Expert Systems With Applications 216 (2023) 119446

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

SocialHaterBERT: A dichotomous approach for automatically detecting hate speech on Twitter through textual analysis and user profiles

Gloria del Valle-Cano [a], Lara Quijano-Sánchez [a,b,*], Federico Liberatore [c,b], Jesús Gómez [d]

[a] Escuela Politécnica Superior, Universidad Autonóma de Madrid, Madrid, Spain
[b] UC3M-Santander Big Data Institute, Universidad Carlos III de Madrid, Madrid, Spain
[c] School of Computer Science & Informatics, Cardiff University, UK
[d] Dirección General de Coordinación y Estudios, Secretaría de Estado de Seguridad, Ministerio del Interior, Madrid, Spain

ARTICLE INFO

Keywords:
Hate speech
Twitter
Deep learning
Social network analysis
BERT
Topic modeling

ABSTRACT

Social media platforms have evolved into an online representation of our social interactions. We may use the resources they provide to analyze phenomena that occur within them, such as the development and viralization of offensive and hostile content. In today's polarized world, the escalating nature of this behavior is cause for concern in modern society. This research includes an in-depth examination of previous efforts and strategies for detecting and preventing hateful content on the social network Twitter, as well as a novel classification approach based on users' profiles, related social environment and generated tweets. This paper's contribution is threefold: (i) an improvement in the performance of the *HaterNet* algorithm, an expert system developed in collaboration with the Spanish National Office Against Hate Crimes of the Spanish State Secretariat for Security (Ministry of the Interior) that is capable of identifying and monitoring the evolution of hate speech on Twitter using an LTSM + MLP neural network architecture. To that end, a model based on BERT, *HaterBERT*, has been created and tested using *HaterNet*'s public dataset, providing results that show a significant improvement; (ii) A methodology to create a user database in the form of a relational network to infer textual and centrality features. This contribution, *SocialGraph*, has been independently tested with various traditional Machine Learning and Deep Learning algorithms, demonstrating its usefulness in spotting haters; (iii) a final model, *SocialHaterBERT*, that integrates the previous two approaches by analyzing features other than those inherent in the text. Experiment results reveal that this last contribution greatly improves outcomes, establishing a new field of study that transcends textual boundaries, paving the way for future research in coupled models from a diachronic and dynamic perspective.

1. Introduction

Messages that aim to promote and feed a dogma against certain individuals or groups are one of the most serious problems of the digital era (Paz, Montero-Díaz, & Moreno-Delgado, 2020). This phenomenon, that thrives on other people's hatred and spreads like a disease among users (Müller & Schwarz, 2021), can be observed in a variety of social media platforms, particularly Twitter, where users who are free to express their opinions without fear of censorship or filtering find it simple to send offensive messages especially, when creating multiple anonymous accounts (Paulson, 2021). In recent years, the number of hate crimes in Spain[1] and in general throughout the world continues on an upward trend (Gover, Harper, & Langton, 2020; Morgante, 2021; Müller & Schwarz, 2020). According to the *2019's report on the evolution of hate crimes in Spain* (Spanish Ministry of Interior, 2019), threats and insults are the most common criminal acts, with the Internet (54.9%)

Abbreviations: BERT, Bidirectional Encoder Representations from Transformers; LTSM, Long Short-Term Memory; MLP, Multi Layer Perceptron; BOW, Bag of Words; TFIDF, Term Frequency Inverse Document Frequency; GLOVE, Global Vectors; POS-TAG, Part-Of-Speech Tagging; LR, Linear Regression; SVM, Support Vector Machine; NB, Naive Bayes; KNN, K-Nearest Neighbors; CNN, Convolutional Neural Network; NER, Named Entity Recognition; GRU, Gated Recurrent Unit; MUSE, Multilayer Self-Evolving; NLP, Natural Language Processing; LDA, Latent Dirichlet Allocation; LSI, Latent Semantic Indexing; HDP, Hierarchical Dirichlet Process; LOOCV, Leave-One-Out Cross-Validation

* Correspondence to: Escuela Politécnica Superior, Universidad Autónoma de Madrid, C/ Francisco Tomás y Valiente, 11, Campus de Cantoblanco, 28049 Madrid, Spain.

E-mail addresses: gloria.valle@estudiante.uam.es (G.d. Valle-Cano), lara.quijano@uam.es (L. Quijano-Sánchez), liberatoreF@cardiff.ac.uk (F. Liberatore), jge@interior.es (J. Gómez).

[1] https://www.interior.gob.es/opencms/pdf/servicios-al-ciudadano/delitos-de-odio/estadisticas/informe-evolucion-2019.pdf

https://doi.org/10.1016/j.eswa.2022.119446
Received 21 October 2021; Received in revised form 22 June 2022; Accepted 15 December 2022
Available online 20 December 2022

Table 1
Most relevant datasets found in English.

Source	No of classes	Labels	Size	Year	Cohen's kappa coefficient	Labeling process
Waseem and Hovy (2016)	4	racist, sexist, both, normal	16,907	2016	0.85	experts and crowdsourcing
Davidson, Warmsley, Macy, and Weber (2017)	3	hateful, offensive (but not hateful), neither	24,802	2017	0.92	crowdsourcing
Founta et al. (2018)	7	offensive, abusive, hateful speech, aggressive, cyberbullying, spam, normal	80,000	2018	–	crowdsourcing
Basile, Bosco, Fersini, Debora, Patti, Pardo, et al. (2019) HatEval (en)	2	hate, non-hate	13,000	2019	0.83	crowdsourcing
Kumar and Pranesh (2021) TweetBLM	2	hate, non-hate	9165	2021	–	students

Table 2
HaterNet dataset distribution, where 73.89% of the corpus is tagged as non-hate (0) and 26.11% as hate (1). *Downloaded* reflects the number of tweets that where randomly downloaded in the indicated date ranges; *Selected* includes those that passed a filter comprised of 6 hate speech vocabulary dictionaries and 1 of insults; *Labeled* indicates those randomly chosen for expert labeling.

Class	Downloaded	Selected	Labeled
Non-hate (0)	–	–	4433
Hate (1)	–	–	1567
Total	2 million	8710	**6000**

Table 3
HatEval (es) dataset distribution, where 58.5% of the corpus is tagged as non-hate (0) and 41.5% as hate (1).

Class	Train	Test	Total
Non-hate (0)	2921	940	3861
Hate (1)	2079	660	2739
Total	5000	1600	**6600**

and social media (17.2%) being the most popular platforms for carrying them out. As a result, the main goal of this work is to aid in the fight against prejudice-based discriminatory behaviors by providing information to Spanish security agencies and police forces about hate speech messages and trends on Twitter, thus, helping predict possible hate crimes or triggers and design preventive measures. This work has been done in collaboration with the Spanish National Office Against Hate Crimes of the Spanish State Secretariat for Security (*Oficina Nacional de Lucha Contra los Delitos de Odio*, ONDOD).

Following a thorough review of the state of the art in hate speech detection, it was observed that almost all studies focus on analyzing the text of Twitter messages, with the majority of algorithms trained for English (Poletto, Basile, Sanguinetti, Bosco, & Patti, 2021), three for Spanish (Plaza-del Arco, Molina-González, Ureña-López, & Martín-Valdivia, 2021; Basile et al., 2019; Pereira-Kohatsu, Quijano-Sánchez, Liberatore, & Camacho-Collados, 2019), and a few exceptions for multi-language versions or other languages (Aluru, Mathew, Saha, & Mukherjee, 2020; Battistelli, Bruneau, & Dragos, 2020; Florio, Basile, Polignano, Basile, & Patti, 2020; Sreelakshmi, Premjith, & Soman, 2020). In this review, no work that combines user profiles and relationships with textual analysis was found. Opening thus, an excellent scientific opportunity to build a multimodal model. To address this chance, three different approaches to dealing with hate speech on Twitter have been developed:

- In the first, the problem of finding a cutting-edge algorithm that serves as a baseline for detecting hate is addressed using only text as input. In Pereira-Kohatsu et al. (2019) (the alpha version of this project) the *HaterNet*'s algorithm, based on an LTSM + MLP neural network, was shown to improve the techniques designed so far. As a result, in this paper, we focus on improving *HaterNet* and the proposals that followed. All of these proposals revolve around the use of the BERT algorithm. For this, *HaterBERT* is built as a first contribution, a model based on BERT (Devlin, Chang, Lee, & Toutanova, 2019) that only analyzes the text of tweets to classify them as hateful or not.
- In the second, the impact on detecting hate of textual and numerical characteristics of users' profiles, users' past activity within the social network, and the users' environment is explored.[2] For this, we create *SocialGraph* (a collection of descriptive characteristics of Twitter's users) and study their significance in determining whether a profile is hater or not.
- In the third, the problem of developing a classifier that improves on existing hate detection techniques is addressed. As a result, *SocialHaterBERT* is created, an algorithm that unifies the two previous contributions by combining as input parameters the text of the message with the characteristics that define the user within the social network.

In short, the main contribution of this work is the development of a methodology that extracts certain characteristics from Twitter's user profiles with the goal of modeling attributes along with the text of the tweet itself, outperforming the best base algorithm that only uses textual information by 4% and proving to be critical in detecting hate within the social network.

Hence, our research aims at providing an answer to the following hypotheses.

H1 Transformer-based models such as BERT present a good approach to the problem of detecting hate speech, since it requires a contextual understanding of tweets.

H1.1 Specifically, BETO is the best model that classifies hate speech in the Spanish language.

[2] The processing of personal data for the purposes of this research complies with the requirements for the lawful processing of personal data under the European Union General Data Protection Regulation (GDPR). See Art. 6.1 and 9.2 GDPR.

Table 4
Summary of the studies found in the English literature, along with the datasets, scoring, models, and strategies used in each. In bold, the best performing approaches.

Dataset	Paper	Date	Preprocessing strategies	Models	Best F1-score
951,736 Yahoo comments	Djuric et al. (2015)	May. 2015	BOW, TF, TF-IDF, **paragraph2vec embeddings**	**LR**	–
Tweet recollection (own)	Zia et al. (2016)	Nov. 2016	unigrams, TF-IDF, retweets, favs, autenticidad de la página	**SVM**, NB, kNN	0.971
Waseem (2016)	Waseem (2016)	Jan. 2016	char n-grams, skip n-grams, word n-grams, tweet length, author's gender , POS-TAG, clusters	**LR**	0.912
Waseem and Hovy (2016)	Waseem and Hovy (2016)	Jun. 2016	author's gender, tweet length, description length, location, char n-grams, word n-grams	**LR**	0.7393
6502 Facebook comments (Del Vigna12, Cimino23, Dell'Orletta, Petrocchi, & Tesconi, 2017)	Del Vigna12 et al. (2017)	Jan. 2017	POS-TAG, sentiment analysis, word2vec, CBOW, n-grams, word polarity	SVM, **LSTM**	0.731
Waseem and Hovy (2016)	Badjatiya, Gupta, Gupta, and Varma (2017)	Apr. 2017	char n-grams, **random embeddings**, GloVe	LR, RF SVM, **GBDT**, DNN, CNN, **LSTM**	0.930
Davidson et al. (2017)	Davidson et al. (2017)	May. 2017	n-grams, TF-IDF, POS-TAG, readability sentiment, URLs	**LR**, NB, DT, RF, SVM	0.900
Waseem (2016), Waseem and Hovy (2016)	Park and Fung (2017)	Jun. 2017	word embeddings, random embeddings, char n-grams	CharCNN, WordCNN, **HybridCNN**	0.8270
6655 from (Waseem, 2016)	Gambäck and Sikdar (2017)	Aug. 2017	word embeddings, random embeddings, char n-grams	**CNN**	0.7829
WZ, WZ-S.amt, WZ-S.exp, WS.gb, WZ.pj, DT, RM, (Zhang, Robinson, & Tepper, 2018)	Zhang and Luo (2019)	Oct. 2018	n-grams, POS-TAG, TF-IDF, menciones, hashtags, misspellings, emojis, **word embeddings**	**CNN+sCNN**, CNN+GRU	0.820–0.940
5143 Twitter & Facebook comments (Salminen et al., 2018)	Salminen et al. (2018)	Mar. 2019	n-grams, TFIDF, word2vec, doc2vec	LR, DT, RF, Adaboost, **SVM**	0.96
Waseem and Hovy (2016), Davidson et al. (2017)	Mozafari, Farahbakhsh, and Crespi (2019)	Oct. 2019	–	BERT+LSTM, BERT, BERT+NLL, **BERT+CNN**	0.880, 0.920
Davidson et al. (2017)	Kovács, Alonso, and Saini (2021)	Abr. 2021	–	CNN-LSTM, **RoBERTa + FastText**	0.798

H2 The context and user characteristics inside the social network are useful for classifying hate on Twitter.

H3 The development of multimodal classification models, particularly for the problem of detecting hate, is an improvement over models based solely on text.

The rest of this paper is structured as follows: In Section 2 we conduct a thorough study of state of the art techniques in hate speech recognition. Next, Section 3 defines this paper's three approaches. Following, Section 4 describes the experimental design and obtained results. Finally, Section 5 concludes the paper and proposes future research lines.

2. State of the art

This section presents the current state of the art on hate speech detection. For this, we conducted a thorough bibliographic review in accordance with a Bibliographic Review Protocol, which can be found in Appendix B, in it, we studied: (i) definitions and concepts, (ii) datasets used and available, and (iii) previous related works.

2.1. Definitions

Hate speech has a major flaw that makes it difficult to categorize: subjectivity. Because what constitutes hate speech, in addition to domain and context, is often open to interpretation, the magnitude and

Table 5
Main characteristics of the approaches created for the Spanish language, along with the datasets used, validation methods, models and obtained results.

Dataset	Paper	Date	Best Model	Validation	F1-score
HaterNet Pereira-Kohatsu et al. (2019)	Pereira-Kohatsu et al. (2019)	Oct. 2019	LSTM+MLP	LOOCV	0.611
HaterNet Pereira-Kohatsu et al. (2019) & *HatEval (es)* Basile et al. (2019)	Aluru et al. (2020)	Apr. 2020	mBERT	70-20-10	0.733–0.734
HaterNet Pereira-Kohatsu et al. (2019) & *HatEval (es)* Basile et al. (2019)	Plaza-del Arco et al. (2021)	Mar. 2021	BETO	10 k-Fold	0.772–0.776

scope of the problem varies by project. This necessitates the definition of hate speech a priori, which results in a wide range of labeling in the datasets and their quantity on a practical level.

In Fortuna and Nunes (2018) the authors present multiple hate speech definitions and compare views from different sources. In this paper, the concept of hate speech is understood within the framework of a hate crime as defined by the Organization for Security and Cooperation in Europe (OSCE)[3]:

> *A hate crime is any criminal offense, including those committed against people or property, where the protected legal asset is chosen for its, real or perceived, connection, sympathy, affiliation, support or belonging to a group. A group is based on a common characteristic of its members, such as their real or perceived race, national or ethnic origin, language, color, religion, age, disability, sexual orientation, or another similar factor.*

In this way, the scope of this work is defined as a binary classification problem: hate or non-hate.

2.2. Datasets

In terms of the datasets used to train the various algorithms developed to date it can be seen that, based on the approach and hate speech definitions addressed in each work, there is a great variety of choices depending on the classification label (Fortuna & Nunes, 2018; Poletto et al., 2021). Three distinct groups can be identified. The first is the binary classification, which is made up of two distinct values: hate and non-hate (Basile et al., 2019). The second is based on three or more mutually or non-exclusive values, such as dividing strong hatred into offensive, aggressive, sexist, or racist categories (Mathur, Shah, Sawhney, & Mahata, 2018; Waseem & Hovy, 2016). The latter is based on combined annotation, which consists of a first division based on abusive or non-abusive language, followed by more specific classes such as hate speech, derogatory, or profane language (Fersini, Rosso, & Anzovino, 2018). Also noteworthy is the datasets' tagging source, main options include: annotations by experts (whether judges or expert developers on the subject) (Pereira-Kohatsu et al., 2019), tagging by non-expert volunteers (Poletto et al., 2021), crowdsourcing platforms (Basile et al., 2019; Davidson et al., 2017), or automatic classifiers (Ribeiro, Calais, Santos, Almeida, & Meira Jr, 2018).

The various datasets studied can be viewed from various perspectives; however, given the study's focus, they are grouped by language. It is important to note that, while some datasets were collected from other social networks, such as Facebook (Del Vigna12 et al., 2017; Salminen et al., 2018), Twitter datasets are used to a greater extent. There are at least 32 public datasets that come from Twitter (Poletto et al., 2021), the social network reviewed in this study and discussed below.

[3] More in BOE-A-2019-777 (in Spanish) or in HateCrimeData2019(OSCE) (in English).

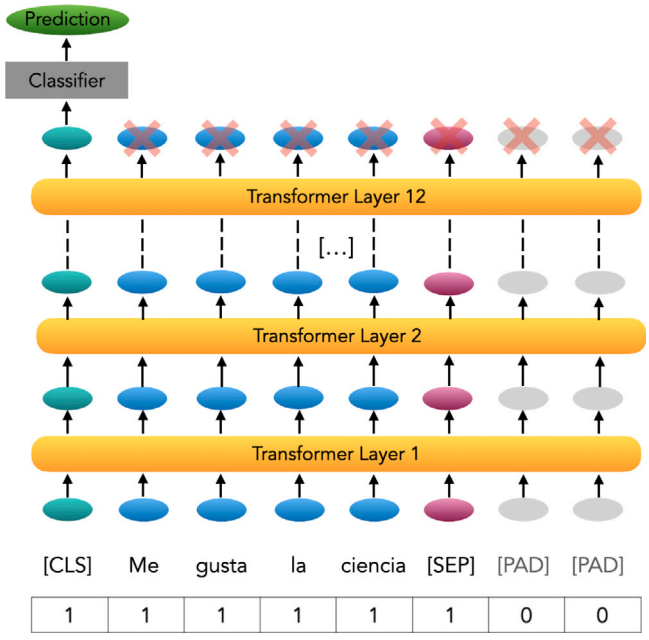

Fig. 1. Illustration of BertForSequenceClassification's preprocessing of a phrase for Fine-Tuning. The sequence of digits corresponds to the attention mask, and the sentence has been padded to be the same length as all the other sentences in the dataset.

Table 6
Summary of the hyperparameters tested for *HaterBERT*.

Hyperparameters	Alternatives
Epochs	[2, 3, 4, 5]
Learning Rate	[2e−5, 3e−5, 5e-5]
Random Seed	[**2018**, 2019, 2020, 2021, 2022, 2023]
Batch Size	[16, 32]
Epsilon	[1e−6, 1e-8]
Max. Length	256

2.2.1. Datasets in English

As one might expect, the vast majority of existing datasets are in English. The most popular or cited, as well as those with the most tweets, are collected in this study (see Table 1). Where the dataset by Basile et al. (2019) is the one that, due to its binary labeling, best suits our approach. As a result, we use it in our experiments (see Section 4.1).

2.2.2. Datasets in Spanish

Given our collaboration with the ONDOD, the language that is primarily addressed in this work is Spanish. There are only two datasets

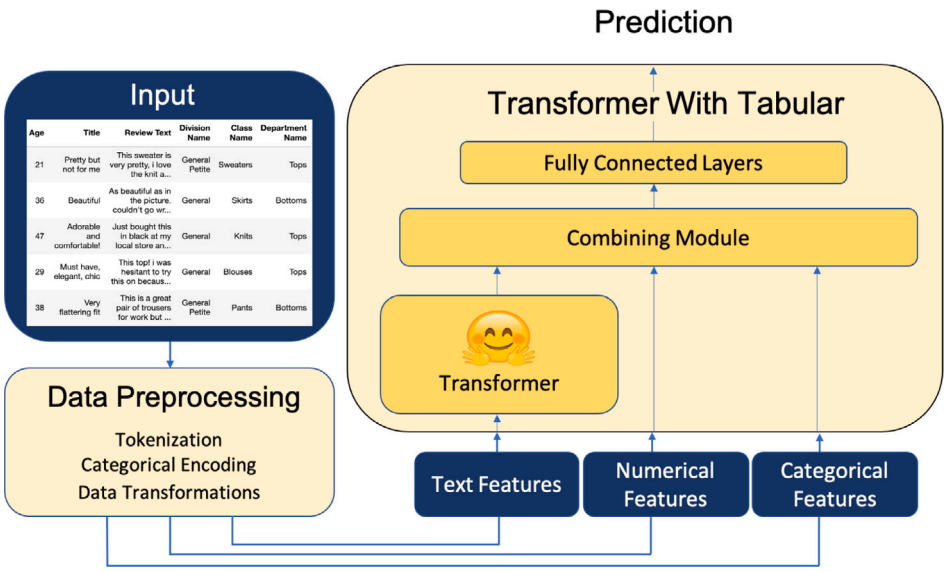

Fig. 2. Diagram the Multimodal Transformers structure. This toolkit allows the elaboration of multimodal models that combine transformers with extra numerical and categorical variables. Source https://github.com/georgian-io/Multimodal-Toolkit/.

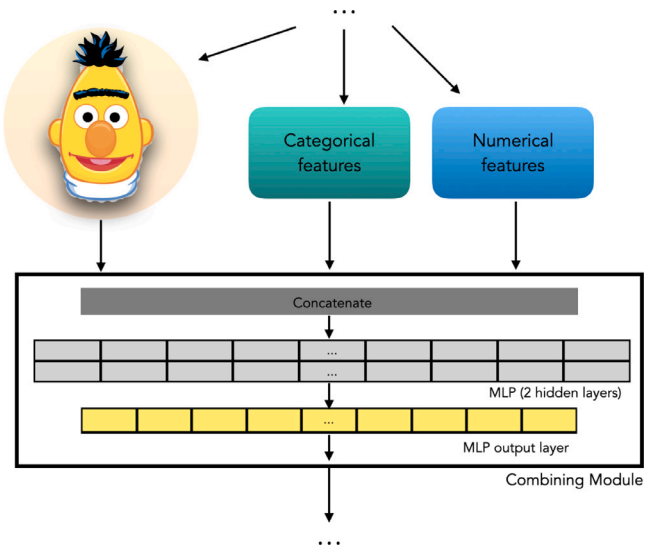

Fig. 3. Multimodal Transformers' combining module and input details.

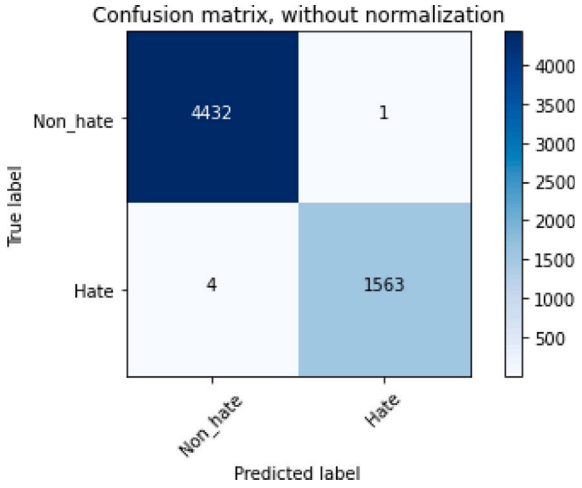

Fig. 4. *HaterBERT* best confusion matrix performed with LOOCV.

in Spanish that present a binary classification of hate or non-hate, both published in 2019 and with a similar number of tweets:

- *HaterNet* (Pereira-Kohatsu et al., 2019)[4]: Contains 6000 tagged tweets downloaded between February and December 2017. It was hand-tagged using majority voting by four experts from various sociological backgrounds. This dataset has a score of 0.588 according to Cohen's kappa coefficient, indicating that it falls within a fairly reliable range of moderate agreement. Table 2 shows the dataset's details. The benefit of this dataset is that it includes not only a tweet's text and associated tag, but also its identifier, which allows it to be searched using the Twitter API today. This factor is particularly important for the ability to download additional data for the current study, which is critical when choosing a reference dataset.
- *HatEval in Spanish* (Basile et al., 2019)[5]: It was proposed by the organizers of SemEval2019 task5 HatEval, which consisted of detecting hate speech against two targets: women and immigrants. The data was collected using the Figure Eight crowdsourcing platform from July to November 2017 and from July to September 2018. With a Cohen's kappa coefficient of 0.89, this dataset showed sufficient reliability, indicating a value of almost perfect agreement. Following that, the labeling was revised by two Spanish-speaking experts using a majority vote, though the final reliability score was not recorded. The text of the tweet and the label associated with it are included in this dataset (see Table 3).

2.3. Related works

The challenge of countering hate speech dates over 50 years back (Bustos Martínez, De Santiago Ortega, Martínez Miró, & Rengifo Hidalgo, 2019). However, due to the increased use of technology and the escalation of information generated on a daily basis, identifying hate speech represents a new issue, where all related studies date from 2015 (Poletto et al., 2021), and the growth of these has been remarkable since then.

[4] Available at: https://zenodo.org/record/2592149#.YMp6Ky0lNaI, last accessed December 24, 2022

[5] Available at: https://www.aclweb.org/anthology/S19-2007/, last accessed December 24, 2022

Table 7
Comparison between *HaterBERT* and *HaterNet* using *HaterNet*'s dataset (Table 2).

Model	Author	Validation	Precision	Recall	F1	AUC	Accuracy	Errors	Hyperparameters
LSTM+MLP	Pereira-Kohatsu et al. (2019)	LOOCV	0.6250	0.5980	0.6110	0.8280	–	–	–
HaterBERT	This proposal	LOOCV	0.9992	0.9986	**0.9989**	0.9986	0.9992	5	Epochs: 5 Batch Size: 32 Learning Rate: 5e−5 Epsilon: 1e-6

2.3.1. Studies in English

The first studies carried out on detecting hate in social networks use mainly hybrid strategies based on lexical resources (e.g. BOW, TFIDF, GLOVE, POS-TAG) and Machine Learning models (e.g. LR, SVM, NB, KNN) (Davidson et al., 2017; Djuric et al., 2015; Silva, Mondal, Correa, Benevenuto, & Weber, 2016; Waseem & Hovy, 2016; Zia et al., 2016). The first neural network-based method was introduced in 2017 (Badjatiya et al., 2017), and its effectiveness was compared to that of previous methods, though traditional models still produced good results in some cases. The most relevant studies found in the literature are listed in Table 4.

There are numerous studies that provide relevant insights when it comes to combating hate speech, in addition to the creation of classifying models to identify hate online. In this spirit, the studies by Olteanu, Castillo, Boy, and Varshney (2018) and de Oliveira et al. (2020) perform real-time data collection, visualization, and monitoring. In a different way, in Mathew, Kumar, Goyal, Mukherjee, et al. (2018), authors analyze hate speech and user responses that counter these opinions (also known as counterspeech). Although the authors' primary focus is on hate and counterspeech, they also conduct lexical, linguistic, and psycholinguistic analyses on user account data, discovering that hate tweets from verified accounts have significantly higher virality than those from unverified accounts. Also, hate accounts seem to use more words about negative emotions. Meanwhile, users who counter hate use more words related to government or laws. In a follow-up study, Ribeiro et al. (2018) found that hate users differ from normal users in terms of activity patterns, word usage, and network structure. Moreover, ElSherief, Kulkarni, Nguyen, Wang, and Belding (2018) addresses the absence of a target-based classification, analyzing directed and generalized hate speech. They show that directed hate speech is explicitly directed at an individual entity and is more informal while generalized hate speech targets a particular community or group, and is dominated by hate towards determined categories, such as nationality, religion, gender or sexual orientation.

While supervised approaches achieve near-perfect results, this is only within specific datasets. For this reason, there are several studies that focus on the investigation of errors and biases both in the datasets and in the strategies taken to detect hate. The reported challenges in Arango, Pérez, and Poblete (2019) are mostly attributable to data overfitting and sampling issues. Furthermore, Badjatiya, Gupta, and Varma (2019) identify flaws with the interpretation of the phrases, resulting in an inherent bias in them. They propose a method to reduce it by using neural network models such as CNN and text replacement techniques such as NER, POS-TAG, and centroids. MacAvaney et al. (2019) meet challenges such as linguistic nuances, varying definitions of what constitutes hate speech, and data availability constraints for training and testing. They also introduce a multi-view SVM strategy for reducing interpretability issues in neural networks. The requirement for model automation and their relationship with the real world are also emphasized.

There are no studies that contain additional information in the models than their textual properties until 2021. According to Vijayaraghavan, Larochelle, and Roy (2021), social and cultural context enhances performance greatly when compared to models based solely on text, though they only include the geographical origin of the tweets and the relationship between users. Another interesting work is Perifanos and Goutsos (2021), which proposes to integrate visual input from images exchanged in a multimodal learning environment, outlining that this could increase model precision.

Following the completion of this review, it is possible to conclude that, to the best of the authors' knowledge, there is no published work that incorporates attributes based on user's profiles with textual features in a single and multimodal model, which is the core proposition of this work.

2.3.2. Studies in Spanish

The publications made for the Spanish language are listed in the Table 5. In the study conducted by Pereira-Kohatsu et al. (2019), authors implement an intelligent system that monitors and visualizes hate within the social network in addition to creating a model to identify hate. Following a review of various approaches, a model based on LSTM + MLP is used in conjunction with a TFIDF-enhanced preprocessing of the input text. The input data is divided into words, emojis, and tweet embeddings, the latter of which is obtained using a word embeddings technique. Authors also introduce the previously mentioned *HaterNet* dataset (see Table 2).

Regarding Aluru et al. (2020), the effectiveness of four models is tested in different languages: MUSE + CNN-GRU, Translation + BERT, LASER + LR y mBERT, with extensive hyperparameter optimization. The effectiveness of using mBERT in a variety of languages is then confirmed, despite the fact that it is not the most appropriate for each of them. Although the results of a pre-BERT translation are not dissimilar to those of mBERT, BERT is trained in English, and accuracy is highly dependent on the quality of the translation. On the other hand, it is observed that using transformers is much more useful for datasets with sufficient information, whereas the results from LASER + LR may be more promising for smaller corpora.

It is in 2021 when the first official publication with BETO was made by Plaza-del Arco et al. (2021), achieving the best result to date and demonstrating that the use of a BERT in Spanish is better suited to the language.

As a result, this project seeks a solution based on BERT that is comparable to other algorithms that rely solely on textual classification as a base algorithm. In this way, we can make a broad comparison with the current state of the art, accurately reporting the improvement that the later proposed multimodal model implies.

2.3.3. Other languages

Although, as previously stated, English is the most studied language, there are a variety of studies based on other languages that provide useful insights. In French, for example, Battistelli et al. (2020) emphasizes the importance of context in detecting hate while Defersha, Kekeba, and Kaliyaperumal (2021) focus on the relevance of parameter tuning in machine learning classifiers, such as SVM.

Also, in Florio et al. (2020) the platform *Contro l'odio* is created to monitor hate speech against immigrants on Twitter in the Italian sphere, examining the temporal robustness of AlBERTo.[6] In this last

[6] https://github.com/marcopoli/AlBERTo-it last accessed December 24, 2022

Table 8
Comparison between *HaterBERT* and (Aluru et al., 2020) using *HaterNet*'s dataset (Table 2).

Model	Author	Validation strategy	F1 (Training size)						Hyperparameters
			16	32	64	128	256	Total	
mBERT	Aluru et al. (2020)	70–20–10	0.4395	0.4285	0.4048	0.4861	0.5999	0.7329	–, Max Length: 128
HaterBERT	This proposal	70–20–10	0.5025	0.5787	0.65401	0.6906	0.7459	**0.7667**	Epochs: 5, Batch Size: 32, Learning Rate: 5e−5, Epsilon: 1e−6, Max. Length: 256

Table 9
Comparison between *HaterBERT* and (Plaza-del Arco et al., 2021) using HatEval (es) dataset (Table 3).

Model	Author	Validation strategy	Precision	Recall	F1	AUC	Accuracy	Errors	Hyperparameters
BETO	Plaza-del Arco et al. (2021)	10 k-Fold	0.6928	0.8303	0.7553	–	–	359	Epochs: 3, Batch Size: 16, Learning Rate: 2e−5, Max. length: 80
HaterBERT	This proposal	10 k-Fold	0.8666	0.8710	**0.8673**	0.8709	0.8680	66	Epochs: 3, Batch Size: 16, Learning Rate: 2e−5, Max. length: 256

study, it is also stated that the model is very sensitive to the dataset's temporal distance, but that with an adequate time window, the performance increases, since hate speech is very sensitive to certain social events. Moreover, Celli, Lai, Duzha, Bosco, and Patti (2021) propose an interesting Italian corpus focused on politics and they suggested that a presence of hate labels above 40% boosts the performance of classifiers.

For the Portuguese language, da Silva and de Freitas (2022) use BERTTimbau, a BERT-based approach to classify hate speech, performing some preprocessing and oversampling technique on three datasets, obtaining better results than other classification models.

The work by Garland, Ghazi-Zahedi, Young, Hébert-Dufresne, and Galesic (2020a) studies in addition a counterspeech strategy in German. Prior to this study, authors analyzed that bullying is more likely to be viral and effective (Garland, Ghazi-Zahedi, Young, Hébert-Dufresne, & Galesic, 2020b). Moreover, Paasch-Colberg, Strippel, Trebbe, and Emmer (2021) go beyond the common hate or no-hate dichotomy with an in-depth analysis of several comments identifying various types of hate speech and offensive language targeting immigrants and refugees.

With the goal of creating a benchmark Arabic dataset for hate speech and abusive content, Mulki, Haddad, Bechikh Ali, and Alshabani (2019) propose the Levantine Hate Speech and Abusive Twitter Dataset (L-HSAB). It is worth mention that Mayda, Demir, Dalyan, and Diri (2021) generated a hate speech dataset comprising 10224 tweets, facing the lack of study in Turkish. Finally, Sreelakshmi et al. (2020) highlight the presence of English in other languages by studying mixed Hindi and English tweets, as well as Modha et al. (2021) presenting one HASOC subtrack for English, Hindi, and Marathi.

2.3.4. Limitations and research opportunities

As presented in Table 4, recent works in the English language have been using BERT as their language classification model. However, all the contributions make use of datasets which are different or not fully comparable. The same happens in the contributions for the Spanish language. Therefore, it is not possible to draw a final conclusion regarding the best model in the literature. Thus, the first hypothesis (H1) tested in this paper concerns comparing the performance of BERT based models to others in the context of hate speech detection in Twitter.

Concerning the Spanish language, Plaza-del Arco et al. (2021) shows BETO's capabilities. However, their comparison is limited to only one previous model. Therefore, a comprehensive comparison is required to verify the superiority of BETO against other BERT-based methods, such as mBERT (Aluru et al., 2020) for the detection of hate speech in the Spanish language. This is the focus of hypothesis H1.1.

Hence, the goal of H1 and H1.1 is the definition of the best text-based classification model, which could then be used as a basis for more advanced models.

A natural step consists in extending text-based models by introducing other predictive features. To this end, the only relevant works in the literature, as previously explained in Section 2.3, are those by Vijayaraghavan et al. (2021) and Perifanos and Goutsos (2021). However, none of them exploits the information provided by the user characteristics. Therefore, the goal of hypothesis H2 is to verify the impact of features based on context and user characteristics in the task of hate speech detection.

Finally, having assessed the predictive capabilities of context and user characteristics, hypothesis H3 is concerned with the improvement in the performance obtained by jointly considering text, context and user characteristics features.

3. Methodology and design

This section introduces the design of the three approaches created for detecting hate speech on Twitter.

3.1. HaterBERT

We now explain the design of *HaterBERT*, our base model for textual hate or no hate classification.[7] This model is based on BERT. The following are the modifications made to the transformer and the tools used to make them:

Base libraries: Tensorflow,[8] Keras,[9] Pytorch,[10]

Transformers Libraries : HuggingFace,[11] which has NLP tools and pre-trained transformers (BERT (Devlin et al., 2019) and its Spanish version BETO (Cañete, Chaperon, Fuentes, Ho, Kang, & Pérez, 2020))

BERT Fine-Tuning library: DE-LIMIT,[12] (Aluru et al., 2020).

3.1.1. Preprocessing

Using as a base the *pre-trained BERT* of HuggingFace, we use the *BertTokenizer* provided thereby since this transformer has a specific fixed vocabulary and a particular way of transforming words into tokens and masks. Although, the following modifications are made for each text input in the *encode*:

- Tokenize the sentence.
- Add the token *[CLS]* at the beginning of the sentence.
- Add the token *[SEP]* at the end of the sentence.
- Assign the tokens to their corresponding token IDs.
- Revision of the tweet's text to correct leet alphabet (compound writing in which letters are substituted for numbers from 0–9), trying to camouflage insults or bad words.

After that, padding is used to ensure that all sequences are the same length, by filling in 0 at the end of each sequence until it is the same length as the longest ($n = 256$). Next, for each sentence the attention mask is created for the corresponding identifiers. It is decided that:

- If the token ID is 0, it is padding and fires as a 0 in the mask.
- If the token ID is greater than 0, then it is a token and it is set to 1.

The outputs are then converted into tensors before being processed. This makes them suitable for generating Pytorch's *DataLoader*,[13] which aids memory management and training speed. Fig. 1 illustrates this process.

3.1.2. Classifier

We use *BertForSequenceClassification*,[14] a class that contains an input layer adapted for text sequences or sentences, to use BERT for sentiment analysis, and specifically for hate detection. Following that, the BERT,[15] pre-trained model for the English version and BETO,[16] for the Spanish version are chosen. Finally, *AdamW* is used for the fine-tuning phase (Loshchilov & Hutter, 2017).

[7] Although the classification problem considered is binary, the methodology presented can be easily extended to the multiclass version of the problem.

[8] https://www.tensorflow.org/?hl=es-419 last accessed December 24, 2022

[9] https://www.tensorflow.org/guide/keras?hl=es last accessed December 24, 2022

[10] https://pytorch.org last accessed December 24, 2022

[11] https://huggingface.co/transformers/index.html last accessed December 24, 2022

[12] https://github.com/hate-alert/DE-LIMIT last accessed December 24, 2022

[13] https://pytorch.org/docs/stable/data.html last accessed December 24, 2022

[14] https://huggingface.co/transformers/_modules/transformers/models/bert/modeling_bert.html#BertForSequenceClassification last accessed December 24, 2022

[15] https://github.com/google-research/bert last accessed December 24, 2022

[16] https://github.com/dccuchile/beto last accessed December 24, 2022

Table 10
Comparison between *HaterBERT* and (Plaza-del Arco et al., 2021) using *HaterNet*'s dataset (Table 2).

Model	Author	Validation strategy	Precision	Recall	F1	AUC	Accuracy	Errors	Hyperparameters
BETO	Plaza-del Arco et al. (2021)	10 k-Fold	0.7045	0.6282	0.6580	–	–	106	Epochs: 2, Batch Size: 16, Learning Rate: 2e−5, Max. length: 80
HaterBERT	This proposal	10 k-Fold	0.9165	0.9100	0.9132	0.9101	0.9335	99	Epochs: 2, Batch Size: 16, Learning Rate: 2e−5, Epsilon: 1e−6, Max. length: 256
HaterBERT	This proposal	10 k-Fold	0.9766	0.9791	**0.9778**	0.9701	0.9828	73	Epochs: 5, Batch Size: 32, Learning Rate: 5e−5, Epsilon: 1e−6, Max length: 256

3.2. Socialgraph

To get *HaterBERT* to feed on the characteristics of the social network, it is first necessary to get all the relative information. To do this, given a dataset D consisting of tweets that may or may not contain hate, we first collect:

1. Information related to each tweet (i.e text, author, number of retweets, responses, etc.). The collected fields can be seen in Table A.1 in the Appendix.
2. Information regarding the users who authored each tweet (i.e. username, biography, url of the profile image, number of user tweets, number of followers, etc.). The collected fields can be seen in Table A.2 in the Appendix. With this we intend to broaden the analysis by modeling the user who has posted each tweet.
3. Each user's last 200 tweets, complemented with the information from point 1. This allows us to model the types of contributions that each user makes on a regular basis.
4. The user profiles mentioned or retweeted by each author in those 200 tweets, so that we can learn about their environment.

All this information is the base on which the attributes of *SocialGraph* are built. Below we describe its construction process.

3.2.1. Constructing the graph and calculating centrality measures

Using a *Neo4j* database,[17] we build a graph with three types of nodes:

User: node that collects all of the user's information.

Tweet: node that collects all the information related to tweets.

Multimedia: node that collects the url referring to the multimedia content or link (to news) that is shared within a tweet.

And three types of links between them: Quoted, Retweeted or Shared.

We then proceed to compute centrality measures in the graph, or in other words, in the user network (see Table A.3 in the Appendix). Given that, centrality measures have showed to be effective at quantifying the relative importance of actors in a social network (Grando, Noble, & Lamb, 2016; Rajeh, Savonnet, Leclercq, & Cherifi, 2020). For example, a node's ability to influence others is affected much more by its strategic placement within a social network than by the number of followers it has.

3.2.2. Summary statistics

We analyze the information downloaded through Twitter's API and infer a series of new characteristics in order to get a better overall picture of each user. These characteristics are obtained via:

Counting: In this case, we only perform basic statistical operations on the total number of tweets downloaded per user (e.g., the number of times the user's tweets are retweeted, the number of bad words per tweet, the average number of tweets per day, the number of hashtags used, the number of user errors, etc.).

Clustering: where we group the analyzed content and extract the most relevant clusters (i.e top 6 of most shared domains, top 10 of most enabled places, top 5 of most retweeted users, etc.)

Modeling: attributes such as the number of negative, positive, or neutral tweets, the categories to which the image of the user profile belongs, the top 15 topics of each user, and so on are inferred using ad hoc designed classifiers.

Tables A.4–A.6 in the Appendix describe each of these characteristics, as well as the methods used to extract them.

[17] https://neo4j.com last accessed December 24, 2022

3.2.3. Transforming and coding

To be part of the input of any model we must transform the set of characteristics into a set of attributes. Each of the characteristics' tables (Tables A.1 to A.6) indicates the type of variable associated with each characteristic, these can be grouped into:

- Numeric: they have been standardized using *StandardScaler*[18] so that the distribution has a mean value of 0 and a standard deviation of 1.
- Categorical: An extensive transformation has been carried out under a topic modeling technique. For these variables, it is necessary to summarize their textual content, for example, the topics most used by the user in their tweets or the topic that best encompasses their profile description. For this work, three topic modeling techniques have been tested using *Gensim*[19]: LDA, LSI and HDP. Generally, LDA is the best model for topic modeling, but in the present case with short texts, especially when you do not want to specify topics beforehand, HDP can offer a much more consistent solution (Sroka, 2020). After a study of its performance (out of the scope of this paper), this information has been corroborated, which is why HDP has been used as a library since it offers greater coherence in the results.

Tables A.7 and A.8 in the Appendix, detail each of the attributes generated in *SocialGraph*, what characteristic is used to build it from the Tables A.1 to A.6, what method and transformation is used, what values it reaches, and a description thereof.

In summary, we define the set of attributes inside *SocialGraph* as :

$$SocialGraph = X_{\text{profile}} \times X_{\text{activity}} \times X_{\text{centrality}} \tag{1}$$

Where:

- $X_{\text{profile}} \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_4 \times \mathbb{Z}_4 \times \mathbb{Z}_{20} \times \mathbb{Z}_3 \times \mathbb{Z}_9$
  Denotes the space of variables associated with the information intrinsic to a user's profile (name, type of image ...). The space is made up of 7 categorical variables, each of which can take the number of values i associated with the multiplicative group Z_i.
- $X_{\text{activity}} \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_4 \times \mathbb{Z}_{12} \times \mathbb{Z}_3 \times \mathbb{Z}_{12} \times \mathbb{Z}_{12} \times \mathbb{R}^{61}$
  Denotes the space of variables associated with the user's activity in the social network and its aggregate statistics (percentage of tweets every hour, number of total tweets ...). The space is made up of 7 categorical variables, each of which can take the number of values i associated with the multiplicative group Z_i associated and 61 numerical variables.
- $X_{\text{centrality}} \cong \mathbb{R}^7$
  Denotes the space made up of 7 numerical variables associated with the centrality measures relative to the user.

As a result, *SocialGraph* characteristics' dimension is:

$$|SocialGraph| = |X_{\text{profile}}| + |X_{\text{activity}}| + |X_{\text{centrality}}| = 7 + 68 + 7 = 82 \tag{2}$$

3.3. SocialHaterBERT

In order to improve on previous algorithms that only used the text of the tweet to be analyzed as input, *SocialHaterBERT* is created as a multimodal model that combines textual classifiers with social network characteristics. As a result, *HaterBERT*'s classifier after experimental optimization of its parameters (described next in Section 4.1) and *SocialGraph* after an experimental attribute selection (described next in Section 4.3) form the foundation of *SocialHaterBERT*.

[18] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html, last accessed December 24, 2022

[19] https://radimrehurek.com/gensim/, last accessed December 24, 2022

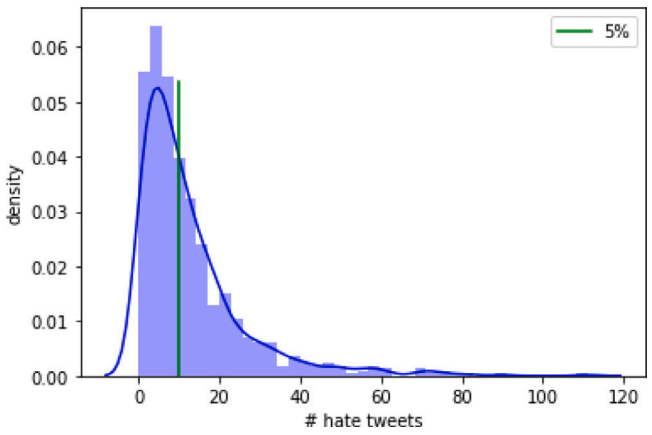

Fig. 5. *ACHaterNet* users' hate distribution. 5% is chosen as the cut-off point.

Table 11
HatEval in english Dataset Division (Basile et al., 2019) (13,000 tweets).

	Train		Test	
	Women	Immigrants	Women	Immigrants
Hate	44.44%	39.76%	42%	42%
Non hate	55.56%	60.24%	58%	58%

Table 12
Comparison between *HaterBERT* and (MacAvaney et al., 2019) using HatEval (en) dataset (Table 11).

Model	Author	Precision	Recall	F1	AUC	Accuracy	Errors
BERT	MacAvaney et al. (2019)	–	–	0.7481	–	0.7470	–
HaterBERT	This proposal	0.7816	0.7887	**0.7799**	0.7877	0.7810	219

In summary, we define the set of attributes inside *SocialHaterBERT* as :

$$SocialHaterBERT = X_{\mathcal{T}} \times X_{\text{profile}} \times X_{\text{activity}} \times X_{\text{centrality}} \times Y_{\mathcal{L}} \quad (3)$$

Where:

- $X_{\mathcal{T}}$ Denotes the text of the associated tweet to classify.
- $X_{\text{profile}} \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_4 \times \mathbb{Z}_4 \times \mathbb{Z}_{20} \times \mathbb{Z}_3 \times \mathbb{Z}_9$
  Denotes the space of variables associated with the information intrinsic to the authors's profile.
- $X_{\text{activity}} \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_4 \times \mathbb{Z}_{12} \times \mathbb{Z}_3 \times \mathbb{Z}_{12} \times \mathbb{Z}_{12} \times \mathbb{R}^{30}$
  Denotes the space of variables associated with the author's activity inside the social network.
- $X_{\text{centrality}} \cong \mathbb{R}^7$
  Denotes the space of variables associated with the centrality measures relative to the author.
- $Y_{\mathcal{L}} \cong \mathbb{Z}_2$ Denotes the classification label corresponding to the record (Tweet) that is defined as Hate (1) / Non-Hate (0).

As a result, *SocialHaterBERT* characteristics' dimension is:

$$\begin{aligned} |SocialHaterBERT| &= |X_{\mathcal{T}}| + |X_{\text{profile}}| + |X_{\text{activity}}| + |X_{\text{centrality}}| + |Y_{\mathcal{L}}| \\ &= 1 + 7 + 37 + 7 + 1 = 53 \end{aligned} \quad (4)$$

For the construction of the model, we make use of the *Multimodal Transformers*[20] library, which is used to incorporate multimodal data on text data for classification and regression tasks. In this way, a pre-trained transformer along the combination module's parameters and the transformer are trained as a supervised task (see Fig. 2).

[20] https://multimodal-toolkit.readthedocs.io/en/latest/, last accessed December 24, 2022

Table 13
Classification results for the detecting whether a user is hater or not.

Model	Precision	Recall	F1	Accuracy
Naive Bayes	0.8677	0.8618	0.8563	0.8565
Logistic Regression	0.8995	0.8831	0.8858	0.8879
KNN	0.6794	0.6713	0.6708	0.6771
SVM	0.8801	0.8709	0.8728	0.8744
Random Forest	**0.9958**	**0.9952**	**0.9955**	**0.9955**
MLP	0.8411	0.8296	0.8314	0.8341

Table 14
Detail of the hyperparameters tested for *SocialHaterBERT*.

Hyperparameters	Alternatives
Epochs	[1, 2, 3, 4, 5, 6,**7**, 8, 9]
Learning Rate	[1e−5, **2e-5**, 3e−5, 4e−5, 5e-5]
Activation	ReLU
Batch Size	[4, 6, **16**, 32]
Epsilon	[**1e-5**, 1e-12]

SocialHaterBERT's architecture is as follows: To distribute the data for classification, the text, numeric, categorical and prediction columns are specified in a dictionary. After this, *BertTokenizer* and *BertForSequenceClassification* are instantiated respectively, which also allows the *Fine-Tuning* of it. Then, in the *Combining Module* (shown in Fig. 3) a hidden two-layer MLP is created with a *ReLu* activation function, as it improves training. Finally, before the output layer (Fig. 3) results are combined using the logical sum of the attributes, as it proved to be the best combination option (see Section 4.4 Table 15).

4. Experiments and results

This section details and analyzes the results obtained in different experiments designed to test this paper's posed hypotheses and carried out respectively on *HaterBERT*, *SocialGraph* and *SocialHaterBERT*.

- To test **H1** and **H1.1**, a comparison with the state of the art in Spanish and English, as well as the corresponding hyperparameter configuration, is presented for the case of *HaterBERT*.
- Second, the dataset used to conduct the following experiments is described, along with an argument for the lack of alternatives.
- To test **H2**, results demonstrating the utility of *SocialGraph* in identifying hater profiles are presented.
- Finally, to test **H3**, the hyperparameters assessed for *SocialHaterBERT*, as well as those that best adapt to the model, are illustrated, along with a comparison with *HaterBERT*.

It is important to note that the results are evaluated using the F1-Score as the primary metric, given that false negatives and positives are more important in topics such as detecting hate (Lipton, Elkan, & Narayanaswamy, 2014), though other metrics are also considered.

4.1. Haterbert : Optimization and comparison with the state of the art

The experiments performed with *HaterBERT*, as well as a comparison to the state of the art, are detailed below. Is worth noting that, for the sake of brevity, only the best results are shown. Table 6 shows *HaterBERT*'s optimized hyperparameters.

To begin, we compare *HaterBERT* to its predecessor, *HaterNet*, using the same validation method (LOOCV) and dataset as in the original paper (see Table 7).

As it can be seen, *HaterBERT* significantly improves the outcome. Fig. 4 illustrates the confusion matrix for *HaterBERT*'s optimal hyperparameter configuration, where it is reflected that only 5 out of 6000 tweets are misclassified. Although the relevance of the results should be assessed, it is important to note that LOOCV trains the model using $n-1$ data points, which involves practically the entire dataset. In practical terms, what it does is a better fit of the model to the available data, which can lead to a greater risk of overfitting.

Table 15
Detail of strategies carried out for the Combining Module of *SocialHaterBERT*.

Experiment	Description	AUC	Recall	Precision	F1
SHAT-1	Text only	0.7791	0.6943	0.5121	0.6222
SHAT-2	Attention-based sum before the output layer	0.8394	0.7653	0.7364	0.7501
SHAT-3	Weighted sum before the output layer	0.8536	0.5952	0.8928	0.7142
SHAT-4	Logical sum before the output layer	0.8923	0.7826	0.7031	**0.8023**

Table 16
Comparison between *HaterBERT* and *SocialHaterBERT*.

Model	Accuracy	F1	AUC	Precision	Recall
HaterBERT	0.8343	0.7645	0.7354	0.8506	0.7354
SocialHaterBERT	0.8472	**0.8023**	0.8923	0.7031	0.7826

As a consequence, more comparisons with respect to state of the art methods are made using other more robust validation strategies, i.e. stratified with different divisions and k-foldCV. Specifically, a comparison with the other two works in Spanish is made.

Table 8 details the results comparing *HaterBERT* with those reported in Aluru et al. (2020), where they use mBERT. To perform a correct comparison, the same experimental configuration reflected in Aluru et al. (2020) is used: a 70% training, 20% test, and 10% validation stratified division of the *HaterNet*'s dataset. This table shows that the results reported in Aluru et al. (2020) are outperformed by *HaterBERT*. Although mBERT may provide good results on a broad level, the problem is that it relies on pre-trained transformers on a set of monolingual corpora from various languages, so it lacks a detection mechanism for the language in question, and the token can easily be confused with another language. BETO (*HaterBERT*'s base) was, on the other hand, pre-trained with a dataset specifically in the Spanish language, making it far more suitable for Spanish datasets.

Tables 9 and 10 replicate the experiments reported in Plaza-del Arco et al. (2021) (to facilitate comparison), and show a comparison of *HaterBERT*'s performance against said paper using 10k-FoldCV on the HatEval in Spanish and *HaterNet*'s datasets, respectively. Is worth noting that Plaza-del Arco et al. (2021) does not go into detail about implementation or fine-tuning, despite the fact that they use BETO without uppercase sensitivity. Moreover, unlike Plaza-del Arco et al. (2021), this research began with the hypothesis that the sensitivity of capital letters can provide more information about the polarity of a tweet, thus supporting the problem of hate detection, because users can use them to emphasize their ideas. In addition, the optimization of hyperparameters (see Table 6) and the correction of the text to avoid the leet alphabet have also been carried out differently. All of these nuances are likely reasons for the performance differences between the two BETO-based algorithms, where *HaterBERT* outperforms Plaza-del Arco et al. (2021) approach.

Note that unlike Aluru et al. (2020) and Plaza-del Arco et al. (2021), in this work a $maxlength = 256$ has been chosen in preprocessing and padding. While this makes the algorithm take longer to process the input, it may be another reason why *HaterBERT* shows better results than the other two BERT-based proposals.

Finally, for the sake of reproducibility, even though our goal is to implement a tool for the Spanish authorities, we have performed additional experiments with an English dataset comparing *HaterBERT* with a renowned state of the art BERT-based classifier (MacAvaney et al., 2019).[21] For that, in Table 11 we describe the dataset used in said approach and in Table 12 we present the comparison. Note that MacAvaney et al. (2019) does not detail the hyperparameters configuration, and that in this proposal we chose: 3 epochs a batch size of 16 and a learning rate of $2e^{-}5$.

In general, *HaterBERT* improves on all of the previously obtained results in Spanish, confirming Hypothesis 1 and 1.1, as well as results based on BERT in English. As a result, in the following sections, we will use *HaterBERT* as the base classifier to improve.

[21] Given that we have already proved that *HaterBERT* outperforms *HaterNet* that to its publication date was proved to be the best classifier.

4.2. ACHaterNet Dataset

Because text queries for tweets older than 14 days cannot be made without a company key for the Twitter API, the id of the tweet, not just its corresponding text, is required for the extraction of characteristics described in Section 3.2. Among all the datasets reviewed in this work, the *HaterNet* dataset is the only one that meets the required characteristics and therefore, the one selected to perform the experiment setup. Furthermore, all existing datasets (including *HaterNet*) have a handicap (Ribeiro et al., 2018). In fault, because of inappropriate content, many tweets and users of the original datasets are deleted over time. Therefore, of the 6,000 tweets of the original *HaterNet* dataset, only 3,391 are available to date, from which we can extract all the data described in Eq. (1). This reduced dataset is the one we will work with from now on and will be referred to as *ACHaterNet*.[22] Note that, in order to compare the result of *SocialHaterBERT*, it is necessary to retrain the base model of *HaterBERT* with *ACHaterNet* as shown in Table 16.

4.3. Socialgraph : for haters detection

To test Hypothesis 2 and see if (and which) of the characteristics collected in *SocialGraph* are relevant to the hate classification, we use traditional binary Machine Learning classification models (i.e., Naive Bayes, Support Vector Classification, Logistic Regression, K-Nearest Neighbors and Random Forests) and a MLP (Shobha & Rangaswamy, 2018) to train a classifier that spots haters profiles. In these models the input X is the collection of attributes described in Eq. (1) and the class to predict Y is created using *HaterBERT* on the set of 200 tweets downloaded from each user (see Section 3.2), such that:

$$\text{hater} = \begin{cases} 1 & \text{if the n° of user tweets classified by} \\ & HaterBERT \text{ as hateful} > 5\%) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Where the value 5% has been chosen after studying the distribution in *ACHaterNet* of all users' hateful tweets, as seen in Fig. 5 representing the upper tail and contrasted with ONDOD's experts.

Experiment results can be found in Table 13, where Random Forest clearly outperforms the rest. From these results, where the $F1 = 0.9955$, we can conclude that social network characteristics are indeed helpful for spotting haters, thus fulfilling Hypothesis 2. In addition, Random Forest serves as a variable selection method. After this experimentation we reduce *SocialGraph*'s attribute size for classification tasks from 82 to 51 as shown in the Eqs. (3) and (4). The discarded variables are those related to `activity_hourly_X` ($X \in [0-23]$) and `activity_weekly_X` ($X \in [0-6]$) (highlighted in red in Tables A.7 and A.8).

4.4. SocialHaterBERT : Experiment Results

Finally, in order to demonstrate whether the model proposed in this work, *SocialHaterBERT* supposes an improvement in hate classifiers, we proceed to test, as in *HaterBERT*, different hyperparameter configurations. Table 14, lists these hyperparameters, with the bolded values indicating those that improved the model's performance.

[22] This database, as well as all of the developed code in this paper, will be available at the time of publication for reproducibility at https://github.com/glorelvalle/hs-project.

Table A.1
Tweet extraction attributes.

Attribute	Type	Description
user_id	int	user identifier
screen_name	str	Username
tweet_id	int	tweet identifier
tweet_text	str	tweet text
tweet_creation_at	datetime	tweet creation date
n_favs	int	number of favorites
n_rts	int	number of retweets
is_rt	boolean	the tweet is a retweet
rt_id_user	int	id of the retweeted user
rt_id_status	int	id of the retweeted tweet
rt_text	str	text of the retweeted tweet
rt_creation_at	datetime	creation date of the retweeted tweet
rt_fav_count	int	number of favorites (if is retweeted)
rt_rt_count	int	number of retweets (if is retweeted)
is_reply	boolean	the tweet is a response
reply_id_status	int	id of the tweet being replied
reply_id_user	int	user id to which it responds
is_quote	boolean	the tweet is a quote from another
quote_id_status	int	id of the quoted tweet
quote_id_user	int	id of the quoted user
quote_text	str	text of the quoted tweet
quote_creation_at	datetime	creation date of the quoted tweet
quote_fav_count	int	number of favorites quoted
quote_rt_count	int	number of retweets quoted

Table A.2
User extraction attributes.

Attribute	Type	Description
user_id	int	user id
uname	str	user profile name
virtual	boolean	virtual node
screen_name	str	Username
description	str	biography or description
location	str	location if any
verified	boolean	Verified account
profile_image_url	str	profile picture url
default_profile	boolean	profile update
default_image_profile	boolean	profile picture update
geo_enabled	boolean	real location enabled
created_at	datetime	account creation date
statuses_count	int	number of user tweets
listed_count	int	number of lists
followers_count	int	number of followers
followees_count	int	number of followed
favorites_count	int	number of favorites

Table A.3
Centrality measures in *SocialGraph*.

Measure	Description
betweenness	computes the shortest path to the graph's centrality
eigenvector	measure of a node's influence on the network
in-degree	number of edges pointing to node
out-degree	number of edges pointing outside the node
clustering	fraction of pairs of neighboring nodes adjacent to each other
degree	number of edges adjacent to the node
closeness	average distance of all reachable nodes to node

Finally, the performance of *SocialHaterBERT* trained with *ACHaterNet* for different attribute combination strategies before the output layer is shown in Table 15, where the SHAT-4 strategy is the one that performs best. Note that the experiments have been performed with a stratified sample of 80-10-10.

After that, in Table 16 we proceed to compare the results of *SocialHaterBERT* and *HaterBERT*, where *SocialHaterBERT* outperforms the latter in 4%, thus, demonstrating Hypothesis 3.

5. Conclusions and future work

In an already polarized world, social networks are a double-edged sword with the appearance of phenomena such as hate speech. In the present work, its presence on Twitter has been detected and analyzed. For this, a base algorithm, *HaterBERT*, has been designed, which improves current Spanish classifiers' results by 3%–27%.

Furthermore, the presence of hate speech on Twitter has been analyzed through an extensive study that has served to extrapolate essential characteristics of it. To do this, a procedure has been developed for the extraction and manipulation of these characteristics, *SocialGraph*, which has been demonstrated with an F1 of 99% and a Random Forest classifier that provides valuable data for the identification of hater profiles.

These findings lead to the development of *SocialHaterBERT*, a novel multimodal model that combines categorical and numerical variables from the social network with text input from tweets, providing not only a new way to understand hate speech on social media in general but also demonstrating how the context of social media improves textual classification, which is the most valuable contribution of this paper. In particular, we achieved a 4% improvement over the *HaterBERT*'s base algorithm and a 19% improvement over our original algorithm, *HaterNet* (Pereira-Kohatsu et al., 2019).

In terms of practical contributions, the model presented in this study has been developed in collaboration with ONDOD. This research has confirmed that the contribution of the users' characteristics is key to identifying hate in the network, thus, this work has opened a field of study limited to date to textual bounds and paving a way for future research. The classifier resulting from this work can be embedded in a continuous online monitoring tool. The competent authorities can analyze the hate messages identified by the tool to detect hate spikes, triggers, and also to devise mitigation strategies.

Future research should look into aspects such as a review of hate's history and evolution on the network, trends, public and anonymous users affected by it, and aggressors' profiles, with the goal of encouraging the discovery of relationships with the dissemination and virality of hate on social networks. Following that, interactions with one another might be investigated, resulting in an extension of *SocialGraph*'s characteristics and a prediction of each tweet's virality.

CRediT authorship contribution statement

Gloria del Valle-Cano: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft. **Lara Quijano-Sánchez:** Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization, Supervision. **Federico Liberatore:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Jesús Gómez:** Conceptualization, Resources, Project administration.

Data availability

The authors do not have permission to share data.

Acknowledgments

We would like to thank Mario Hernandez Ramos, Professor of Constitutional Law, Universidad Complutense de Madrid, Spain, and Head of the Spanish delegation for the Committee on Artificial Intelligence of the Council of Europe (CAHAI) for revising the legal and ethical aspects of this research, specially those concerning the compliance of the European Union General Data Protection Regulation (GDPR). The research of Quijano-Sánchez was conducted with financial support from the Spanish Ministry of Science and Innovation, grant PID2019-108965GB-I00. The research of Liberatore is partially funded by the European Commission's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie, grant number MSCA-RISE 691161 (GEOSAFE), and the Government of Spain, grant MTM2015-65803-R. All the financial support is gratefully acknowledged.

Table A.4
SocialGraph summary statistics attributes via counting.

Attribute	Type	Description
status_retrieving	int	number of saved tweets
status_start_day	datetime	start date of tweet extraction
status_end_day	datetime	end date of tweet extraction
status_average_tweets_per_day	float	average tweets per day
activity_hourly_X	int	number of tweets at each day hour, 24 attributes being X ∈ [00-23]
activity_weekly_X	int	number of tweets at each week day, 7 attributes being X ∈ [0-6]
rt_count	int	total number of saved tweets
geo_enabled_tweet_count	int	number of tweets with geolocation enabled
num_hashtags	int	number of hashtags used
num_mentions	int	number of mentions
num_urls	int	number of domains shared by the user
baddies	list(str)	bad words or insults used by the user
n_baddies	int	number of baddies
n_baddies_tweet	float	number of baddies per tweet
len_status	float	average tweet length
times_user_quotes	int	number of times other users are quoted
num_rts_to_tweets	int	number of times user tweets are retweeted
num_favs_to_tweets	int	number of times user tweets are favorite
leet_counter	int	number of times the user uses the leet alphabet

Table A.5
SocialGraph summary statistics attributes via clustering.

Attribute	Type	Description
top_languages	dict(language(str), account(int))	top 5 languages most used by the user by number of tweets
top_sources	dict(vía(str), account(int))	top 5 ways to tweet by number of tweets
top_places	dict(place(str), account(int))	top 10 places most enabled by the user by number of tweets
top_hashtags	dict(hashtag(str), account(int))	top 10 hashtags most used by the user by number of tweets
top_retweeted_users	dict(user(str), account(int))	top 5 most retweeted users by the user by number of tweets
top_mentioned_users	dict(user(str), account(int))	top 5 users most mentioned by the user by number of tweets
top_referenced_domains	dict(dominio(str), account(int))	top 6 domains most shared by the user by number of tweets

Table A.6
SocialGraph summary statistics attributes via modeling.

Attribute	Type	Classifier	Source	Description
categories_profile _image_url	dict(dict(category, score, hierarchy=None))	Client Watson Visual Recognition (IBM)	VisualRecognitionV3	user's profile image categories
negatives positives neutral	int	Sentiment analysis classifier (transformers)	finiteautomata/beto-sentiment-analysis Pérez, Giudici, and Luque (2021)	negatives number of positives number of neutral
negatives_score positives_score neutral_score	float	Sentiment analysis classifier (transformers)	finiteautomata/beto-sentiment-analysis Pérez et al. (2021)	negatives score positives score neutral score
hate non_hate	int	Ad hoc classifier	*HaterBERT*	number of hate tweets number of non hate tweets
hate_score non_hate_score	float	Ad hoc classifier	*HaterBERT*	hate score non hate score
top_categories	dict(category(str), account(int))	Spanish Category classifier (Python Library)	subject_classification_spanish	top 15 twwet categories
misspelling_counter	int	Spanish Spell checker	pyspellchecker (Python Library)	number of errata committed by the user

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

In the following Tables we describe the characteristics extracted from user's Twitter profiles and tweets posted for its inclusion in *SocialGraph*.

Appendix B

A search is conducted according to the scheme presented below in order to thoroughly review the literature related to the current proposal. First, a set of keywords is chosen, which is then organized in a query, which searches various bibliographic search databases. These keywords are supplemented with variations such as plurals or related words related to the search. Table A.9 displays the query sent to Mendeley.

A query for the IEES, WOS SCOPUS and Mendeley databases is built with them divided into multiple blocks, with several versions that obey their respective syntax. Given the findings and the fact that hate speech identification is a relatively young field, it is thought reasonable to

Table A.7
Detail of the categorical variables in *SocialGraph*. NC = does not change.

Variable	Original variable(s)	Group	Method	Categories	Description
verified	NC	profile	boolean classification	0: No, 1: Yes	user is verified
hater	NC	activity	boolean classification	0: No, 1: Yes	user has more than 5% hate tweets
vecino_hater	NC	activity	boolean classification	0: No, 1: Yes	the user has at least one neighbor with more than 5% hate tweets
profile_changed	default_profile	profile	boolean classification	0: No, 1: Yes	the user ever updated his profile
clase_NER	screen_name + uname	profile	NER tag search (Spacy)	0: PER, 1: MISC, 2: ORG, 3: UND	tipo de nombre
clase_DESCR	description	profile	cleaning (NLTK) + Topic Modeling (Gensim)	0: opinion, 1: studies, 2: politics, 3: activities	description type
clase_LOC	location	profile	cleaning + ad hoc dict + pycountry	0-19: geographic world areas or provinces in the case of Spain	geographical area enabled by the user
clase_FECHA	created_at	profile	division into three regions	0: < 2015, 1: [2015–2019], 2: > 2019	time of user creation
clase_IMG	categories _profile_image _url	profile	Topic Modeling (Gensim)	0: people, 1: clothing, 2: building, 3: animal, 4: nature, 5: technology, 6: sports, 7: objects, 8: food	profile image type
clase_HASHTAGS	top_hashtags	activity	Correlation matrix + Topic Modeling	0: politics, 1: press, 2: sports, 3: others	hashtag type
clase_CATS	top_categories	activity	Topic Modeling (Gensim)	0: Spain, 1: culture, 2: art, 3: society 4: cartoons, 5: Catalonia, 6: graphical arts, 7: drawings, 8: opinion, 9: illustrations, 10: politics, 11: others	most repeated categories by the user in tweets
clase_DOMS	top_referenced _domains	activity	wikipedia + Topic Modeling	0: social networks, 1: information, communication and news, 2: entertainment	type of domain most shared by the user
clase_RTSCAT	top_retweeted _users	activity	Topic Modeling (Gensim)	0: Spain, 1: culture, 2: art, 3: society 4: cartoons, 5: Catalonia, 6: graphical arts, 7: drawings, 8: opinion, 9: illustrations, 10: politics, 11: others	most retweeted user type
clase_MENCAT	top_mentioned _users	activity	Topic Modeling (Gensim)	0: Spain, 1: culture, 2: art, 3: society 4: cartoons, 5: Catalonia, 6: graphical arts, 7: drawings, 8: opinion, 9: illustrations, 10: politics, 11: others	most mentioned user type

Table A.8
Detail of the numerical variables in *SocialGraph*. NC = does not change.

Variable	Original Variable(s)	Group	Method	Description
n_LESP	top_languages	activity	Ad hoc function	percentage of hate tweets in Spanish
n_LENG	top_languages	activity	Ad hoc function	percentage of hate tweets in English
n_LOTR	top_languages	activity	Ad hoc function	percentage of hate tweets in other language (no Spanish or English)
activity_hourly_X	NC	activity	Ad hoc function	percentage of tweets per hour (X=24)
activity_weekly_X	NC	activity	Ad hoc function	percentage of tweets per week day (X=7)
negatives	NC	activity	Ad hoc function	negative connotation percentage of tweets
positives	NC	activity	Ad hoc function	positive connotation percentage of tweets
neutral	NC	activity	Ad hoc function	neutral connotation percentage of tweets
n_hate	NC	activity	Ad hoc function	hate tweets percentage
n_nohate	NC	activity	Ad hoc function	non hate tweets percentage
n_baddies	NC	activity	Ad hoc function	percentage of baddies per tweet
eigenvector	NC	centrality	–	eigenvector score
in_degree	NC	centrality	–	in degree score
out_degree	NC	centrality	–	out degree score
degree	NC	centrality	–	degree score
clustering	NC	centrality	–	clustering score
closeness	NC	centrality	–	closeness score
betweenness	NC	centrality	StandardScaler	number of shortest paths to it
status_average_tweets _per_day	NC	activity	StandardScaler	average number of times user tweets per day

(*continued on next page*)

Table A.8 (*continued*).

Variable	Original Variable(s)	Group	Method	Description
times_user_quotes	NC	activity	StandardScaler	number of times user quotes others
negatives_score	NC	activity	–	mean score of negative tweets
positives_score	NC	activity	–	mean score of positive tweets
neutral_score	NC	activity	–	mean score of neutral tweets
hate_score	NC	activity	–	score media de tweets de odio
no_hate_score	NC	activity	–	score media de tweets de no odio
statuses_count	NC	activity	StandardScaler	total number of tweets
followers_count	NC	activity	StandardScaler	total number of tweets followers
followees_count	NC	activity	StandardScaler	total number of tweets followees
favorites_count	NC	activity	StandardScaler	total number of tweets favorites
listed_count	NC	activity	StandardScaler	number of lists user is on
num_hashtags	NC	activity	StandardScaler	number of hashtags used
rt_count	NC	activity	StandardScaler	total number of retweets
num_mentions	NC	activity	StandardScaler	number of mentions made
num_urls	NC	activity	StandardScaler	number of shared urls
len_status	NC	activity	StandardScaler	average tweet length
num_rts_to_tweets	NC	activity	StandardScaler	number of times user tweets are retweeted
num_favs_to_tweets	NC	activity	StandardScaler	number of times user tweets are favorited
misspelling_counter	NC	activity	StandardScaler	number of times user makes mistakes or errors
leet_counter	NC	activity	StandardScaler	number of times user uses leet alphabet

Table A.9
Final query made to Mendeley. Each row represents one of the ANDed blocks of the query, corresponding to its search section.

Section	Block
TITLE–ABS–KEY	"hate speech detection" OR "counter speech detection"
TITLE–ABS–KEY	"social network" OR "Twitter" OR "social media" OR "social graph" OR "social graphs"
ALL	"hate" OR "hater" OR "haters" OR "hateful user" OR "hateful users" OR "aggressive" OR "offensive"
ALL	"multimodal" OR "tabular"
TITLE–ABS–KEY	"misogyny" OR "against women" OR "xenophobia" OR "racism" OR "immigrants" OR "cyberbullying"
ALL	"BERT"

include only the articles published after 2016. This first search yields 216 results in WOS, 173 in SCOPUS, 250 in Mendeley, and 56 in IEES, all of which have matches. After eliminating duplicate results, a quick scan of each of them is conducted to ensure that they are indeed related to the project, resulting in the discovery of 47 articles.

References

Aluru, S. S., Mathew, B., Saha, P., & Mukherjee, A. (2020). Deep learning models for multilingual hate speech detection. ArXiv preprint arXiv:2004.06465.

Arango, A., Pérez, J., & Poblete, B. (2019). Hate speech detection is not as easy as you may think: A closer look at model validation. In *Proceedings of the 42nd international ACM sigir conference on research and development in information retrieval* (pp. 45–54).

Plaza-del Arco, F. M., Molina-González, M. D., Ureña-López, L. A., & Martín-Valdivia, M. T. (2021). Comparing pre-trained language models for Spanish hate speech detection. *Expert Systems with Applications*, *166*, Article 114120.

Badjatiya, P., Gupta, S., Gupta, M., & Varma, V. (2017). Deep learning for hate speech detection in tweets. In *Proceedings of the 26th international conference on world wide web companion* (pp. 759–760).

Badjatiya, P., Gupta, M., & Varma, V. (2019). Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In *The world wide web conference* (pp. 49–59).

Basile, V., Bosco, C., Fersini, E., Debora, N., Patti, V., Pardo, F. M. R., et al. (2019). Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *13th international workshop on semantic evaluation* (pp. 54–63). Association for Computational Linguistics.

Battistelli, D., Bruneau, C., & Dragos, V. (2020). Building a formal model for hate detection in french corpora. *Procedia Computer Science*, *176*, 2358–2365.

Bustos Martínez, L., De Santiago Ortega, P. P., Martínez Miró, M., & Rengifo Hidalgo, M. S. (2019). Hate speeches: an epidemic that spreads in the network. State of the art on racism and xenophobia in social networks (discursos de odio: una epidemia que se propaga en la red. Estado de la cuestión sobre el racismo y la xenofobia en las redes sociales). *Revista Mediaciones Sociales,(18)*, 25–42.

Cañete, J., Chaperon, G., Fuentes, R., Ho, J.-H., Kang, H., & Pérez, J. (2020). Spanish pre-trained BERT model and evaluation data. In *PML4DC at ICLR 2020*.

Celli, F., Lai, M., Duzha, A., Bosco, C., & Patti, V. (2021). Policycorpus XL: An Italian corpus for the detection of hate speech against politics. In *CEUR workshop proceedings (Vol. 3033)*. Cited by: 0. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85121248469&partnerID=40&md5=9cb08868914b9376548c287248c220be.

da Silva, F. L. V., & de Freitas, L. A. (2022). Brazilian portuguese hate speech classification using bertimbau. In *Proceedings of the international florida artificial intelligence research society conference, FLAIRS (Vol. 35)*. Cited by: 0; All Open Access, Hybrid Gold Open Access. http://dx.doi.org/10.32473/flairs.v35i.130594. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85131117521&doi=10.32473%2fflairs.v35i.130594&partnerID=40&md5=5095185743832d8bec7f74d187e9caec.

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media (Vol. 11) (No. 1)*.

de Oliveira, G. A., de Oliveira Albuquerque, R., de Andrade, C. A. B., de Sousa, R. T., Orozco, A. L. S., & Villalba, L. J. G. (2020). Anonymous real-time analytics monitoring solution for decision making supported by sentiment analysis. *Sensors*, *20*, http://dx.doi.org/10.3390/s20164557.

Defersha, N. B., Kekeba, K., & Kaliyaperumal, K. (2021). Tuning hyperparameters of machine learning methods for afan oromo hate speech text detection for social media. (pp. 596–604). http://dx.doi.org/10.1109/ICCCT53315.2021.9711850, Cited by: 0. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85126736805&doi=10.1109%2flCCCT53315.2021.9711850&partnerID=40&md5=f993f43a9fabffc7d647aa0e9b9ba932,

Del Vigna12, F., Cimino23, A., Dell'Orletta, F., Petrocchi, M., & Tesconi, M. (2017). Hate me, hate me not: Hate speech detection on facebook. In *Proceedings of the first Italian conference on cybersecurity* (pp. 86–95).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., & Bhamidipati, N. L. (2015). Hate speech detection with comment embeddings. In *Proceedings of the 24th international conference on world wide web*.

ElSherief, M., Kulkarni, V., Nguyen, D., Wang, W. Y., & Belding, E. (2018). Hate lingo: A target-based linguistic analysis of hate speech in social media. arXiv:1804.04257.

Fersini, E., Rosso, P., & Anzovino, M. (2018). Overview of the task on automatic misogyny identification at IberEval 2018.. *IberEval@ SEPLN*, *2150*, 214–228.

Florio, K., Basile, V., Polignano, M., Basile, P., & Patti, V. (2020). Time of your hate: The challenge of time in hate speech detection on social media. *Applied Sciences*, *10*(12), 4180.

Fortuna, P., & Nunes, S. (2018). A survey on automatic detection of hate speech in text. *ACM Computing Surveys*, *51*(4), 1–30.

Founta, A. M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., et al. (2018). Large scale crowdsourcing and characterization of twitter abusive behavior. In *Twelfth international AAAI conference on web and social media*.

Gambäck, B., & Sikdar, U. K. (2017). Using convolutional neural networks to classify hate-speech. In *Proceedings of the first workshop on abusive language online* (pp. 85–90). Vancouver, BC, Canada: Association for Computational Linguistics.

Garland, J., Ghazi-Zahedi, K., Young, J.-G., Hébert-Dufresne, L., & Galesic, M. (2020a). Countering hate on social media: Large scale classification of hate and counter speech. http://dx.doi.org/10.18653/v1/2020.alw-1.13.

Garland, J., Ghazi-Zahedi, K., Young, J.-G., Hébert-Dufresne, L., & Galesic, M. (2020b). Impact and dynamics of hate and counter speech online. arXiv preprint arXiv:2009.08392.

Gover, A. R., Harper, S. B., & Langton, L. (2020). Anti-Asian hate crime during the COVID-19 pandemic: Exploring the reproduction of inequality. *American Journal of Criminal Justice*, *45*(4), 647–667.

Grando, F., Noble, D., & Lamb, L. C. (2016). An analysis of centrality measures for complex and social networks. In *2016 IEEE global communications conference* (pp. 1–6). IEEE.

Kovács, G., Alonso, P., & Saini, R. (2021). Challenges of hate speech detection in social media. *SN Computer Science*, *2*(2), 1–15.

Kumar, S., & Pranesh, R. R. (2021). TweetBLM: A hate speech dataset and analysis of black lives matter-related microblogs on Twitter. arXiv:2108.12521.

Lipton, Z. C., Elkan, C., & Narayanaswamy, B. (2014). Thresholding classifiers to maximize f1 score. *ArXiv*, 1402–1892.

Loshchilov, I., & Hutter, F. (2017). Decoupled weight decay regularization. ArXiv preprint arXiv:1711.05101.

MacAvaney, S., Yao, H.-R., Yang, E., Russell, K., Goharian, N., & Frieder, O. (2019). Hate speech detection: Challenges and solutions. *PLoS One*, *14*(8), Article e0221152.

Mathew, B., Kumar, N., Goyal, P., Mukherjee, A., et al. (2018). Analyzing the hate and counter speech accounts on twitter. ArXiv preprint arXiv:1812.02712.

Mathur, P., Shah, R., Sawhney, R., & Mahata, D. (2018). Detecting offensive tweets in hindi-english code-switched language. In *Proceedings of the sixth international workshop on natural language processing for social media* (pp. 18–26).

Mayda, I., Demir, Y. E., Dalyan, T., & Diri, B. (2021). Hate speech dataset from Turkish tweets; [türkce tweetlerden nefret söylemi veri seti]. In *Proceedings - 2021 innovations in intelligent systems and applications conference*. Cited by: 0. http://dx.doi.org/10.1109/ASYU52992.2021.9599042. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85123196096&doi=10.1109%2fASYU52992.2021.9599042&partnerID=40&md5=5f7971982aa194755b9604e89a65905e.

Modha, S., Mandl, T., Shahi, G. K., Madhu, H., Satapara, S., Ranasinghe, T., et al. (2021). Overview of the HASOC subtrack at FIRE 2021: Hate speech and offensive content identification in English and indo-aryan languages and conversational hate speech. In *ACM international conference proceeding series*, (pp. 1–3). Cited by: 0; All Open Access, Green Open Access. http://dx.doi.org/10.1145/3503162.3503176. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85124344402&doi=10.1145%2f3503162.3503176&partnerID=40&md5=76fcac9b114d9c88cc1ea94a0b4bcd7a.

Morgante, V. (2021). *Make America hate again: a quantitative analysis on the effects of presidential rhetoric during the Obama and Trump administration* (Ph.D. thesis), Rutgers University-Camden Graduate School.

Mozafari, M., Farahbakhsh, R., & Crespi, N. (2019). A BERT-based transfer learning approach for hate speech detection in online social media. In *International conference on complex networks and their applications* (pp. 928–940). Springer.

Mulki, H., Haddad, H., Bechikh Ali, C., & Alshabani, H. (2019). L-HSAB: A levantine Twitter dataset for hate speech and abusive language. In *Proceedings of the third workshop on abusive language online* (pp. 111–118). Florence, Italy: Association for computational linguistics, http://dx.doi.org/10.18653/v1/W19-3512, URL: https://aclanthology.org/W19-3512.

Müller, K., & Schwarz, C. (2020). From hashtag to hate crime: Twitter and anti-minority sentiment. Available at SSRN 3149103.

Müller, K., & Schwarz, C. (2021). Fanning the flames of hate: Social media and hate crime. *Journal of the European Economic Association*, *19*(4), 2131–2167.

Olteanu, A., Castillo, C., Boy, J., & Varshney, K. (2018). The effect of extremist violence on hateful speech online. In *Proceedings of the international AAAI conference on web and social media (Vol. 12), (No. 1)*.

Paasch-Colberg, S., Strippel, C., Trebbe, J., & Emmer, M. (2021). From insult to hate speech: Mapping offensive language in german user comments on immigration. *Media and Communication*, *9*(1), 171–180, Cited by: 3; All Open Access, Gold Open Access, Green Open Access. http://dx.doi.org/10.17645/mac.v9i1.3399. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85101171824&doi=10.17645%2fmac.v9i1.3399&partnerID=40&md5=bd6d3458e914cbae74cc2927d9ad96f9.

Park, J., & Fung, P. (2017). One-step and two-step classification for abusive language detection on Twitter. In *ALW@ACL*.

Paulson, L. F. (2021). Free to hate: Hate crimes' intertwinement with the evolution of free speech in the United States.

Paz, M. A., Montero-Díaz, J., & Moreno-Delgado, A. (2020). Hate speech: A systematized review. *Sage Open*, *10*(4), Article 2158244020973022.

Pereira-Kohatsu, J. C., Quijano-Sánchez, L., Liberatore, F., & Camacho-Collados, M. (2019). Detecting and monitoring hate speech in Twitter. *Sensors*, *19*(21), 4654.

Pérez, J. M., Giudici, J. C., & Luque, F. (2021). Pysentimiento: A python toolkit for sentiment analysis and socialNLP tasks. arXiv:2106.09462.

Perifanos, K., & Goutsos, D. (2021). Multimodal hate speech detection in greek social media. Preprints.

Poletto, F., Basile, V., Sanguinetti, M., Bosco, C., & Patti, V. (2021). Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, *55*(2), 477–523.

Rajeh, S., Savonnet, M., Leclercq, E., & Cherifi, H. (2020). Investigating centrality measures in social networks with community structure. In *International conference on complex networks and their applications* (pp. 211–222). Springer.

Ribeiro, M. H., Calais, P. H., Santos, Y. A., Almeida, V. A., & Meira Jr, W. (2018). Characterizing and detecting hateful users on twitter. In *Twelfth international AAAI conference on web and social media*.

Salminen, J., Almerekhi, H., Milenković, M., Jung, S.-g., An, J., Kwak, H., et al. (2018). Anatomy of online hate: developing a taxonomy and machine learning models for identifying and classifying hate in online news media. In *Twelfth international AAAI conference on web and social media*.

Shobha, G., & Rangaswamy, S. (2018). Machine learning. In V. Gudivada, & C. Rao (Eds.), *Handbook of statistics (Vol. 38)* (pp. 197–228). Elsevier.

Silva, L., Mondal, M., Correa, D., Benevenuto, F., & Weber, I. (2016). Analyzing the targets of hate in online social media. In *Tenth international AAAI conference on web and social media*.

Spanish Ministry of Interior (2019). Report on the evolution of hate crimes in Spain 2019 (informe sobre la evolución de los delitos de odio en España 2019). In *Estadísticas/servicios al ciudadano*. http://www.interior.gob.es/documents/642012/3479677/informe+evolucion+2019/631ce020-f9d0-4feb-901c-c3ee0a777896. (Last accessed 22 June 2022).

Sreelakshmi, K., Premjith, B., & Soman, K. (2020). Detection of hate speech text in Hindi-English code-mixed data. *Procedia Computer Science*, *171*, 737–744.

Sroka, E. C. (2020). Don't be afraid of nonparametric topic models (Part 2: Python). *Medium, Towards Data Science*.

Vijayaraghavan, P., Larochelle, H., & Roy, D. (2021). Interpretable multi-modal hate speech detection. arXiv:2103.01616.

Waseem, Z. (2016). Are you a racist or am I seeing things? Annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science* (pp. 138–142).

Waseem, Z., & Hovy, D. (2016). Hateful symbols or hateful people? Predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop* (pp. 88–93).

Zhang, Z., & Luo, L. (2019). Hate speech detection: A solved problem? The challenging case of long tail on twitter. *Semantic Web*, *10*(5), 925–945.

Zhang, Z., Robinson, D., & Tepper, J. (2018). Detecting hate speech on twitter using a convolution-gru based deep neural network. In *European semantic web conference* (pp. 745–760). Springer.

Zia, T., Akram, M. S., Nawaz, M. S., Shahzad, B., Abdullatif, A., Mustafa, R., et al. (2016). Identification of hatred speeches on Twitter. In *Proceedings of 52nd the IRES international conference* (pp. 27–32).