# **HULAT** at SemEval-2023 Task 10: Data Augmentation for Pre-trained Transformers Applied to the Detection of Sexism in Social Media

# Isabel Segura-Bedmar

Computer Science Department, Universidad Carlos III de Madrid, Leganés, Spain isegura@inf.uc3m.es

## **Abstract**

This paper describes our participation in SemEval-2023 Task 10, whose goal is the detection of sexism in social media. We explore some of the most popular transformer models such as BERT, DistilBERT, RoBERTa, and XL-Net. We also study different data augmentation techniques to increase the training dataset. During the development phase, our best results were obtained by using RoBERTa and data augmentation for tasks B and C. However, the use of synthetic data does not improve the results for task A. We participated in the three subtasks. Our approach still has much room for improvement, especially in the two fine-grained classifications. All our code is available in the repository https://github.com/isegura/hulat\_ edos.

#### 1 Introduction

Sexism can be defined as behaviors or beliefs that support gender inequality, and result in discrimination, generally against women. Contrary to what one might believe, sexism is still very present also in the most advanced and technologically advanced societies (Ridgeway, 2011). Proof of this is that many gender stereotypes are still present in our belief system today (for example, men should not wear dresses). Unfortunately, social networks are used to spread hateful and sexist messages against women (Rodríguez-Sánchez et al., 2020).

During the last few years, various research efforts (Rodríguez-Sánchez et al., 2022; Fersini et al., 2022) have been devoted to the development of automatic tools for the detection of sexist content. While these automated tools have addressed the classification of sexist content, this is a highlevel classification, without providing additional information that allows us to understand why the content is sexist. The goal of SemEval-2023 Task 10, Explainable Detection of Online Sexism (EDOS)(Kirk et al., 2023), is to promote the development of fine-grained classification models for

detecting sexism in posts written in English, which were collected from Gab and Reddit. The organizers of the task proposed three subtasks: A) Binary Sexism Detection, B) Category of Sexism, a four-class classification task, and C) Fine-grained Vector of Sexism, an 11-class classification. A detailed description of these classifications can be found at Kirk et al. (2023).

In our approach, we explored some of the most popular pre-trained transformer models such as BERT (Devlin et al., 2019), DistilBERT (Sanh et al., 2019), RoBERTa (Zhuang et al., 2021), and XLNet (Yang et al., 2019). Moreover, we used different data augmentation techniques such as EDA (Wei and Zou, 2019) and NLPAug library to create synthetic data. Then, synthetic data and training data were used to fine-tune the models. Based on our experiments during the development phase, we decided to use the RoBERTa transformer model to estimate our predictions for the test dataset during the test phase.

We participated in the three subtasks. In task A, our system obtained a Macro F1-score of 0.8298, ranking 43th, with a total of 84 teams in the final ranking. The top system achieved a Macro F1score of 0.8746, while the lowest Macro F1-score was 0.5029. About half of the systems achieved a Macro F1-score below 0.83. In task B, our system ranked in the 45th position out of the 69 participating systems. Our Macro F1-score was 0.5877, while the lowest and highest Macro F1-scores were 0.229 and 0.7326, respectively. In task C, our team ranked in the 27th position out of the 63 participating systems. The lowest and highest Macro F1-scores were 0.06 and 0.56, respectively. About half of the systems achieved a Macro F1-score below 0.42, while our system had a Macro F1-score of 0.44.

Our systems, which ranked roughly in the middle of the three rankings, show modest results on the

<sup>1</sup>https://github.com/makcedward/nlpaug

three subtasks. Our approach still has much room for improvement, especially in the two fine-grained classifications. The results showed that the use of synthetic data does not appear to provide a significant improvement in the performance of the transformers. All our code is available in the repository https://github.com/isegura/hulat\_edos.

#### 2 Dataset Overview

The goal of this task is to detect sexist content. The task is composed of three subtasks: A, B and C. Task A is a binary classification task to distinguish between sexism and non-sexism texts. Task B and C aim to make a more fine-grained classification with four and eleven classes, respectively.

The full dataset consists of 20,000 posts written in English. Half of the posts were taken from Reddit and the other half from Gab. Gab is a social network known for its far-right users. The dataset was divided in three splits with a ratio of 70:10:20. That is, 14,000 posts were used for training, 2,000 for development, and 4,000 for the final evaluation.

We have studied the class distribution in each task. In task A, a binary classification, the two classes are not balanced, where the not-sexist class is the majority class (see Fig. 1a). As expected, the same distribution was observed in the three splits (which have been provided by the organizers). We also plot the distribution of categories for task B (see Fig. 1b. To obtain the distribution of these categories, we removed those records that were annotated as "not sexist". The majority category is "2. derogation". The second class with a larger number of instances is "3. animosity". The other two classes are the minority classes, "4. prejudiced discussions" and "1. threats, which have a similar number of instances. The same distribution is observed in the three dataset splits.

Regarding the distribution of the vectors in task C (see Fig. 1c), the vector subcategory "2.1 descriptive attacks" is the majority class, while "3.4 condescending explanations or unwelcome advice" is the minority class. The vectors follow a distribution similar to that of their corresponding categories. For example, the vectors with the largest number of instances are usually the vectors of the category "2. derogation", followed by the vectors corresponding to the category "3. animosity".

We studied the length of the texts in tokens (see Fig. 2) to set the max\_length argument in the transformers. This argument controls the length of the

padding and truncation. To calculate the number of tokens in a text, we split it by white space. The mean number of tokens is around 23.3 with a standard deviation of 11.7, and the maximum length is 58 tokens.

We also want to know if there are differences in the length of the texts between the two main classes: sexist and non-sexist (Fig. 3). The distribution of length for sexist texts is slightly more skewed towards longer texts than the non-sexist distribution.

Figure 4 shows the length distribution of the texts for each category in task B. We can see that the texts classified as "4. prejudiced discussions" appear to be longer than the other texts (mean number of tokens is around 27.8 with a standard deviation of 10.9). The category "1. threats, plans to harm and incitement" has the shortest texts, with an average length around 22.9 tokens and a standard deviation of 11.7. The other two categories, "2. derogation" and "3. animosity", show very similar distribution with an average length of around 24 tokens for their texts.

We also study the length distribution of texts for each vector. As there are eleven vectors, it is very difficult to compare their distributions. For this reason, we created a density graph for the vectors within each category (see Appendix, Fig. 5). All vectors have a very similar distribution of text length. Texts classified as "4.1 supporting mistreatment of individual women" or "4.2 supporting systemic discrimination against women as a group" tend to have the largest average length between 27 and 30 tokens. The vector "2.1 descriptive attacks" has an average length of 26 tokens. The vector "1.2 incitement and encouragement of harm" has the smallest average length (around 22 tokens). The other vectors have an average length between 23 and 25 tokens. Therefore, there do not seem to be significant differences between the length of the texts of each vector.

# 3 System Overview

#### 3.1 Transformers

We explore some of the most successful transformer models such as BERT (Devlin et al., 2019), DistilBERT (Sanh et al., 2019), RoBERTa (Zhuang et al., 2021), and XLNet (Yang et al., 2019). These models were chosen because they are widely used for text classification (Arabadzhieva-Kalcheva and Kovachev, 2022; Minaee et al., 2021).

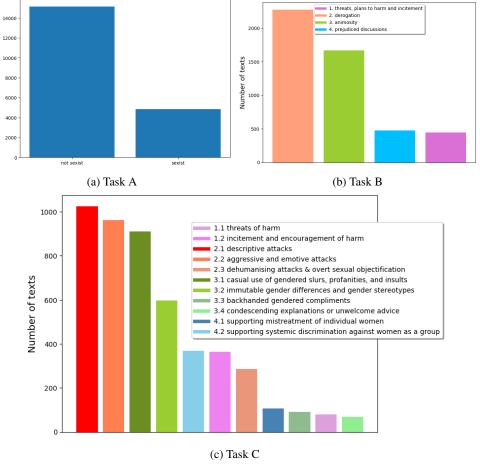


Figure 1: Class distribution for each task.

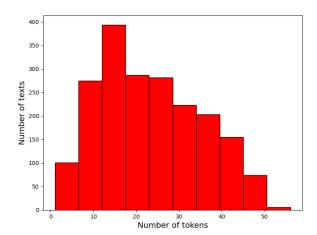


Figure 2: Distribution of text length (number of tokens).

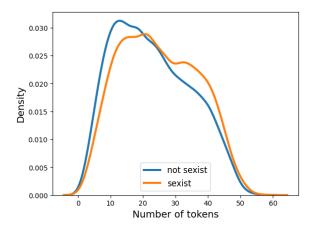


Figure 3: Density graph of the length of texts for the classes sexist and not sexist (task A).

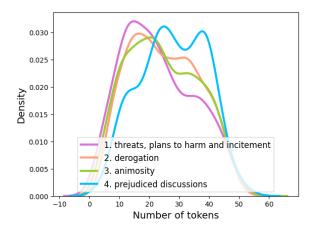


Figure 4: Density graph of the length of texts for each category (task B).

BERT (Devlin et al., 2019) is a popular transformer model due to its excellent results in many NLP tasks. BERT is an encoder trained using two strategies: masked language modeling (MLM) and next sentence prediction (NSP). DistilBERT (Sanh et al., 2019) is a smaller version of BERT, which can achieve similar results to BERT but with less training time.

RoBERTa (Zhuang et al., 2021) is based on BERT. RoBERTa was pre-trained using additional data. Unlike BERT, RoBERTa does not use the next sentence prediction (NSP) strategy. Regarding the MLM strategy, some tokens are dynamically masked during pre-training. Another difference with BERT is that RoBERTa uses a byte-level BPE tokenizer, which has a larger vocabulary than BERT (50k vs 30k). Therefore, RoBERTa has a larger vocabulary that can provide better results, but with an increase in complexity.

XLNet (Yang et al., 2019) is an autoregressive model. That is, it was pre-trained to predict the next token for a given input sequence of tokens. XLNet does not use any masked strategy. Instead of this, it uses permutation language modelling that can capture context by training an autoregressive model on all possible permutations of words in a sentence. In this way, bidirectional contextualized word representations are obtained. Like BERT, this model was trained with Wikipedia and BooksCorpus, but also with Giga5, ClueWeb 2012-B, and Common Crawl.

#### 3.2 Data augmentation

Data augmentation (DA) aims to increase the training data size by applying different transformations to the original dataset. For example, in computer

vision, some modifications can be performed by cropping, flipping, changing colors, and rotating pictures. While those transformations are easier to implement in computer vision, they are challenging in NLP, because they can alter the grammatical structure of a text. These transformations for NLP tasks include swapping tokens (but also characters or sentences), deletion or random insertion of tokens (but also characters or sentences), and back translation of texts between different languages.

Another advantage is that these techniques help to enhance the diversity of the examples in the dataset. Moreover, they also help to avoid overfitting. Unfortunately, data augmentation does not always improve the results in NLP tasks (Li et al., 2022).

In this task, we used different data augmentation techniques such as EDA (Wei and Zou, 2019) and NLPAug library to create synthetic data.<sup>2</sup>

EDA uses four simple operations: synonym replacement, random insertion, random swap, and random deletion. The first operation randomly chooses n words in a sentence (which are not stopwords). Then, these words are replaced with synonyms from WordNet, a very large lexicon for English.<sup>3</sup> Random insertion chooses a random word (which is not a stopword). Then, it finds a synonym that is inserted in a random position in the sentence. The original word is not removed from the sentence. The third operation, random swap, randomly chooses two words in the sentence and swaps their positions. The fourth operation, random deletion, randomly removes a word from a sentence. These operations can be repeated several times. We used the textaugment library, which implements these operations.<sup>4</sup>

NLPAug also provides an efficient implementation of DA techniques. In particular, NLPAug offers three types of augmentation: Characterlevel augmentation, Word-level augmentation, and Sentence-level augmentation. In each of these levels, NLPAug provides all the operations described above, that is, synonym replacement, random deletion, random insertion, and swapping. Regarding synonym replacement, the most effective way is using word embeddings to select the synonyms. This technique allows us to obtain a sentence with the same meaning but with different words. NLPAug

<sup>&</sup>lt;sup>2</sup>https://github.com/makcedward/nlpaug

<sup>3</sup>https://wordnet.princeton.edu/

<sup>4</sup>https://github.com/dsfsi/textaugment

uses non-contextual embeddings (such as Glove, word2vec, etc) or contextual embeddings (such as BERT, RoBERTa, etc).

We evaluated different combinations of data augmentation techniques for the task. Our experiments showed us that the best techniques are: i) synonym replacements provided by EDA, which is based on WordNet, and ii) synonym replacements provided by NLPAug based on BERT. Based on this, we decided to generate for each text, two new instances: one using EDA and the other using NLPAug. We decided not to add more synthetic instances in order not to increase the training time. In task A, we only create new instances for the minority (sexist) class because there is a large unbalance between the two classes (10,602 non-sexist instances versus 3,398 sexist instances). Thus, the augmented dataset for training task A contains 10,194 instances of the "sexist" class. Tables 1 and 2 show a comparison between the number of instances in the training dataset for tasks B and C, respectively, using and not using data augmentation.

Category	without DA	DA
1. threats, plans to harm	310	930
2. derogation	1,590	4,770
3. animosity	1,165	3,495
4. prejudiced discussions	333	999

Table 1: Number of instances for each category in task B, using and not using data augmentation.

Vector	without DA	DA
1.1	56	168
1.2	254	762
2.1	717	2151
2.2	673	2,019
2.3	200	600
3.1	637	1,911
3.2	417	1,251
3.3	64	192
3.4	47	141
4.1	75	225
4.2	258	774

Table 2: Number of instances for each vector in task C, using and not using data augmentation.

Based on our results in the development phase, for task B and C, we decided to use RoBERTa combined with data augmentation techniques to generate the final predictions on the test dataset provided by the organizers. However, for task A, we only used RoBERTa, because the data augmentation techniques did not appear to improve the results for the binary classification.

#### 4 Results

HULAT participated in the three subtasks. Below we present our results in each task.

#### 4.1 Task A

As was previously said, we fine-tuned a RoBERTa model using the full training dataset, without using synthetic data. Our system provided a Macro F1-score of 0.8298, obtaining the 43rd position of a total of 84 participating systems. The highest Macro F1-score was 0.8746, while the lowest was 0.5029. About half of the systems achieved a Macro F1-score below 0.83.

Model	Aug.	P	R	F1-score
RoBERTa	No	.852	.817	.832
ROBERTA	Yes	.830	.819	.824
DEDT (umassed)	No	.841	.815	.827
BERT (uncased)	Yes	.824	.813	.818
XLNet	No	.829	.817	.823
ALINEI	Yes	.836	.818	.826
DistilDEDT (unassed)	No	.815	.813	.814
DistilBERT (uncased)	Yes	.788	.824	.803
DEDT (1)	No	.811	.814	.813
BERT (cased)	Yes	.80	.828	.812
D:-4:IDEDT (1)	No	.815	.802	.808
DistilBERT (cased)	Yes	.798	.826	.810

Table 3: Macro-averaged scores for task A on the final test dataset. P stands for Precision, R for Recall. RoBERTa (without DA) was the model used to create our submission on the test dataset.

Table 3 shows our final results on the test dataset for task A. It also reports the results of all combinations that we studied.

We evaluated both the uncased and the cased versions of BERT. BERT uncased shows better results than the cased version. The use of data augmentation does not improve the results of the BERT model in none of their versions, cased or uncased. DistilBERT obtains slightly lower results than BERT, though its training time is much better. Data augmentation helps to increase recall, but with worse precision. The improvement in F1 is not significant. There are hardly any differences between the results of the cased model and those obtained under the uncased version of DistilBERT.

XLNet has very similar results to those obtained by the uncased version of BERT. The data augmentation techniques do not appear to improve the results. RoBERTa defeats all previous approaches. In particular, RoBERTa achieves better precision than DistilBERT and BERT. Regarding the results obtained by data augmentation, the use of synthetic data negatively affects the precision of RoBERTa.

In sum, all the models show very close results, and data augmentation does not improve the results. RoBERTa slightly outperforms the other models.

#### 4.2 Task B

In this task, we fine-tuned the RoBERTa model using the full training and the synthetic data created with the data augmentation techniques described in section 3. Our system ranked in the 45th position out of the 69 participating systems. Our Macro F1-score was 0.5877, while the lowest and highest Macro F1-scores were 0.229 and 0.7326, respectively.

Model	Aug.	P	R	F1-score
RoBERTa	No	.601	.598	.595
ROBERTA	Yes	.554	.620	.572
BERT (uncased)	No	.613	.598	.599
BERT (uncased)	Yes	.599	.585	.589
DEDT (1)	No	.599	.570	.581
BERT (cased)	Yes	.614	.534	.563
VI NI 4	No	.601	.598	.595
XLNet	Yes	.546	.582	.553
DistilBERT (cased)	No	.550	.508	.522
	Yes	.540	.558	.540
DistilDEDT (unassed)	No	.542	.496	.508
DistilBERT (uncased)	Yes	.512	.544	.524

Table 4: Macro-averaged scores for task B on the final test dataset. P stands for Precision, R for Recall. RoBERTa with DA was the model used to create our submission on the test dataset.

Table 4 shows the results on the test dataset for task B. We evaluated all combinations that we studied. Although our experiments during the development phase showed that using augmented data increased the performance of all transformers in task B, the final results on the test dataset show the opposite.

In task B, we again evaluated both the uncased and the cased versions of BERT. Although both versions obtain close results, the uncased version shows slightly better precision and recall than the cased one. For the cased version of BERT, data augmentation improves the precision but significantly lowers the recall. It also has a negative effect on the performance of the BERT uncased model.

Contrary to BERT, the cased version of Distil-BERT is slightly superior to its uncased version. However, the results are so close that these differences are not statistically significant in the models. The use of data augmentation shows an improvement in Macro F1-score (in both versions of Distil-BERT), but with a slight decrease in the precision. Therefore, unlike BERT, DistilBERT gets some im-

provements thanks to the use of data augmentation.

XLNet outperforms DistilBERT, showing similar results to BERT. As with BERT, data augmentation does not appear to help XLNet in classifying the four categories for sexism.

RoBERTa achieves a Macro F1-score of 0.595. Data augmentation increases the recall, but with a significant decrease of the precision. However, RoBERTa with data augmentation obtained the best results on the development set during the development phase. For this reason, we decided to use this combination for our final submission on the test phase.

Table 5 shows the results of RoBERTa with data augmentation for each category. Although the category "1. threats, plans to harm and incitement" has the lowest number of instances in the dataset (see Fig. 1b), it shows the top F1 (0.624). The posts in this category are shorter than the posts in the rest of the categories (see Fig. 4). Moreover, an analysis of these texts show that they usually use a very violent vocabulary. Indeed, some of their most common words are: "bitch", "kill", "rape", "fuck", "punch", "beat", "kick", "hang", "death", or "slap". The category with the lowest F1 is "4. prejudiced discussions". The lower score may be due to the fact of this category has very few instances compared to the second (derogation) and third (animosity) categories (see Fig 1b). Moreover, its texts tend to be longer than the texts of the first category (threats) (see Fig. 4). The scarcity of examples in this category together with the fact that they do not use aggressive vocabulary as it was in the first category, may make very challenging to classify them.

Category	P	R	F1-score	Instances
1	.546	.730	.624	89
2	.698	.464	.558	454
3	.527	.66	.586	333
4	.446	.627	.522	94

Table 5: Results provided by RoBERTa and data augmentation on the test dataset (task B) for categories: "1. threats, plans to harm and incitement", "2. derogation", "3. animosity", and "4. prejudiced discussions". P stands for Precision, R for Recall.

#### 4.3 Task C

In task C, we used the same approach as for task B, that is, RoBERT and data augmentation techniques. Our system obtained a Macro F1-score of 0.4458, which ranked in the 27th position out

of the 63 participating systems. The lowest and highest Macro F1-scores were 0.06 and 0.56, respectively. About half of the systems achieved a Macro F1-score below 0.42.

Table 6 shows the results on the test dataset for task C. We evaluated all combinations that we studied during the development phase. Unlike tasks A and B, data augmentation techniques have a positive effect on the results for task C for all transformers. The cased version of BERT slightly outperforms the uncased version. DistilBERT provides lower results than BERT. Both versions of DistilBERT, cased and uncased, show very close results. XLNet outperforms BERT. RoBERTa obtains the best scores, outperforming the other models. In addition, the use of augmented data helps to increase the results. In sum, RoBERTa trained with training and synthetic data is the best approach for task C.

Model	Aug.	P	R	F1
RoBERTa	No	.368	.346	.346
ROBERTA	Yes	.469	.443	.453
XLNet	No	.335	.312	.308
ALNE	Yes	.440	.409	.417
DEDT (1)	No	.264	.291	.273
BERT (uncased)	Yes	.459	.368	.383
DEDT (I)	No	.309	.304	.295
BERT (cased)	Yes	.408	.365	.379
D:-4:IDEDT (4)	No	.263	.277	.266
DistilBERT (cased)	Yes	.458	.327	.338
D:-4:IDEDT (1)	No	.302	.275	.267
DistilBERT (uncased)	Yes	.412	.353	.363

Table 6: Macro-averaged scores for task C on the final test dataset. P stands for Precision, R for Recall. RoBERTa with DA was the model used to create our submission on the test dataset.

Table 7 shows the results of RoBERTa with data augmentation for each vector. The model could not classify any instance of the vector "3.4 condescending explanations or unwelcome advice", which only has 14 instances in the test dataset, and 47 in the training dataset. Although our model was trained with synthetic examples (in particular, 94 for this label), the total number of examples for this vector is still very scarce. Although the vector "1.2 incitement and encouragement of harm" is not one of the vectors with the largest number of instances, it does show the best F1-score (0.657). As was previously discussed for category 1, the texts classified with this vector tend to be shorter and include very violent words such as "bitch", "fuck", "kill", or "kick". The vector "3.1 casual use of gendered slurs, profanities, and insults" achieve the second highest F1-score (0.646). Vector 3.1 is the third

Category	P	R	F1	Instances
1.1	.461	.375	.413	16
1.2	.632	.684	.657	73
2.1	.552	.541	.546	205
2.2	.497	.572	.532	192
2.3	.436	.421	.428	57
3.1	.644	.648	.646	182
3.2	.487	.495	.491	119
3.3	.384	.277	.322	18
3.4	0	0	0	14
4.1	.571	.380	.457	21
4.2	.5	.479	.489	73

Table 7: Results provided by RoBERTa and data augmentation on the test dataset (task C) for the 11 vectors: "1.1 threats of harm", "1.2 incitement and encouragement of harm", "2.1 descriptive attacks", "2.2 aggressive and emotive attacks", "2.3 dehumanising attacks and overt sexual objectification", "3.1 casual use of gendered slurs, profanities, and insults", "3.2 immutable gender differences and gender stereotypes", "3.3 backhanded gendered compliments", "3.4 condescending explanations or unwelcome advice", "4.1 supporting mistreatment of individual women", "4.2 supporting systemic discrimination against women as a group". P stands for Precision, R for Recall.

one with the highest number of instances in the dataset. Regarding the other vectors, we observe that the fewer instances a vector has, the lower the F1-score it obtains.

When RoBERTa is trained without using synthetic data, it can not classify any instance of the three vectors: 1.1, 3.3, and 3.4. Therefore, data augmentation techniques improve the results of task C.

# 5 Conclusion

Our team participated in the three tasks with an approach based on RoBERT fine-tuned with training data and synthetic data created by data augmentation techniques. This approach shows very modest results on the three tasks (our systems approximately rank in the middle of the three rankings). We still have much room for improvement, especially in the two fine-grained classifications. While data augmentation does not achieve a significant improvement in task A, it obtains a positive effect on the results in task C. Although the use of augmented data provided the best results for task B during the development phase, our final results on the final test dataset show the opposite.

In future work, we plan to extend our research on data augmentation techniques to augment the training data. For example, we plan to use back translation (Sugiyama and Yoshinaga, 2019). In addition, we will exploit other datasets for the detection of sexist content, such as the EXIST dataset (Rodríguez-Sánchez et al., 2021) or MAMI (Fersini et al., 2022), to also approach the task from two different scenarios: multilingual and multimodal.

# 6 Acknowledgments

This work is part of the R&D&i ACCESS2MEET project (PID2020-116527RB-I0), financed by MCIN AEI/10.13039/501100011033/.

## References

- Neli Arabadzhieva-Kalcheva and Ivelin Kovachev. 2022. Comparison of bert and xlnet accuracy with classical methods and algorithms in text classification. In 2021 International Conference on Biomedical Innovations and Applications (BIA), volume 1, pages 74–76. IEEE.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.
- Elisabetta Fersini, Francesca Gasparini, Giulia Rizzi, Aurora Saibene, Berta Chulvi, Paolo Rosso, Alyssa Lees, and Jeffrey Sorensen. 2022. Semeval-2022 task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 533–549.
- Hannah Rose Kirk, Wenjie Yin, Bertie Vidgen, and Paul Röttger. 2023. SemEval-2023 Task 10: Explainable Detection of Online Sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Association for Computational Linguistics.
- Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open*, 3:71–90.
- Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3):1–40.
- Cecilia L Ridgeway. 2011. Framed by gender: How gender inequality persists in the modern world. Oxford University Press.

- Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, and Laura Plaza. 2020. Automatic classification of sexism in social networks: An empirical study on twitter data. *IEEE Access*, 8:219563–219576.
- Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, Laura Plaza, Julio Gonzalo, Paolo Rosso, Miriam Comet, and Trinidad Donoso. 2021. Overview of exist 2021: sexism identification in social networks. *Procesamiento del Lenguaje Natural*, 67:195–207.
- Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, Laura Plaza, Adrián Mendieta-Aragón, Guillermo Marco-Remón, Maryna Makeienko, María Plaza, Julio Gonzalo, Damiano Spina, and Paolo Rosso. 2022. Overview of exist 2022: sexism identification in social networks. *Procesamiento del Lenguaje Natural*, 69:229–240.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *The 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing @ NeurIPS 2019*.
- Amane Sugiyama and Naoki Yoshinaga. 2019. Data augmentation using back-translation for context-aware neural machine translation. In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 35–44.
- Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389, Hong Kong, China. Association for Computational Linguistics.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.
- Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

## A Appendix

In this section, we provide supplementary material for our research.

Figure 5 shows the distribution of the text length for the vectors within each category: "1. threats, plans to harm and incitement", "2. derogation", "3. animosity", and "4. prejudiced discussions".

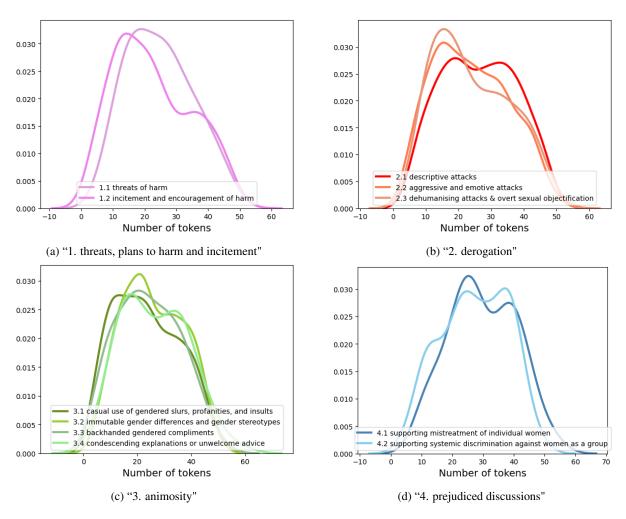


Figure 5: Density graph of the length of texts for the vectors within each category.