News-RO-Offense - A Romanian Offensive Language Dataset and Baseline Models Centered on News Article Comments

Andreea Cojocaru
University Politehnica of Bucharest
313 Splaiul Independetei, Bucharest, Romania
andreea.cojocaru99@stud.acs.upb.ro

Andrei Paraschiv
University Politehnica of Bucharest
313 Splaiul Independetei, Bucharest, Romania
andrei.paraschiv74@upb.ro

Mihai Dascalu
University Politehnica of Bucharest
313 Splaiul Independetei, Bucharest, Romania
mihai.dascalu@upb.ro

ABSTRACT
The use of offensive language can lead to uncomfortable situations, psychological harm, and in particular cases even to violence. Social networks and websites struggle to reduce the prevalence of these types of messages by using an automated detector. In this paper, we propose a novel Romanian language dataset for offensive message detection. We manually annotated 4,052 comments on a Romanian local news website into one of the following classes: non-offensive, targeted insults, racist, homophobic, and sexist. In addition, we establish a baseline of five automated classifiers, out of which the model based on RoBERT and two layers of CNN achieves the highest performance with an average F1-score of .74.

Author Keywords
Romanian Language; Hate-speech; Offensive Language; Natural Language Processing; Romanian Offensive Dataset

ACM Classification Keywords
I.2.7 Natural Language Processing: Text analysis.

General Terms
Dataset; Offensive Language; Natural Language Processing.

DOI: 10.37789/rochi.2022.1.1.12

INTRODUCTION
Anger, fatigue, frustration, and conflict situations can make most people slightly aggressive. Most of the time, this is expressed in words, sometimes leading to swearing or even insults. Such behavior can lead to abuse or, more than often, offend people, and these types of aggression can have a negative impact on their mental health. Whether we are talking about profane language, insults, or, worse, racist, homophobic, and sexist remarks, all of these can cause insecurities and anxiety. People can hide online behind the screen, thus providing the perfect opportunity for this kind of abusive behavior. Since a large proportion of the world's population uses social networks, message boards, or chat applications, exposure to offensive language or hate speech is highly probable. Additionally, there is a strong correlation between extremist messages on social networks and the spread of dangerous ideologies [12].

Filtering out offensive messages with human supervision can be a tedious and cumbersome task. There is a strong incentive to develop automatic hate speech detection, and there are many studies that propose various approaches, from classic machine learning to deep learning classification techniques [1, 3, 7]. Most of these algorithms require human-annotated training examples written in the specific language of the analyzed messages, in order to classify offensive and non-offensive texts. Unfortunately, not all spoken languages have the same richness in available datasets, since most of the research has focused on the English, German, Italian, or Spanish languages. In contrast, at present, there is only one dataset available for offensive speech detection in Romanian [5].

In this paper, we propose a novel Romanian language dataset for offensive and hate speech detection, News-RO-Offense, with 4052 records[1]. In addition, we present several approaches for the automatic detection of insults, racism, homophobia, and sexism using classical machine learning and deep learning models.

RELATED WORK
There are several ways to classify offensive speech based on the type of message (e.g, insult, cyberbullying, sexism, racism, abuse), the perceived target of the message (e.g., misogyny, homophobic, antisemitic), or if the target is a person or group. Zampieri et al. [14] proposed a three-level classification for offensive messages: the first level differentiates between offensive versus non-offensive messages, then the second level distinguishes between targeted and untargeted profanities, whereas the targeted texts are labeled based on the target categories on the third level, namely: individual, group, or other. In contrast, Waseem et al. [13] make a distinction between generalized, directed, explicit, and implicit offenses.

Even if the majority of studies focus on the English language, more and more do address other languages. Struß et al. [11] presented a classification based on the harshness of the offenses into PROFANITY, INSULT, and ABUSE classes; additionally, the authors considered the explicitness of the hate speech into implicit and explicit messages. For the Italian language, Sanguinetti et al. [9] annotated a Twitter-based corpus

[1] https://github.com/readerbench/news-ro-offense

with several hate speech aspects: aggressiveness, offensiveness, irony, and stereotype.

There is a growing interest in low-resource languages, i.e. languages with a small footprint on the Internet. For instance, Hande et al. [4] explore the Kannada language, a Dravidian language spoken in the Karnataka State of South India by bringing forth a dataset based on Youtube comments using a similar three-level annotation schema to the one proposed by Zampieri et al. [14].

According to statista.com, the most frequently used social media platforms in Romania are Facebook and Instagram [2]. Although these platforms are not spared from hate speech, the prevalence of such messages is very low. Additionally, these platforms are much harder to crawl, with most comments being visible only to logged-in users. As such, we focused our attention on comments placed by online readers on news articles. Many contentious topics in the newspapers rile people up, and these types of websites often practice little or no filtering in their comments section.

Data Extraction

For our News-RO-Offense dataset, we have chosen the "stiri-decluj.ro" website as the data source. Since it is a local news website, we assumed that users tend to be more involved in the matters at hand, whereas the discourse might deviate quickly. The website's comment validation mechanism considers a short list of words for censoring. Since this can be easily circumvented, people's comments can become very toxic. We employed a random selection to cover articles from various topics; 417 article titles and URLs were crawled, out of which 236 were selected for scraping the corresponding comments.

In this first step, a database of articles and comments with more than 4759 records was created, with corresponding date tags. We also stored a REPLY-TO field, marking the unique ID of the referenced comment within different threads.

Data Labeling

Specific offensive typologies stood out when manually browsing through the comments corpus and become classes used for our annotation process. We distinguish between four kinds of profane messages, namely: *Targeted insults*, *Racist* comments, *Homophobic*, and *Sexist* messages. Even if the classes seem to be independent and distinguishable, it was necessary to establish predefined rules for accurate labeling.

Looking at the insults, we are only interested in the targeted ones and disregard any unaddressed profanities. As such, the use of pejorative words that are not directed at a person or a group is considered non-offensive if the meaning of the sentence still remains the same when those words are removed; even if this might be perceived by some as offensive, the toxicity level of these messages is much lower. For instance, a comment like "Cum pula mea sa ma fac ministru atunci?" (eng., "How the fuck should I become a minister then?") would be labeled as non-offensive.

[2]https://www.statista.com/topics/7134/social-media-usage-in-romania

Using multi-label annotations does increase the annotation effort and reduced the chance for annotator consensus, and due to the limited resources at our disposal, we aimed for a single-label annotation scheme, such that only one class was chosen in comments where there are multiple kinds of offenses. Thus, we considered the most predominant and recurring class that appeared in the given comment, and, based on that, we decided on the annotation. To exemplify the presented situation, we select the following comment:

> *"Păi clar, așa te a făcut și doamna măta pe tine,pt că la noi e orice posibil.Ungure aș vrea să te cunosc ,să văd cât de coșuros și ce față de lăbar coșuros ai,după "ecran" te dai mare dar sper că autorități să se autosesizeze,tu ai aere de terorist,trădători de țară."*
>
> eng., *"Well, of course, that's how your mother made you, because everything is possible here. You hungarian guy, I would like to meet you, to see your pimpled head and to see your pimpled wanker face, behind the screen you are a badass, but I hope that the authorities will do something, you are a terrorist, you country traitor."*

We can observe that the quoted comment could be classified as a *targeted insult* or a *racist* message. In this case, we labeled it as racist since the emphasis is on the ethnicity of the target. There may be differences between the annotators perspectives, but these are special situations that were rarely encountered. Further, if comments contain words that represent ethnic insults, such as "țigan" ("gypsy"), "bozgor" ("hunky"), they are always classified as racist despite that the context is not denigrating.

The labeling process was performed using Microsoft Excel, with all comments and their replies ordered in a threaded manner to have a better understanding of the context. For instance, having four examples of homophobic comments in the order of appearance, the last comment, seen individually, not knowing the context, could be viewed as non-offensive; however, given the context of the thread (i.e., a gay pride parade), clearer labeling could be performed.

A second annotator was required to validate the labeling process on samples of 100 comments. This validation phase was performed in two steps. The first iteration was used to align the understanding of the annotation rules, followed by a second iteration, on a different 100 sample set, that provided the final agreement score.

Table 1 presents the number of errors made in each phase; these errors were established by reaching a consensus between the annotators on differently labeled entries. Figure 1 presents the confusion matrix between the annotators in the second step; Cohen's kappa of .859 denotes a high agreement. All remaining records were labeled by the first annotator.

The final class distribution of entries is presented in Table 2.

Result	Iteration1	Iteration2
Initial number of differences	18	17
Anno1 mistakes	2	8
Anno2 differences	16	9
Initial accuracy	82%	83%
Final accuracy	84%	91%

Table 1. Annotation differences.

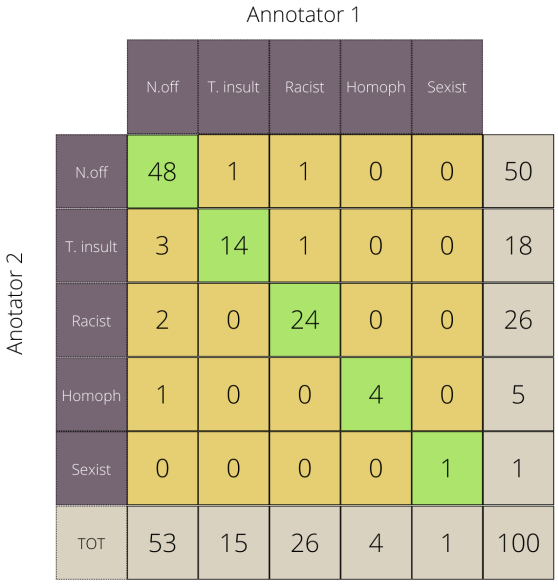

Figure 1. Confusion matrix - human annotators.

Dataset Statistics

In this subsection, we present a brief statistical overview of the proposed dataset. The messages were tokenized using two methods, the first based on white spaces and the second using the BERT tokenizer. Additionally, we identified class-specific words by counting their occurrence per class. If the occurrences of a word in a given class are greater than all other classes combined, then we considered it as being specific to that class. Table 3 introduces the counts of specific words per class. As expected, the highest count was in the non-offensive class since it has the largest variability. Also, a noticeable difference for the *sexist* class can be observed which is significantly lower than all other classes. The total number of distinct words was 15,705, with 3,757 present in all or multiple classes equally. Figure 2 plots the word cloud for the *racist* class.

CLASSIFICATION MODELS

Classical Machine Learning

In all classical machine learning methods, we use TF-IDF to extract the features from the pre-processed text, as described in the following subsection.

Class	Label	# examples	Percentage
Non-offensive	0	2682	66.19%
Targeted insult	1	777	19.18%
Racist	2	252	6.22%
Homophobic	3	186	4.59%
Sexist	4	155	3.82%
TOTAL		**4052**	

Table 2. Dataset class distribution.

Class	avg #words	avg #BERT tokens	#specific words
Non-offensive	46.70	31.8	8200
Targeted insult	40.28	29.73	2055
Racist	46.31	35.5	752
Homophobic	45.26	30.8	581
Sexist	36.25	22.2	360

Table 3. Word metrics per class.

Data Preprocessing

The first step in the preprocessing pipeline consists of case-lowering all words and then removing any kind of Romanian-specific stopwords [3]. Since most replied comments start with the following phrase: "@name-of-parent:", we deemed it irrelevant and removed it. Additionally, we removed all URLs in the texts.

After cleaning the dataset in the first step, we applied lemmatization on all records to group the inflectional forms of the same word under a single unit, thus capturing several forms of the same word under the same token. For this, we used the "spacy" library, namely the "ro_core_news_sm" model [4].

However, the lemmatizer does not recognize words written without them, as the Romanian alphabet contains diacritics. Most people in Romania, especially in informal settings, do not write with diacritics when using the keyboard. A such, we used the ReaderBench framework [5] to restore diacritics before the lemmatization.

Thus, our full pre-processing pipeline for the classical machine learning algorithms trans:

- Initial comment: "Ce este clar domnule Sfarlea ,sunteti un om de nimic care se bazeaza pe furat."
- Preprocessing without adding diacritics: "clar domn sfarlea om bazeaza fura"
- Preprocessing with diacritics: "clar domn sfârlea om baza fura"

[3] https://www.ranks.nl/stopwords/romanian
[4] https://spacy.io/models/ro
[5] https://github.com/readerbench/ReaderBench

Data Balancing
Given the differences in class distributions from Table 2 with a majority class having more than 60% of all examples, we applied a simple data balancing strategy consisting of two operations, namely undersampling from the majority class and oversampling from the minority classes.

Hyperparameters Tuning
We used the grid-search method with five folds per combination of parameters to determine the SVM hyper-parameters, namely C (i.e., the regularization parameter) and the kernel type (i.e., linear, polynomial, RBF, sigmoid).

Deep Learning Models

For our deep learning experiments, we used Transformer-based models, namely BERT-based [2] architectures. All neural network experiments were performed using Tensorflow and were run on the Google Colab platform.

Data Balancing
As mentioned before, the dataset is unbalanced and we considered class weights for the neural networks [10] through which all classes become equally important by lowering the probability of having a higher prediction over the majority class.

Data preprocessing
All messages were processed using the WordPiece tokenizer that splits words into subword tokens. Also, sequences longer than 512 tokes were truncated, whereas shorter ones were padded with null tokens up to this maximum length.

Hyperparameter Tuning and Training
The maximum length of the input text and the batch size were considered hyperparameters for the neural models. As the batch size increases, the maximum length has to decrease and vice versa. In order to achieve a balance, we performed a grid search to find the best combination using for the batch size the values: 8, 16, and 32, and for the maximum length: 128, 256. In our training process, we used a validation set and early stopping as a regularization method to avoid overfitting. The monitored metric for the early stopping was the loss, while the patience was set to five epochs.

Neural Architectures
For our experiments, we considered two flavours of BERT, namely RoBERT-base [6] trained specifically for the Romanian language and a multi-language version and bert-base-multilingual-cased (multiBERT) [2] trained on 104 different language, including Romanian. Regardless of the considered model, the architectures remain the same; the only changes involve the tokenizer, the BERT encoding layer and its configuration.

BERT + MLP A Multilayer Perceptron (MLP) is a simple addition on top of the BERT encoder with good results on multiple NLP problems. In our case, the BERT + MLP architecture has the following layers: a BERT-layer, average pooling, dropout, followed by a fully connected layer and another layer with five outputs, each with softmax activation in order to perform classification. We optimized our parameters

Figure 2. Racist class wordcloud.

trough grid-search, leading to our final model which used an 128 large MLP layer, a dropout rate of 20% and a maximum token sequence length of 128. In the training process we used an Adam optimizer with a learning rate of $5e^{-5}$ with a 0.01 decay rate and the gradient norm clipped at 1.0.

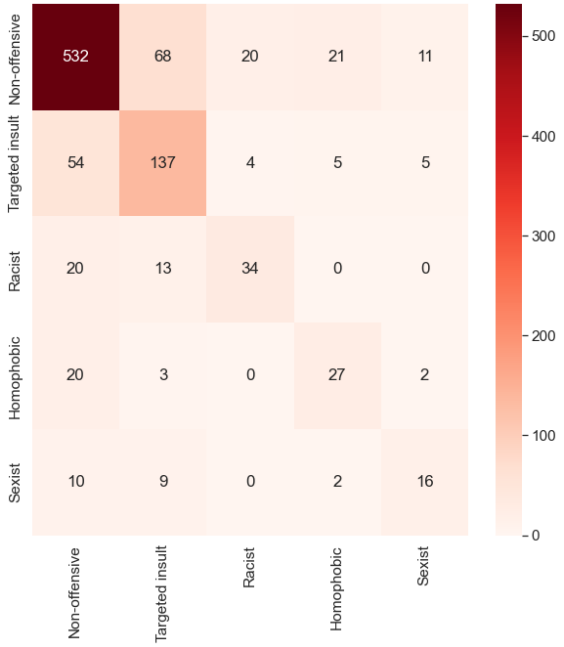

Figure 3. RoBERT + CNN confusion matrix.

BERT + CNN

Adding CNN layers for feature extraction is a common technique in NLP [15]. Convolutional layers can be used in conjunction with a BERT encoder to improve the performance of the model, as suggested by Safaya et al. [8]. Our architec-

ture has two convolution layers, a maximum Pooling layer, a fully connected layer, followed by a dropout layer and the classification layer with 5 outputs, one for each class. The first convolution layer has 32 channels, while the second one has 64 channels, both of them having kernel sizes equal to 2. These layers are followed by the Maximum Pooling layer which reduces the dimensions and highlights the contrasts. The network is flattened and connected to the 512 sized dense layer and the final classification layer. Similar to the BERT + MLP model, we used a maximum token sequence length of 128.

RESULTS

The results of our experiments with the previously detailed models are presented in Table 4.

On the first column of Table 4 we can observe the models and the best configuration in terms of hyperparameters. For the SVM model, we performed an additional experiment with a binary classification in which we merged the offensive classes into a single one. The SVM model has the best precision of all models for the offensive classes in a multi-class setup, but a poor one for the negative class. This is reflected also in the Recall since most of the homophobic and sexist samples were classified as non-offensive. For these classes, only 6 of the 39 samples in the test set were correctly detected. In a binary setting, the SVM model is more balanced.

Analyzing the BERT-based neural models, both MLP and CNN networks have comparable results, while greatly surpassing the SVM model. The main difference is given by the BERT encoding layer used as the RoBERT-based models obtain significantly better results than the multiBERT models. Also, it is noticeable that the CNN architectures for MultiBERT perform slightly worse than MLP. As expected, the CNN architecture performed better for the RoBERT model, being the best configuration in our experiments. In contrast to the SVM, RoBERT + CNN no longer tries to add most samples in the non-offensive class. Also, we notice a slightly lower precision than the SVM, but substantially higher than all other models. The highest improvement of the BERT-based models is its recall, with RoBERT + CNN having the best value for sexist messages from all the classes. We consider this important since this class has the fewest support examples, thus being harder to detect.

Also, the RoBERT + CNN model has the best results in terms of F1-score per class, delivering the best balance between Precision and Recall, and significantly surpassing all other models for the sexist class. The model also has the highest overall accuracy and weighted F1-score.

The confusion matrix for this model is depicted in Figure 3 where we notice a balanced result. Nonetheless, most errors were still in the classification of offensive messages into the non-offensive class.

DISCUSSION

Our aim given the best classification model (i.e., RoBERT + CNN) is to derive better annotation rules and seek ways to improve the detection process.

Error Analysis

The misclassifications from the best model are grouped into five types of errors (i.e., Annotation errors, Misinterpretations, Typos, Generalization, and Subtlety) to better understand the cause of the erroneous classifications (see Table 4 for examples).

First, grouped under *Annotation errors*, there are possible human mistakes that, on a second glance, are correctly classified by the model. Although the model makes a correct prediction, these cases are treated as errors due to annotation mistakes.

The second category of errors refers to interpretation difficulties. We want the automated detector to properly classify whether a comment is offensive and in what manner, but even we, as humans, cannot always decide on the correct class. Given that such problems were encountered by the annotators during data labeling, as well as contradictions between them in the agreement phase, it is normal that the model produces such errors.

Further, we analyze the impact of using typos or spelling mistakes to obfuscate words on purpose. There are a lot of misspelled alternatives to writing a word, and these errors create problems for the tokenizer as the decomposition into subword tokens becomes flawed. For example, if we take the two forms of the misspelled pejorative word used in the first two examples, we can see that the BERT tokenizer produces totally different representations, leading to, potentially, different projections into the embedding space. This can be especially impactful if these typos appear in insulting words that make the entire comment offensive. Nevertheless, this also depends on the context. If the model manages to infer from the context that a pejorative word would follow, then these problems are mitigated. However, there are circumstances where this does not happen, such as in the second example from this category.

Generalization errors refer to the usage of words that may occur in a derogatory context, but in the current context, they do not indicate such behavior. In the first example from this category, the discussion is about specific nations, but not at a denigrating level. In contrast, the comment from the second example is against homophobic insults. Both were incorrectly predicted.

The last category refers to subtle insults; this includes comments that do not use any profane or derogatory words but using common sense one can derive its offensive nature.

We consider that the found errors are expected. Our best model manages to correctly classify clear and obvious examples, and it also deals well with most complex samples.

Limitations

There are several limitations to our dataset. One of the most important ones is the local nature of the data source. Even if the website caters to a narrow audience around one of Romania's largest cities, Cluj, it may still capture a partially representative section of online discourse. Moreover, there are many regionalisms, words, and expressions specific to that area. Also, if we refer to the racist category, the comments are mostly about common racist talking points from Cluj county.

Model	Class	Precision	Recall	F1-score	Accuracy	Weighted F1-score
SVM exp.A C = 2 kernel = 'rbf'	Non-offensive Targeted insult Racist Homophobic Sexist	0.48 **0.62** **0.93** **0.55** **0.50**	0.72 0.63 0.37 0.15 0.15	0.58 **0.63** 0.53 0.24 0.24	0.56	0.54
SVM exp.B C = 2 kernel = 'rbf'	Non-offensive Offensive	0.69 0.67	0.73 0.63	0.71 0.65	0.68	0.68
RoBERT + MLP batch_size = 8 max_len = 128	Non-offensive Targeted insult Racist Homophobic Sexist	**0.84** 0.57 0.54 0.38 0.27	0.78 0.59 **0.64** **0.55** 0.32	0.81 0.58 **0.59** 0.45 0.29	0.71	0.71
multiBERT + MLP batch_size = 32 max_len = 128	Non-offensive Targeted insult Racist Homophobic Sexist	0.79 0.53 0.54 0.38 0.36	**0.83** 0.45 0.40 **0.55** 0.20	0.81 0.48 0.47 0.45 0.26	0.70	0.69
RoBERT + CNN batch_size = 16 max_len = 128	Non-offensive Targeted insult Racist Homophobic Sexist	**0.84** 0.60 0.59 0.49 0.47	0.82 **0.67** 0.51 0.52 **0.43**	**0.83** **0.63** 0.54 **0.50** **0.45**	**0.74**	**0.74**
multiBERT + CNN batch_size = 32 max_len = 128	Non-offensive Targeted insult Racist Homophobic Sexist	0.78 0.41 0.42 0.40 0.20	0.79 0.41 0.39 0.45 0.12	0.78 0.41 0.40 0.42 0.13	0.65	0.65

Table 4. Experimental results.

Thus, racist insults related to other cities or countries may not be detected.

Another limitation is the limited dataset size, with reduced support for the homophobic and sexist categories. Also, since we considered a single-label annotation, messages pertaining to more than one class are placed in just one. This problem can be solved by creating multiple mixed classes or annotating the comments using multiple labels; however, this would introduce a higher complexity in the annotation process.

Lastly, since most of the annotation process was performed by only one annotator, there is the risk of infusing personal biases into the dataset. We tried to verify this by employing an additional annotator for validation, but due to the limited size of the cross-validated sample, some biases might slip the detection.

CONCLUSIONS

The detection of offensive language should be a mandatory task, especially for social networks. The identification of different types of abuse allows us to realize the quality of the messages that cross a social network or a website. With real motivation and clear examples, we may prevent abusive behavior against any individual or group, regardless of their status, gender, race, or sexual orientation.

In this paper, we propose a novel Romanian language dataset for offensive speech detection based on comments related to news articles into five classes: non-offensive, targeted insults, racist, homophobic, and sexist. The annotation was performed by a native Romanian-speaking person, with validation on a random sample from a second native speaker. In addition, we establish a strong baseline with various NLP models. Even if classical machine learning models fail to perform satisfactorilyy in a multi-class setup, we show that BERT-based models have the capacity to generalize well from relatively few ex-

Category	Examples	Actual class	Predicted class
Annotation errors	"da-i muie la ma-ta ! Din nou." *Your mother should blow you! Again.*	Non-offensive	Targeted insult
	"dute tu cioara acasa la mamica ta .urita mai esti" *Crow[6], go back home to your mother. You're so ugly*	Targeted insult	Racist
Misinterpretation	"Baaa ce curve bune de supt pula isi gasesc ministrii astia!!!" *Maaan, what cocksucking whores do these ministers find for them*	Sexist	Targeted insult
	"cred ca imi si bagam p....a in iel ca i si zburam dinti si si la baba aia proasta i rupeam gura" *I think I would have stick my c..k in that ring and shattered (his/her) teeth out of (his/her) mouth and I would brake the jaw of that stupid crone*	Targeted insult	Sexist
Use of typos	"la biserica le da m u i e popa..." *at the church the priest is getting b l o w n...*	Targeted insult	Non-offensive
	"Ziganii au IQ 80, is redusi mintal din nastere" *Jipsies have 80 IQ, they are retarded from birth on*	Racist	Non-offensive
Generalization	"Mai bine cu rusii si cu sarbi decut cu voi :))) *Better with russians than with serbs like you*"	Non-offensive	Racist
	"Sunt GAY si resping cu fermitate toate afirmatiile demai jos! VAI! Unde am ajuns! SINTEM TARA EUROPEANA SAU NU? UNDE VA ESTE CIVILIZATIA SI VALORILE?" *I am GAY and I firmly reject these allegations! Oh my! What we have become! ARE WE EUROPEANS OR NOT? WHERE ARE THE CIVILIZATION AND VALUES*	Non-offensive	Homophobic
Subtle insults	"Esti prea de la taranoaia sa intelegi ceva din asfalt." *You too much of a peasent to know what asphalt means*	Targeted insult	Non-offensive

Table 5. The Classification of detection errors and examples.

amples and can be used in a reliable fashion. As expected, BERT models pre-trained on the considered language perform significantly better than multilingual pre-trained ones.

In regard to future work, the dataset can be extended with comments from additional websites to the increase variability of examples. Additionally, the use of multiple annotators could further improve the robustness of the data.

REFERENCES

1. Ona De Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate speech dataset from a white supremacy forum. *arXiv preprint arXiv:1809.04444* (2018).
2. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
3. Shashank Gupta and Zeerak Waseem. 2017. A comparative study of embeddings methods for hate speech detection from tweets. (2017).
4. Adeep Hande, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2020. KanCMD: Kannada CodeMixed dataset for sentiment analysis and offensive language detection. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*. 54–63.
5. Mihai Manolescu and Çağrı Çöltekin. 2021. ROFF-A Romanian Twitter Dataset for Offensive Language. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*. 895–900.
6. Mihai Masala, Stefan Ruseti, and Mihai Dascalu. 2020. Robert–a romanian bert model. In *Proceedings of the 28th International Conference on Computational Linguistics*. 6626–6637.
7. Mr Mohiyaddeen and Sifatullah Siddiqi. 2021. Automatic hate speech detection: A literature review. *Available at SSRN 3887383* (2021).
8. Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. 2054–2059.
9. Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An italian

twitter corpus of hate speech against immigrants. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*.

10. Kamaldeep Singh. 2020. How to Improve Class Imbalance using Class Weights in Machine Learning. `https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/`. (2020). Last accesed 14/06/2022.
11. Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, Manfred Klenner, and others. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. (2019).
12. Ashish Sureka and Swati Agarwal. 2014. Learning to classify hate and extremism promoting tweets. In *2014 IEEE joint intelligence and security informatics conference*. IEEE, 320–320.
13. Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding abuse: A typology of abusive language detection subtasks. *arXiv preprint arXiv:1705.09899* (2017).
14. Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666* (2019).
15. Ye Zhang and Byron Wallace. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820* (2015).