TheDBOs@GermEval Shared Task 2025: Supporting BERT Model Training with LLM-Based Synthetic Tweets for Harmful Content Detection

Christoph Papadatos and Claudia Grote and Florian Ludwig
ZITiS
Zamdorfer Straße 88, 81677 Munich
christoph.papadatos@zitis.bund.de
claudia.grote@zitis.bund.de
florian.ludwig@zitis.bund.de

Abstract

Imbalanced and limited data poses significant challenges in the context of harmful content detection. By participating in subtask 2 of the GermEval's 2025 Harmful Content Detection challenge, we investigate the potential of Large Language Models (LLMs) in generating synthetic data to address this issue. We propose a four-step approach to generate and validate a synthetic dataset, which is then used to train BERT-based classification models for detecting tweets containing potential attacks on the democratic basic order (DBO). Our experiments demonstrate that incorporating synthetic data can significantly enhance model recall and precision, leading to improved overall performance. Crucially, our results highlight that careful selection of high-quality synthetic samples is essential for achieving these gains.

1 Introduction

Despite notable progress in automatic detection of hate speech, grasping the subliminal expressions of harmful content remains a challenge to natural language processing (NLP) systems (Albladi et al., 2025). The difficulty of recognition is grounded in the language of hateful content, which is variant in context and takes advantage of pernicious stylistic and rhetorical devices (Kennedy et al., 2022).

Reliably labelled harmful textual content is rare, causing data of interest and variation to form minority classes and unfavourable class imbalance (Zhang et al., 2024; Madukwe et al., 2020; Rathpisey and Adji, 2019). The impact of class imbalance on performance in NLP tasks cannot be underestimated and should be addressed according to the respective task (Henning et al., 2023). Due to the expense of intellectually annotating data, generating additional high-quality and variant training data is a current and important topic in research.

For our solution to Subtask 2 of the GermEval 2025 "Harmful Content Detection" competition (Felser et al., 2025), we developed a sophisticated method for generating synthetic training data. This generated data was designed to augment the provided dataset for classifying attacks on the free democratic basic order (DBO).[1] With our work, we make the following contributions:

- We propose a four-step approach for generating and evaluating synthetic data samples to support the training of machine learning models in detecting attacks on the democratic basic order.
- We investigate the suitability of different strategies for selecting synthetic samples for machine learning model training.
- We show the competitiveness of our solution for the GermEval 2025 "Harmful Content Detection" Shared Task.

An overview of related work relevant to the present study is provided in Section 2, followed by a brief description of the task and data in Section 3. The details of our methodological framework are presented in Section 4: We describe the prompts used and the few-shot examples selected using three different methods. Furthermore, we outline the process of synthetic tweet generation and the subsequent filtering of the generated tweets using validation models, resulting in the final synthetic corpus. Section 5 describes the baseline, the experimental setup and the datasets consisting of different subsets of the synthetic data used for three experiments. We discuss the results in Section 6,

[1] in German "Freiheitliche Demokratische Grundordnung", refers to the unalterable key values and principles constituting the core of the German democratic system, cp. https://www.verfassungsschutz.de/EN/about-us/mission-and-working-methods/protecting-the-constitution/protecting-the-constitution_article.html

Proceedings of the 21st Conference on Natural Language Processing (KONVENS), Volume 2: Workshops, pages 366–374, 2025

Label	Count	Proportion
Nothing	6,277	.84
Criticism	804	.11
Agitation	313	.042
Subversive	60	.008
Total	7,454	1

Table 1: Distribution of the training data according to their labels.

and in Section 7, we conclude the study by highlighting the most important findings.

2 Related Work

Struß et al. (2019) reports on the success of BERT-based classification models, shortly after the introduction of BERT by Devlin et al. (2019), at the GermEval 2019 "Identification of Offensive Language" challenge in the same year. As Albladi et al. (2025) shows in their review, several approaches to improve the performance of encoder models on the classification of harmful content have been introduced, but the problems of data scarcity and lack of variance in the data continue to negatively affect performance. Our approach addresses these causes and aims to improve the classification performance of BERT classifiers by providing additional and variant synthetic training data.

In order to generate annotated training data on a large scale to classify punishable offensive content, the decompositional approach described in Zufall et al. (2019) breaks down a jurisprudential annotation guideline into consecutive binary decisions that are comprehensible for legal laypeople to be followed as instructions for annotation. We apply a similar strategy to guide an LLM, using class descriptions instead, to generate appropriate data using few-shot prompting.

Variant toxic-data augmentation using LLMs has already proven successful. Juuti et al. (2020) compares different data augmentation approaches on several toxic language classifiers. Their binary BERT classifier gained best performance with additional training data generated by an LLM after fine-tuning on minority-class content, in combination with other data. Instead of resource-consuming fine-tuning of an LLM, we rely on the inherent language-understanding capabilities of the LLM with few-shot prompting.

Casula et al. (2024) describes the generation of additional training data by prompting an LLM to paraphrase original data and subsequent filtering to remove low-quality data. This differs from our approach, as we create novel synthetic data based on few-shot examples using an LLM and rank these through majority voting afterwards.

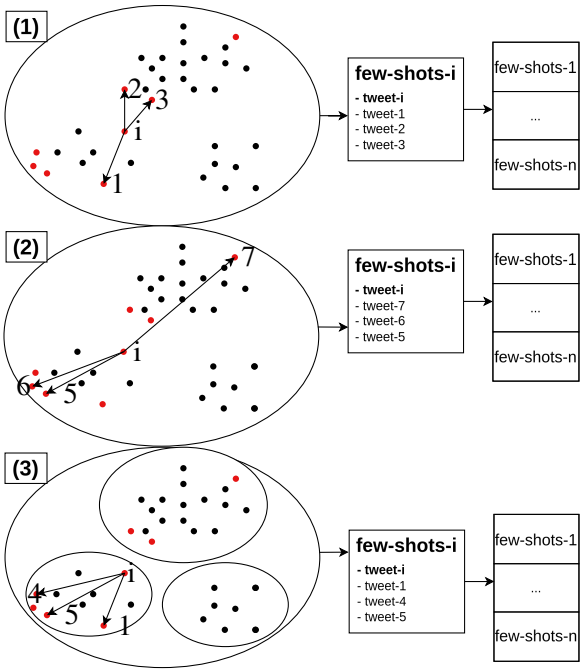

Figure 1: Illustration of the three methods for selecting few-shot examples from tweets that share the same class label l as a reference tweet i. Tweets of class l are shown as red dots. All methods use semantic similarity, measured by cosine similarity in the embedding space, to identify suitable candidates. (1) selects the k most similar tweets, (2) the k most dissimilar tweets. (3) uses topic clustering and selects the k most similar tweets within the same cluster. After selection, the tweets are ordered by their cosine similarity to i. The reference tweet and the selected tweets form a list of view shot examples.

3 Description of the Task and Data

The GermEval 2025 competition on "Harmful Content Detection" (Felser et al., 2025) concerned the classification of German-language social media posts. Subtask 2 addressed the challenge of identifying attacks on the free democratic basic order (DBO) by predicting the membership to one of these four categories for each tweet: (1) *nothing* (no attack on the DBO), (2) *criticism* (legitimate verbal criticism of the government or parties), (3) *agitation* (worrying attacks expressing agitative aspirations), or (4) *subversive* (calls for or support of the overthrow of the government). For this reason, a new dataset, consisting of annotated 7,454 tweets for training purposes and 3,194 tweets for testing purposes, was introduced. This dataset comprises

tweets from a German group considered extremist posted from December 2014 to July 2016 and was annotated by members of Mittweida University of Applied Sciences. The tweets were anonymised with respect to the mentions of usernames. The training data is quantitatively distributed as shown in Table 1.

The unmarked class, *nothing*, holds 84% of the training data, and the size of the smallest minor class, *subversive*, is less than 1% of the cardinality of the majority class, such that the distribution of the training data across the classes is extremely unbalanced.

4 Methods

In this work, we propose a four-step approach for generating suitable synthetic data samples. We start with the selection of few-shot examples, which are used to guide the generation of appropriate synthetic data samples with a LLM (Section 4.1). Next, we generate synthetic data samples with the help of class-specific prompts and the few-shot samples, which were selected in the first step (Section 4.2). We then train an ensemble of classifiers to assess the reliability of the generated synthetic data (Section 4.3). Each sample is assigned an agreement score based on the number of models that correctly predict its intended class. Samples with higher scores are considered more reliable. In the final step, we employ the previously trained classifiers to validate and pre-filter the synthetic data (Section 4.4).

4.1 Few-Shot Selection

In the first step, we prepare few-shot examples for the LLM to guide the generation of synthetic data. We propose three different strategies to select lists of few-shot examples from the training data. The common idea of all three methods is to consider each tweet of the training data as a reference for the selection of k additional tweets of the same class and from the same training set. The reference tweet, together with these additional selected tweets, forms a list of few-shot examples. The additional tweets are arranged in descending order according to their similarity to the reference tweet. For each method, n of these few-shot lists are collected in advance, one list for each of the n tweets in the training set. Figure 1 visualises the following strategies.

(1) Semantic Similarity-Based Selection: The first selection strategy aims to group semantically similar tweets as few-shot examples. For each reference tweet, this is done by considering the similarities to all tweets of the same class in the training corpus. In order to compute the similarities, all tweets were embedded using the embedding model proposed in (Mohr et al., 2024). The k nearest tweets are selected based on cosine similarity on these embeddings and form, together with the reference tweet, the list of few-shot examples.

(2) Semantic Dissimilarity-Based Selection: The second selection strategy selects the k most dissimilar tweets of the same class in order to form the few-shot list. As for the previous selection strategy, similarity is measured using cosine similarity on tweet embeddings. The goal of this selection strategy is to provide the LLM with few-shot examples that cover diverse semantic content, thus encouraging the generation of synthetic tweets that span a broad range of topics.

(3) Topic-Based Selection: We use a BERTopic model (Grootendorst, 2022) to cluster the training data into topics. We assume that tweets, which share the same topic cluster and the same class label, are contextually related, making them suitable as few-shot examples. To select these examples, we retrieve the k nearest tweets to a reference tweet within the same cluster using cosine similarity on tweet embeddings, ensuring topical alignment.

4.2 Synthetic Tweet Generation

The next step of the proposed approach involves the generation of a synthetic training dataset. For this reason, over 200,000 synthetic tweets were generated using the previously constructed few-shot examples and class-specific prompts.

4.2.1 Prompt Design

Synthetic data generation is guided by class-specific prompts, each following a standardised template consisting of up to five components (an exemplary prompt is shown in Appendix A).

(1) Introduction: The first part of the prompt serves as an introduction, outlining the task, and offering guidance on the characteristics of those tweets that belong to the desired class.

(2) Definition: The second part of the prompt provides the German translations of the class def-

initions. This is intended to describe the scope of the tweets the LLM should follow.

(3) Instruction: The third part defines the expected output: it specifies the characteristics of the synthetic tweets to be generated for a given class. This includes a short description of the class as well as the number of tweets to generate.

(4) Few-Shot Examples: This part provides class-specific few-shot examples (see Section 4.1) that the LLM should use as a reference. These examples serve to guide the model in generating tweets that match their style and semantics.

(5) Additional Topic: To address the limited number of available examples in the *subversive* and *agitation* classes, we extended the previous parts of the prompts for these classes by an additional component. This component introduces a topic placeholder, allowing the LLM to generate synthetic tweets on a specified topic, potentially increasing topical diversity within the synthetic dataset.

4.2.2 Generation Pipeline

We treat each tweet in the training data as a reference tweet. For each reference tweet, we identify its class label and use its corresponding class-specific Introductions and Instructions (see Section 4.2.1). For each few-shot selection strategy (Section 4.1), a list of few-shot examples is selected based on the current reference tweet and class label. For the *agitation* and *subversive* classes, the prompts are further extended with a specific topic (see Section 4.2.1), resulting in separate prompts for each topic. The composed prompts are used by a large language model (LLM) during inference to generate synthetic data (see Section 5.5 for details on the LLM used).

4.3 Training of Validation Models

To evaluate the reliability of the generated synthetic tweets, we classify each tweet using an ensemble of German BERT-based models. Since the prompts are class-specific, generated synthetic tweets are considered to belong to their associated classes. For each tweet, we define its reliability score as the number of models in the ensemble that correctly predict its assigned class. Tweets that receive higher reliability scores are regarded as more reliable.

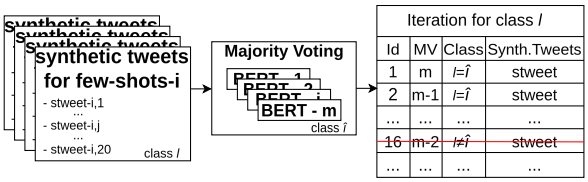

Figure 2: Reliability scoring of synthetic tweets using an ensemble of BERT-based classifiers. Each tweet is assigned a reliability score based on how many models correctly predict its associated class. Higher scores indicate greater reliability. Tweets with a low reliability score are discarded.

All validation models in the ensemble are trained using different variations of the training data. These variations include the following label aggregations and subsets of class labels.

- The original training dataset containing all four classes: *nothing*, *criticism*, *agitation*, and *subversive*
- A modified dataset with two aggregated classes: *nothing* vs. the combined class of *criticism*, *agitation*, and *subversive*
- A filtered dataset excluding the *nothing* class, containing only tweets labeled as *criticism*, *agitation*, or *subversive*

For a detailed overview of the models used, we refer to Section 5.5.

4.4 Pre-Filtering of the Synthetic Data

In the final step of our approach, we use the reliability scores assigned to the synthetic tweets to pre-filter the dataset (see Figure 2). Tweets with reliability scores below half the size of the model ensemble were discarded. The tweets that remain after this filtering process form the final synthetic corpus.

5 Experiments

This section presents an overview of our experimental setup. Model evaluation was conducted on the provided training set using 4-fold cross-validation.

5.1 Baseline Experiment

For our baseline experiment $E0$, we trained German BERT models (Chan et al., 2020) on the training dataset provided by the competition organisers, without including any synthetic data.

	$E1$	$E2$	$E3$
Nothing	14,672	0	0
Criticism	14,672	1,299	686
Agitation	14,672	1,545	1101
Subversive	10,593	1,513	1,398
Aggregate	54,609	4,357	3,185

Table 2: Counts of synthetic tweets with respect to each class label and experiment ($E1$ - $E3$).

5.2 Automatic Selection of Synthetic Data

For experiment $E1$, synthetic tweets were automatically selected based on their reliability scores. We identified the two highest reliability scores within each class and selected all tweets with scores matching these top two values. Due to the disproportionately large number of available examples for the classes *criticism* and *nothing* in the provided training dataset, a correspondingly larger number of synthetic tweets were generated for these classes. As a result, these classes also have a much higher number of synthetic samples with top reliability scores. To prevent class imbalance in the synthetic training set, we randomly sub-selected samples from these high-resource classes to roughly match the number selected for the other classes. The final counts of synthetic samples per class are detailed in Table 2 ($E1$). The synthetic samples were used to enhance the training data. As in the previous experiment, this dataset is then used to train German BERT models.

5.3 Manual Selection of Synthetic Data

In the second experiment $E2$, we drew a random sample from the synthetic dataset. Because the *nothing* class was already sufficiently represented in the original training set, we limited the sampling to the classes *criticism*, *agitation*, and *subversive*. Each synthetic tweet, along with its assigned class label, was manually reviewed, and only those judged to be correctly classified were retained for model training. Due to time constraints imposed by the competition timeline, we were unable to manually review as many samples as had been automatically selected in Experiment $E1$. The resulting number of synthetic training samples chosen for this experiment are shown in Table 2 ($E2$). As in the previous experiments, the selected synthetic data samples were used to enhance the provided training dataset for BERT model training.

5.4 Combining Manual and Automatic Selection of Synthetic Data

In this experiment, we filter the synthetic data, used in the previous experiment $E2$, according to their reliability scores. We selected all tweets whose scores matched the top two reliability scores within their respective class. This procedure resulted in the smallest number of samples, as shown in Table 2 ($E3$), which, however, were considered to be of the highest quality.

5.5 Experimental Parameters

In this section, we briefly describe the base models used and the parameters we chose for our approaches.

Topic Modelling: The BERTopic (Grootendorst, 2022) model was trained with the embeddings based on the embedding model proposed in (Mohr et al., 2024). To obtain fine-grained topic clusters, we applied `UMAP` and `HDBSCAN` with the following parameters: $n\_neighbors = 5$, $n\_components = 50$, and $min\_topic\_size = 5$. Additionally, we configured the `CountVectoriser` to use uni-, bi-, and trigrams, and to ignore all German stop words, to improve the quality of the topic representations.

Validation Models: The following list shows the models chosen as validation models (Section 4.3).

- `google-bert/bert-base-german-cased`[2]
- `EuroBERT/EuroBERT-610m` (Boizard et al., 2025)
- `deepset/gbert-base` (Chan et al., 2020)
- `deepset/gbert-large` (Chan et al., 2020)
- `deepset/gelectra-large` (Chan et al., 2020)
- `deepset/gbert-large-sts`[3]
- `answerdotai/ModernBERT-large` (Warner et al., 2024)

Setup for Generation Pipeline: The synthetic tweets were generated using an uncensored version of the `LLaMA 3.0 70B` model deployed via Ollama[4].

[2]Released by deepset.ai (June 2019) https://www.deepset.ai/, downloaded at https://huggingface.co/google-bert/bert-base-german-cased

[3]Released by deepset.ai (August 2021) https://www.deepset.ai/, downloaded at https://huggingface.co/deepset/gbert-large-sts

[4]https://ollama.com/taozhiyuai/llama-3-uncensored-lumi-tess-gradient:70b-q8_0

	Precision	Recall	F1-Score	F1-Score
	Cross-Validation			*Testset*
$E0$	.69	.68	.68	-
$E1$	.61	**.80**	.67	.67
$E2$	.73	.73	.72	.68
$E3$	**.74**	.73	**.73**	**.69**

Table 3: Classification metrics for different experiments *E0 - E3*. Columns 2–4 show the average results of 4-fold cross-validation on the provided training dataset. The final column reports the F1-score on the provided test set, calculated using the best-performing model selected from each experiment.

Final Models: For our experiments $E0$ $E1$, $E2$ and $E3$, we finetuned a pre-trained German BERT model (Chan et al., 2020). The models were trained for 20 epochs with a batch size of 16, using a fused AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 2×10^{-5}.

6 Results and Discussion

The classification results (see Table 3) highlight the critical role of data quality in achieving strong model performance. Synthetic samples selected solely through the automatic selection method appear to be of the lowest quality. This is indicated by the models trained in Experiment $E1$, which achieved the lowest F1-Score despite being trained on the largest amount of data. Although they achieved high recall, likely due to greater data variance, these models suffered from low precision due to frequently and incorrectly classifying *nothing* cases as relevant instances. In contrast, models trained on a moderate amount of manually selected synthetic data (experiment $E2$) outperformed both the baseline models (experiment $E0$) and those trained on a large quantity of lower-quality data (experiment $E1$). These models achieved significantly higher precision with only a slight decrease in recall, resulting in a higher overall F1-Score. This improvement can be attributed to the improved data quality, despite using considerably fewer synthetic data samples than the models in $E1$. The best performance was achieved by models trained exclusively on highest-quality synthetic data, which was ensured by both manual and automatic filtering of the synthetic data (experiment $E3$). While these models were trained with the least amount of data, these models not only maintained the recall observed in the previous experiment $E2$, but also further improved its precision. As a result, these models recorded the best overall F1-Score. The results obtained on the competition's test data support these findings (see Table 3, last column). The best-performing model from experiment $E3$ achieved the best performance overall, while the best-performing model from experiment $E1$ performed the worst.

Overall, we observed that the most reliable data samples were obtained by combining human judgment with automated filtering, leveraging the strengths of both approaches. While increasing the volume of data can improve recall, it does not necessarily lead to better overall results if the data quality is poor. In contrast, training with higher-quality data consistently enhances precision and overall model effectiveness. Consequently, we propose that future research focus on developing automated selection strategies to filter high-quality synthetic data for model training, thus reducing the effort of manual data inspection.

7 Conclusion

In this paper, we demonstrated that generating high-quality synthetic data samples is a suitable approach to address data scarcity and class imbalance in the GermEval 2025 Shared Task on Harmful Content Detection (Felser et al., 2025). Our findings, based on cross-validation results on the provided training dataset, highlight that careful selection of synthetic examples significantly impacts model performance and robustness. The best-performing models were then evaluated on the competition's unseen testset, where they maintained strong performance consistent with the development results. These outcomes positioned our approach among the top three teams in the competition, validating the effectiveness and generalizability of our method. Future work will focus on automating the selection of synthetic data to further improve model accuracy and reliability.

Limitations

This system description has certain limitations. Most notably, we applied our approach exclusively on subtask 2 of the GermEval 2025 challenge. Furthermore, we only investigate one model for synthetic tweet generation.

Ethics Statement

We use publicly available, anonymised data from the GermEval 2025 shared task. While this ensures reproducibility, we acknowledge that the dataset may contain biases and harmful content. Hate

speech detection remains a subjective and context-dependent task, and automated systems may produce false positives or negatives. The tweet generation methods and the resulting synthetic tweets are intended solely to support model training in the detection of harmful content. Our models are intended solely for research and should not be deployed in real-world applications without careful consideration of fairness, societal impact, and the need for human oversight.

References

Aish Albladi, Minarul Islam, Amit Das, Maryam Bigonah, Zheng Zhang, Fatemeh Jamshidi, Mostafa Rahgouy, Nilanjana Raychawdhary, Daniela Marghitu, and Cheryl Seals. 2025. Hate speech detection using large language models: A comprehensive review. *IEEE Access*, 13:20871–20892.

Nicolas Boizard, Hippolyte Gisserot-Boukhlef, Duarte M Alves, André Martins, Ayoub Hammal, Caio Corro, Céline Hudelot, Emmanuel Malherbe, Etienne Malaboeuf, Fanny Jourdan, and 1 others. 2025. Eurobert: scaling multilingual encoders for european languages. *arXiv preprint arXiv:2503.05500*.

Camilla Casula, Sebastiano Vecellio Salto, Alan Ramponi, and Sara Tonelli. 2024. Delving into qualitative implications of synthetic data for hate speech detection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19709–19726, Miami, Florida, USA. Association for Computational Linguistics.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. *arXiv preprint arXiv:2010.10906*.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jenny Felser, Michael Spranger, and Melanie Siegel. 2025. Overview of the germeval 2025 shared task on harmful content detection. In *Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Workshops*.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. 2023. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–540, Dubrovnik, Croatia. Association for Computational Linguistics.

Mika Juuti, Tommi Gröndahl, Adrian Flanagan, and N. Asokan. 2020. A little goes a long way: Improving toxic language classification despite data scarcity. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2991–3009, Online. Association for Computational Linguistics.

Brendan Kennedy, Mohammad Atari, Aida Mostafazadeh Davani, Leigh Yeh, Ali Omrani, Yehsong Kim, Kris Coombs, Shreya Havaldar, Gwenyth Portillo-Wightman, Elaine Gonzalez, Joe Hoover, Aida Azatian, Alyzeh Hussain, Austin Lara, Gabriel Cardenas, Adam Omary, Christina Park, Xin Wang, Clarisa Wijaya, and 3 others. 2022. Introducing the gab hate corpus: defining and applying hate-based rhetoric to social media posts at scale. *Lang. Resour. Eval.*, 56(1):79–108.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Kosisochukwu Madukwe, Xiaoying Gao, and Bing Xue. 2020. In data we trust: A critical analysis of hate speech detection datasets. In *Proceedings of the fourth workshop on online abuse and harms*, pages 150–161.

Isabelle Mohr, Markus Krimmel, Saba Sturua, Mohammad Kalim Akram, Andreas Koukounas, Michael Günther, Georgios Mastrapas, Vinit Ravishankar, Joan Fontanals Martínez, Feng Wang, and 1 others. 2024. Multi-task contrastive learning for 8192-token bilingual text embeddings. *arXiv preprint arXiv:2402.17016*.

Heng Rathpisey and Teguh Bharata Adji. 2019. Handling imbalance issue in hate speech classification using sampling-based methods. In *2019 5th International Conference on Science in Information Technology (ICSITech)*, pages 193–198. IEEE.

Julia Maria Struß, Melanie Siegel, Josep Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 354–365, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, and 1 others. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

Yaqi Zhang, Viktor Hangya, and Alexander Fraser. 2024. A study of the class imbalance problem in abusive language detection. In *Proceedings of the 8th Workshop on Online Abuse and Harms (WOAH 2024)*, pages 38–51, Mexico City, Mexico. Association for Computational Linguistics.

Frederike Zufall, Tobias Horsmann, and Torsten Zesch. 2019. From legal to technical concept: Towards an automated classification of German political Twitter postings as criminal offenses. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1337–1347, Minneapolis, Minnesota. Association for Computational Linguistics.

A Prompts

This section contains the prompts used for generating synthetic data.

Prompt Part 1

Gebe mir Beispiele von Tweets, die in einer der Kategorien sind, die unten aufgelistet sind. Sie sollen keine Angriffe auf die freiheitliche und demokratische Grundordnung der Bundesrepublik Deutschland beinhalten.

Prompt Part 2

subversive: Ein Wille wird geäußert, die bestehende Regierung mit Gewalt zu stürzen und zu zerstören (z.B. durch militanten Einsatz, Störung des Stromnetzes usw.).

agitation: Agitative Bemühungen werden geäußert. Dazu gehören die Bekanntgabe von Aktionen wie der Verbreitung von Propagandamaterial unkonstitutioneller und terroristischer Organisationen oder die Beschädigung von Staatsymbolen wie der Fahne der Bundesrepublik Deutschland.

criticism: Tweets, in denen legitime Kritik an der Regierung, Amtsträgern, Regierungsangestellten, Behörden oder Parteien geäußert wurde, wurden dieser Klasse zugeordnet.

nothing (kein Angriff auf die freiheitlich-demokratische Grundordnung): Der Tweet enthält keinen Angriff auf die freiheitliche demokratische Grundordnung. Auch neutrale oder positive Aussagen über Regierungsentscheidungen fallen in diese Kategorie. Generell alle Tweets, die nicht in die Kategorie subversive, agitation oder criticism fallen.

Prompt Part 3

Liste mir 20 Beispiel-Tweets auf in deutscher Sprache der Kategorie nothing. Formuliere Tweets und achte darauf, dass ein paar wenige Tweets auch neutrale oder positive Aussagen über Regierungsentscheidungen beinhalten. Schreibe in den übrigen Tweets über die Themen, die in den Beispielen vorkommen!

Prompt Part 4

Orientiere dich dabei an folgenden Beispielen: {few_shots}

Prompt 1: Prompting to obtain completions of class *nothing*. Each prompt is composed of four parts. Part 1 consists of a general prefix to describe the task of proposing examples of tweets belonging to one of the categories described in part 2, in this case, class *nothing* is addressed. Part 2 describes die four classes of the multi-class scheme using unambiguous formulations. Part 3 states the desired quantity of completions, language and category label. It also contains a piece of advice to include neutral or positive mentions of governmental decisions, whereas the other completions have to adopt the topics found in the examples in part 4. Part 4 lists examples operating as few-shots.