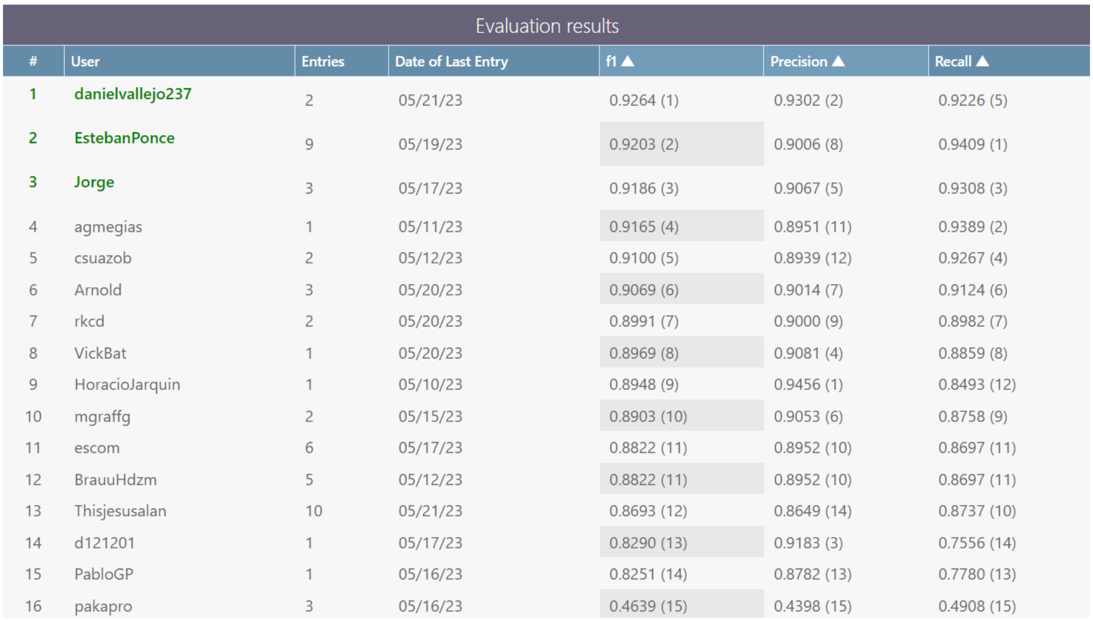
The trained models shown in the previous subsections were used on the dataset provided for the final phase. This dataset consisted of 1,153 tweets and was used for both tracks. The same preprocessing steps explained in the preprocessing subsection were applied to this dataset. Finally, predictions for each instance were made using the corresponding models for each track.

In Figure 1, the results of the participants in the contest for track 1 are shown. It can be observed that we obtained the 11th place in the contest (ESCOM team), keeping in mind that the leading metric is the F1-score.

Similarly, in Figure 2, the results of the participants in the contest for subtask 2 are shown. It can be observed that we obtained the 9th place in the contest (ESCOM team), keeping in mind that the leading metric is the F1-score.

5. Conclusions and future work

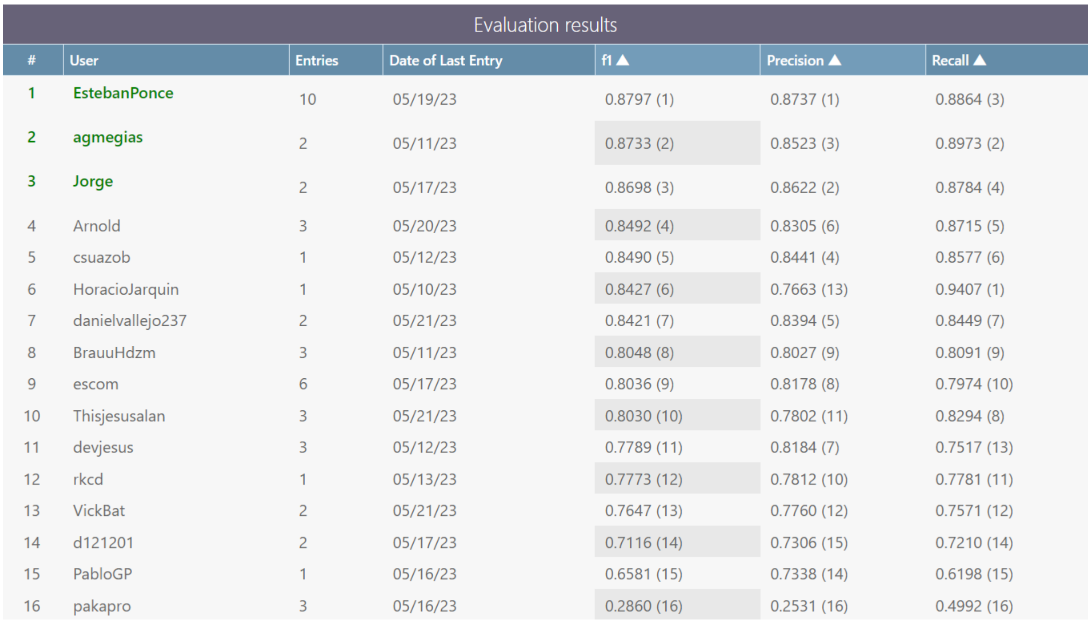
Violence has negative consequences in people. Detecting violent acts helps the authorities to take action and reduce the damage caused by such acts. In this paper, we report our participation in DAVINCIS 2023 task. Two tracks were proposed: violent event identification and violent event category recognition. Different text representations and machine learning methods were tried. Classifiers were fine-tunned and a voting ensemble schema was used to improve the performance. Our proposal obtained the 11th place in the first track and the 9th place for the second task. As future work, we propose to use the images provided in tweets in order to include additional futures. Pretraining language models could also be explored.



Evaluation results

#	User	Entries	Date of Last Entry	f1 ▲	Precision ▲	Recall ▲
1	danielvallejo237	2	05/21/23	0.9264 (1)	0.9302 (2)	0.9226 (5)
2	EstebanPonce	9	05/19/23	0.9203 (2)	0.9006 (8)	0.9409 (1)
3	Jorge	3	05/17/23	0.9186 (3)	0.9067 (5)	0.9308 (3)
4	agmegias	1	05/11/23	0.9165 (4)	0.8951 (11)	0.9389 (2)
5	csuazob	2	05/12/23	0.9100 (5)	0.8939 (12)	0.9267 (4)
6	Arnold	3	05/20/23	0.9069 (6)	0.9014 (7)	0.9124 (6)
7	rkcd	2	05/20/23	0.8991 (7)	0.9000 (9)	0.8982 (7)
8	VickBat	1	05/20/23	0.8969 (8)	0.9081 (4)	0.8859 (8)
9	HoracioJarquin	1	05/10/23	0.8948 (9)	0.9456 (1)	0.8493 (12)
10	mgraffg	2	05/15/23	0.8903 (10)	0.9053 (6)	0.8758 (9)
11	escom	6	05/17/23	0.8822 (11)	0.8952 (10)	0.8697 (11)
12	BrauuHdzm	5	05/12/23	0.8822 (11)	0.8952 (10)	0.8697 (11)
13	Thisjesusalan	10	05/21/23	0.8693 (12)	0.8649 (14)	0.8737 (10)
14	d121201	1	05/17/23	0.8290 (13)	0.9183 (3)	0.7556 (14)
15	PabloGP	1	05/16/23	0.8251 (14)	0.8782 (13)	0.7780 (13)
16	pakapro	3	05/16/23	0.4639 (15)	0.4398 (15)	0.4908 (15)

Figure 1: DAVINCIS 2023 official results for subtask1



Evaluation results

#	User	Entries	Date of Last Entry	f1 ▲	Precision ▲	Recall ▲
1	EstebanPonce	10	05/19/23	0.8797 (1)	0.8737 (1)	0.8864 (3)
2	agmegias	2	05/11/23	0.8733 (2)	0.8523 (3)	0.8973 (2)
3	Jorge	2	05/17/23	0.8698 (3)	0.8622 (2)	0.8784 (4)
4	Arnold	3	05/20/23	0.8492 (4)	0.8305 (6)	0.8715 (5)
5	csuazob	1	05/12/23	0.8490 (5)	0.8441 (4)	0.8577 (6)
6	HoracioJarquin	1	05/10/23	0.8427 (6)	0.7663 (13)	0.9407 (1)
7	danielvallejo237	2	05/21/23	0.8421 (7)	0.8394 (5)	0.8449 (7)
8	BrauuHdzm	3	05/11/23	0.8048 (8)	0.8027 (9)	0.8091 (9)
9	escom	6	05/17/23	0.8036 (9)	0.8178 (8)	0.7974 (10)
10	Thisjesusalan	3	05/21/23	0.8030 (10)	0.7802 (11)	0.8294 (8)
11	devjesus	3	05/12/23	0.7789 (11)	0.8184 (7)	0.7517 (13)
12	rkcd	1	05/13/23	0.7773 (12)	0.7812 (10)	0.7781 (11)
13	VickBat	2	05/21/23	0.7647 (13)	0.7760 (12)	0.7571 (12)
14	d121201	2	05/17/23	0.7116 (14)	0.7306 (15)	0.7210 (14)
15	PabloGP	1	05/16/23	0.6581 (15)	0.7338 (14)	0.6198 (15)
16	pakapro	3	05/16/23	0.2860 (16)	0.2531 (16)	0.4992 (16)

Figure 2: DAVINCIS 2023 official results for subtask2

6. Acknowledgments

This research was funded by CONAHCYT-SNI and Instituto Politécnico Nacional (IPN), through grants SIP-20232011 and EDI.

References

[1] H. Kwak, C. Lee, H. Park, S. Moon, What is Twitter, a social network or a news media?, in: Proceedings of the 19th international conference on World wide web, ACM, 2010, pp. 591–600.

[2] M. Wolfowicz, S. Perry, B. Hasisi, D. Weisburd, Faces of radicalism: Differentiating between violent and non-violent radicals by their social media profiles, Computers in Human Behavior 116 (2021) 106646.

[3] R. Ali, U. Farooq, U. Arshad, W. Shahzad, M. O. Beg, Hate speech detection on twitter using transfer learning, Computer Speech & Language 74 (2022) 101365.

[4] 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) 207–215.

[5] S. M. Jiménez-Zafra, F. Rangel, M. Montes-y Gómez, Overview of IberLEF 2023: Natural Language Processing Challenges for Spanish and other Iberian Languages, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023), co-located with the 39th Conference of the Spanish Society for Natural Language Processing (SEPLN 2023), CEUR-WS.org, 2023.

[6] H. Jarquín-Vásquez, D. I. Hernández Farías, L. J. Arellano, H. J. Escalante, L. Villaseñor-Pineda, M. Montes y Gómez, F. Sanchez-Vega, Overview of da-vincis at iberlef 2023: Detection of aggressive and violent incidents from social media in spanish, Procesamiento del Lenguaje Natural 71 (2023).

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.