Word-Level Detection of Code-Mixed Hate Speech with Multilingual Domain Transfer

Karin Niederreiter
University of Vienna
karin.niederreiter@outlook.com

Dagmar Gromann
University of Vienna
dagmar.gromann@gmail.com

Abstract

The exponential growth of offensive language on social media tends to fuel online harassment and challenges detection mechanisms. Hate speech detection is commonly treated as a monolingual or multilingual sentence-level classification task. However, profane language tends to contain code-mixing, a combination of more than one language, which requires a more nuanced detection approach than binary classification. A general lack of available code-mixed datasets aggravates the problem. To address this issue, we propose five word-level annotated hate speech datasets, EN and DE from social networks, one subset of the DE-EN Offensive Content Detection Code-Switched Dataset, one DE-EN code-mixed German rap lyrics held-out test set, and a cross-domain held-out test set. We investigate the capacity of fine-tuned German-only, German-English bilingual, and German-English code-mixed token classification XLM-R models to generalize to code-mixed hate speech in German rap lyrics in zero-shot domain transfer as well as across different domains. The results show that bilingual fine-tuning facilitates not only the detection of code-mixed hate speech, but also neologisms, addressing the inherent dynamics of profane language use.

1 Introduction

Warning: *This paper contains content that may be offensive or upsetting.*

Ever-growing offensive online contents can negatively impact humans (Nozza and Hovy, 2023; Jagdale et al., 2024), which can be addressed with effective detection mechanisms. Considerable attention from research to this task has emphasized the need for robust and effective as well as multilingual methods (Nkemelu et al., 2022; Nozza and Hovy, 2023). One growing challenge in language detection tasks is code-mixing (Aguilar et al., 2020; Chakravarthi et al., 2020; Salaam et al., 2022; Shankar et al., 2024)

Dataset	Input and Output
German	Hurensohn manchmal denk ich mir einfach nur fick dich ['B-B', 'n', 'n', 'n', 'n', 'n', 'n', 'B-B', 'B'] (Son of a bitch, sometimes I just think fuck you)
English	Holy shit man , I got head last night ['B-B', 'B', 'n', 'n', 'n', 'B-B', 'B', 'n', 'n']
Code-Mixed	Ah , was ist das für ein **Muschibattle** ['n', 'n', 'n', 'n', 'n', 'n', 'n', 'B-B'] (Ah, what a pussy battle is this)
Code-Mixed	**All the way up** von der Mittelschicht ['n', 'n', 'n', 'n', 'n', 'n', 'n'] (All the way up from middle class)

Table 1: Dataset examples and word-level annotation scheme.

Code-mixing can occur inter-sententially, intra-sententially, and within phrases or compounds. Even morphemes from one language can be embedded in words from another (Bohra et al., 2018). In combination with profane language, the complexity of the hate speech detection task is considerably increased by code-mixing. Since code-mixing combines multiple languages, resources that contain these combinations of languages are extremely scarce. This challenge has been addressed with approaches to synthetically create code-mixed datasets (e.g. Salaam et al., 2022). However, such datasets might not accurately represent the human

Findings of the Association for Computational Linguistics: ACL 2025, pages 21093–21104

tendency to dynamically create novel profanities.

To address this limitation, we contribute five word-level, manually annotated hate speech datasets, a lexicon-based English and German dataset from social media, a subset of the DE-EN Offensive Content Detection Code-Switched Dataset (OCD) (Salaam et al., 2022), a cross-domain held-out test set and a code-mixed German-English rap lyrics dataset from the German rap domain. German rap, a subgenre of hip hop, is primarily performed in German and is heavily influenced by U.S. American hip hop. It often includes multilingual code-mixing and profane language, with some artists using more than 10 different languages in their lyrics (Tikhonov, 2020).

As an additional contribution, we evaluate the impact of language-specific fine-tuning on zero-shot domain transfer to code-mixed hate speech as well as across domains, which is compared to the zero-shot performance of state-of-the-art Large Language Models (LLMs). Although the size of the proposed datasets is comparatively limited, this paper contributes a real-world benchmark on multilingual as well as code-mixed word-level hate speech detection, as exemplified in Table 1. The German rap domain is particularly interesting for its profane examples of intra-sentential and intra-phrase code-mixing as well as neologisms as highlighted in Table 1 in bold. To the best of our knowledge, this paper proposes the first datasets and approaches to combine word-level annotation of hate speech and code-mixing, while also considering neologisms[1].

2 Related Work

Hate speech detection has been a highly active field of research over the past decades. Since, to the best of our knowledge, no previous publication considers code-mixing and word-level annotation, this section focuses on approaches that address code-mixed (profane) language and word-level hate speech detection.

Code-Mixed Hate Speech. A majority of research on detecting hate speech in code-mixed examples specializes in English-Hindi. The work that is probably closest to the proposed approach provides a code-mixed English-Hindi hate speech dataset of 4.575 tweets (Bohra et al., 2018). While the language identifier is annotated at word-level, the hate speech label is assigned to entire sentences. Jagdale et al. (2024) compare BERT (Devlin et al., 2019) with HingBERT, a model trained on Hindi-English, and with code-mixed FastText embeddings on the same language pair on detecting code-mixed hate speech, finding that models and embeddings trained on code-mixed data perform better. They use the Hate Speech and Offensive Content Identification (HASOC) dataset (Modha et al., 2021), which provides a shared task and excellent testbed for English, Hindi, and Marathi hate speech examples from Twitter (now X).

Liyanage and Jayakumar (2021) collect Sinhala-English code-mixed examples from Facebook and YouTube and evaluate several statistical machine learning methods on the related hate speech detection task on sentence level, finding ensemble methods to perform best. A similar comparison that also included neural networks was performed by Dhanya and Balakrishnan (2024) on a code-mixed Malayalam-English dataset, finding with little surprise that neural networks outperformed statistical machine learning models.

More closely related to the the languages at hand are Salaam et al. (2022), who propose the human-generated Offensive Content Detection Code-Switched Dataset (OCD) for EN-DE, EN-ES, and EN-FR by requesting bilingual crowd workers to translate and rewrite profane language from Twitter (now X) and Gab posts in the HateXplain dataset (Mathew et al., 2021) to introduce code-mixing. The authors additionally create a synthetic dataset by automatically detecting profane noun phrases, machine translating them, and reintegrating the translated phrases to the source text. While fine-tuning XLM-R (Conneau et al., 2020) on the synthetic training set achieves interesting and insightful results on the human-generated test set, the task is treated as binary sentence classification and fails to reflect on the dynamics of intra-phrase code-mixing in hate speech.

Word-Level Detection. Zampieri et al. (2022) perform Multi-Word Expression (MWE) identification for hate speech detection in English tweets using a lexicon-based and a neural network-based approach, the latter combining BERT with Bi-LSTM and CRF layers. Hind Saleh and Moria (2023) compare the better performing BERT model with hate speech-specific global word embeddings, applying them in a sentence classifier. Similarly, Mou et al. (2020) propose the SubWord Enriched and Signifi-

[1]The datasets and code are available at https://github.com/Meraki89/word-level-code-mixed-hate-speech

cant Word Emphasized (SWE2) framework, which focuses on word-level semantics and subword information, including BERT, LSTM plus attention, and FastText embeddings as methods. The most interesting result is the ability of the presented methods to resist character-level adversarial attacks, increasing the robustness of the approach, which, however, focuses only on English. One hate speech detection approach and dataset considers novel meanings of pejorative words in English (Hoeken et al., 2024), presenting the Hateful Word in Context Classification (HateWiC) task.

Our study focuses on the combination of these two main previously considered aspects of word-level detection and code-mixed hateful sequences. Since code-mixing tends to occur within sentences or even phrases/compounds, we utilize token classification as a method that considers subword information. Apart from analyzing the most challenging categories of pejoratively used sequences for the token classifiers, we specifically consider the dynamic nature of this type of language. Rap lyrics are particularly creative in their use of swear words, which makes them an excellent test bed for detecting code-mixed and novel hateful instances.

3 Dataset

All five datasets detailed in Table 2 contain word-level annotations of explicit hate speech. Two are non-code-mixed datasets for English and German and three are code-mixed datasets.

3.1 Non-Code-Mixed Datasets

Two datasets of word-level hate speech, English and German, were generated with a lexicon-based approach, starting from lists of hateful words from Wiegand et al. (2018), the Carnegie Mellon University School of Computer Science bad word list[2], and the List of Dirty Naughty Obscene and Otherwise Bad Words [3]. The seven most frequent swear words, which collectively make up 90% of online occurrences according to Wang et al. (2014), are considered primary swear words, which are *fuck*, *shit*, *ass*, *bitch*, *nigga*, *hell*, and *whore*. Aside from deduplicating the English and German swear word lists, three main criteria guided the final selection for a balanced and significant dataset: (i) variations of primary swear words, including orthographic variation, (ii) swear words with explicit sexual vulgar connotations, and (iii) swear words pertaining to the nine main hate target categories proposed by Silva et al. (2021). This process yielded a total of 215 German and 373 English swear words. Using these lists, context sentences were extracted from the May 2015 Reddit Comments dataset[4] for English, from a multilingual sentiment analysis dataset (Mollanorozy et al., 2023) and the Multilingual HateSpeech Dataset (Moosa and Najiba, 2022) for German.

3.2 Code-Mixed Datasets

Three additional datasets were created with code-mixed hate speech, one for training a model (OCD) and two for testing zero-shot domain transfer (Rap and Held-Out). The code-mixed training set was created by annotating a subset of the OCD dataset (Salaam et al., 2022) on word-level, which contains code-mixed German-English data, originally with sentence labels. The fourth, test dataset consists of German rap lyrics with substantial German-English code-mixed hate speech as well as purely German hateful sequences. The distribution of deduplicated hate speech categories following the categorization of Silva et al. (2021) is depicted in Figure 1. One additional and predominant category is that of sex, which refers to sexual intercourse or sex organs rather than sexual orientation or physical attributes, and its derogatory use within rap lyrics. Finally, a cross-domain held-out test set was created to evaluate the generalization performance across different domains. Context sentences for this dataset were sourced from the same domains as the previous datasets, but none of the examples or sentences are included in the other datasets. The held-out test dataset consists of 150 sentences in total, equally distributed across the four domains with or without hateful sequences: 50 multilingual context sentences, 25 in German and 25 in English, and 50 sentences with code-mixing from both the German rap domain and the German-English OCD dataset.

All datasets contain examples of sentences with and without hate speech, two without code-mixing and three with code-mixing. The distribution of sentences and labels per dataset is represented in Table 2. Single sentences may contain more than one swear word, which is particularly evident in the code-mixed rap lyrics, and even multi-word hate speech. Examples sentences with the anno-

[2] https://www.cs.cmu.edu/~biglou/resources/bad-words.txt

[3] https://github.com/LDNOOBW/List-of-Dirty-Naughty-Obscene-and-Otherwise-Bad-Words/

[4] https://www.kaggle.com/datasets/kaggle/reddit-comments-may-2015

Dataset	Hateful	Neutral	B-B	B
English	784	441	1041	254
German	393	192	481	53
OCD	152	98	210	63
Rap	85	46	115	2
Held-Out	121	29	185	28

Table 2: Dataset size by sentence (hateful/neutral) and label (B-B/B) count; B-B represents a bad word and B the profane continuation of a bad word

tation schema are provided in Table 1, showing intra-sentential and intra-word code-mixing highlighted in bold. The last line of Table 1 exemplifies a code-mixed example without hate speech and also monolingual sentences without hate speech are contained in all datasets.

3.3 Annotation Schema and Process

In the proposed annotation scheme, as depicted in Table 1, the 'n' tag marks non-hateful words or compounds, 'B-B' indicates a single bad word or the beginning of an explicit hate speech multi-word sequence, and 'B' indicates the continuation of such a multi-word sequence. This annotation is comparable and easily convertible to the BIO scheme, as used in Zampieri et al. (2022) for profane MWE identification, however, was intended to be domain-specific to bad words, that is, single or multi-word explicit hate speech sequences.

All five datasets were annotated by two annotators with German as first language, a high command of English corresponding to a C1 level of the Common European Frame of Reference for Languages, and both with a background in translation studies, linguistics, and computational linguistics. The annotation guidelines provided to the annotators mirrored the criteria described in Section 3.1 to ensure that all profanities, including those not present in the lexicon but conforming to the specified criteria, were accurately annotated. They also ensured that homonymy and polysemy were considered, e.g. *son* is considered hateful in *son of a bitch* but not in *tell them what their son did*. This approach promoted consistency and thoroughness in the entire data annotation process.

The inter-annotator agreement between the two annotators was calculated based on Cohen's kappa and resulted in a κ coefficient of 91.3 on all datasets jointly, which corresponds to an almost perfect agreement. Per dataset the κ scores are English 89.3, German 91.5, OCD 94.4, Rap 95.6, and Held-Out 93.9. The few disagreements were resolved in subsequent joint discussions and the published datasets correspond to a gold standard annotation.

4 Method

This section first explains the selection of the pre-trained language model and details its fine-tuning procedure to obtain different versions of the model, testing their ability to generalize to code-mixed data and other domains, including zero-shot domain transfer. We also performed comparative zero-shot experiments with LLMs. Furthermore, it details the final hyperparameter settings after optimization. The experiments and their objectives are presented as well as the final evaluation with standard metrics in hate speech detection.

4.1 Models

In total, three models were used in this experiment, all building on XLM-R, which was selected for its strong multilingual text encoding, efficient handling of linguistic complexities, and robustness compared to other multilingual language models, such as mBERT. The training objective is designed as a token classification task, where each token in the input sequence is assigned a label as swear word (B-B), continuation of a swear word (B) or not a swear word (n) as exemplified in Table 1. Since XLM-R uses the SentencePiece tokenizer, labels are potentially assigned to subword tokens, which means that the labels need to be matched to the original input words.

All models were fine-tuned on datasets annotated with the word-level hate speech labels. The German hate speech detection model (DFT) was fine-tuned on German data only, while the German-English bilingual model (DEFT) was fine-tuned on the English and German dataset, but both without code-mixing. The Code-Mixed Model (CMM) was fine-tuned on the word-level annotated subset of the German-English code-mixed OCD dataset.

To achieve optimal model performance, hyperparameter optimization was performed using Optuna (Akiba et al., 2019). The optimization process focused on fine-tuning four key hyperparameters, with the goal of balancing the refinement of essential parameters while minimizing unnecessary adjustments to those already well-suited for the task. Batch size, number of epochs, weight decay, and learning rate were selected due to their direct and substantial impact on model performance, sta-

bility, and generalization. Meanwhile, the default AdamW optimizer, paired with the cross-entropy loss function, was employed throughout the optimization process, maintaining consistency with the standard configuration of the XLM-R for Token Classification model. The entire optimization process was conducted within a Jupyter Notebook environment on Google Colaboratory, which dynamically allocates hardware resources, ensuring efficient training.

In the initial optimization phase, a manual trial-and-error approach was used to determine appropriate hyperparameter ranges for each model. This strategy accounted for the slight variations in dataset sizes and the inherently small datasets, striking a balance between minimizing overfitting and promoting robust generalization for each model individually. Empirical observations guided these adjustments, refining the hyperparameter search space and improving the efficiency of the subsequent optimization process with Optuna. This hyperparameter optimization was applied to all three models under identical conditions, with multiple trials per model to identify the best configurations for each model. The hyperparameter ranges and final sets for fine-tuning each model are shown in Table 3. Each model was tested on the in-domain test set as well as on the two held-out test sets.

4.2 Experiments

For a systematic comparison, all three models described in Section 4.1 were evaluated on the two code-mixed test datasets, the held-out rap dataset as well as the multilingual cross-domain held-out dataset. To effectively evaluate the zero-shot domain transfer to code-mixed hate speech, none of the examples from these two datasets are contained in any other dataset.

We are particularly interested in evaluating whether the addition of English to the fine-tuning of a German hate speech detection language model (DEFT) improves its ability to generalize to German-English code-mixed hate speech. Since this is a zero-shot transfer to code-mixing and the German Rap domain, another model trained on German-English code-mixed data (CMM) is used to contrast with the transfer to only the latter domain. In these experiments, the rap dataset is specifically not used in any fine-tuning process to test also the zero-shot domain transfer of the CMM model, which could be expected to perform better than the non-code-mixed DFT or DEFT model. Furthermore, to assess the robustness of the models and their generalization performance across domains, all models were evaluated on the held-out cross-domain multilingual and code-mixed test set.

As a final comparative and baseline analysis, we perform zero-shot experiments on three state-of-the-art Large Language Models (LLMs): GPT-4o (OpenAI, 2023), Mistral Large (Mistral AI, 2024), and DeepSeek-V3 (DeepSeek-AI et al., 2024). All three LLMs were applied to the two test datasets with code-mixed explicit hate speech.

4.3 Evaluation

For comparison of hate speech detection effectiveness on word level, the metrics precision, recall, and F1 score were applied as a common evaluation method in hate speech detection. Since the experiments were designed as token classification task, these metrics were computed on the intersection of the models' predictions with the gold standard sequences labeled as B-B and B.

The decision to exclude exact duplicates from the evaluation, while still including different orthographic variations of the same words, such as *Scheiße* and *Scheisse*, as well as maintaining a high variety of swear and non-swear words, was based on the observation that a small set of swear words accounts for roughly 90% of online profanity occurrences (Wang et al., 2014). By reducing such duplicates, the focus is not skewed toward the frequent repetition of these words, ensuring a more representative evaluation. This approach prevents the models from achieving artificially high scores by repeatedly identifying the exact same words, thereby maintaining consistency in the evaluation process and simplifying comparisons across different models or iterations.

The detection of neologisms is crucial to account for the dynamic nature of language, where new words, including profane words, continuously emerge over time (Würschinger et al., 2016). Consequently, it is essential to test the models' generalizability to recognize not only existing but also novel swear words. In particular, rap lyrics contain a variety of novel swear words, some of which incorporate intra-word code-switching, such as the compound *Muschibattle* (literal translation: pussy battle). To further investigate the models' ability to handle such novel swear words, their prediction was contrasted with a manually extracted list of neologisms evaluated by two annotators from the rap dataset represented in Table 5.

Hyperparameter	DFT	DEFT	CMM
Epochs	3 to 6 (5)	3 to 8 (6)	5 to 15 (14)
Batch Size	4 or 8 (4)	8 or 16 (16)	4 or 8 (8)
Learning Rate Range	1×10^{-6} to 1×10^{-4}	1×10^{-6} to 1×10^{-4}	1×10^{-6} to 1×10^{-4}
Learning Rate Final	(1.29×10^{-5})	(7.50×10^{-5})	(2.12×10^{-5})
Weight Decay Range	1×10^{-6} to 1×10^{-4}	2×10^{-5} to 7×10^{-3}	5×10^{-5} to 5×10^{-3}
Weight Decay Final	(7.29×10^{-5})	(4.97×10^{-3})	(9.26×10^{-4})

Table 3: Hyperparameter ranges and final selection in brackets for each model determined with Optuna

5 Result

In order to determine whether the inclusion of languages involved in the code-mixed hate speech positively impacts the models' performance on zero-shot domain transfer, we first compare the fine-tuning results of the individual models, and then their performance on two held-out test sets on different domains. The three models presented in Section 4.1 were fine-tuned and tested on their hate speech in-domain test datasets with the optimized hyperparameter settings, where the German-only DFT model achieved an F1 score of 82.11%, the German-English DEFT model achieved a 91.53% F1 score, and the code-mixed German-English CMM model obtained 88.89%. These results indicate the models' considerably robust performance in their respective domains.

The achieved test set F1 score by the CMM is considerably higher than the XLM-R-based 58% reported by Salaam et al. (2022), even though the results are not directly comparable since only a subset of the OCD dataset was used and annotated with word-level hate speech labels. This could be indicative of a better performance of token than sentence classification and labeling, even though this assumption requires further evidence with a direct comparison of these two types of task.

Zero-Shot and Cross-Domain Transfer. The results in Table 4 show that the German-English DEFT model achieves the highest performance in both held-out test sets. The rap dataset contains rap lyrics that are not included in any fine-tuning process, which makes it a particularly adequate dataset to evaluate zero-shot transfer to this domain and to code-mixed data. The DEFT model's performance on the rap datasets indicates a positive impact of 10.3 F1 points by adding English to the training process compared to the German-only DFT model. Both of these models were tested on zero-shot transfer to the rap domain as well as to German-English code-mixed hate speech, while the CMM model was already fine-tuned on code-mixed hate speech. Nevertheless, the CMM model is considerably outperformed by the other two models. This is particularly interesting, since the CMM model was expected to at least outperform the German-only DFT model, which, however, achieves 8.73 F1 points more than the CMM model. This could be an indication of the high quality of the proposed English and German hate speech dataset. A higher domain similarity of Reddit comments to rap lyrics as opposed to the Twitter (now X) and Gab posts of OCD could not be directly confirmed by the annotators, even though this topic of domain similarity might warrant further investigation. It should be highlighted that all three models rely on the same multilingual base model XLM-R.

The second held-out cross-domain test dataset, only called Held-Out in Table 4, contained perviously unseen data from the English and German as well as the two code-mixed rap and OCD datasets, which makes it a valuable cross-domain evaluation set. On this dataset, the DEFT model strongly outperformed the other two models by 15.98 F1 points more than the DFT and 19.16 F1 points more than the CMM model, clearly indicating a strong cross-domain performance. This further supports the assumption of the high quality training data and the positive impact of including the languages involved in the code-mixing in the fine-tuning process on the downstream task, as opposed to monolingual German or code-mixed German-English fine-tuning.

Baseline zero-shot experiments with LLMs show that the DEFT model even outperforms the best performing Mistral model and recall tends to be an issue for the tested LLMs. For GPT-4o the main challenge were multi-word sequences, where a substantial number of words that represent no explicit hate speech were annotated as 'B' after an explicit swear word marked with 'B-B', e.g. *motherfucker (B-B) wollen groß verdienen* (motherfucker want to earn

big) was labeled as [‘B-B’, ‘B’, ‘B’, ‘B’]. In fact, multi-word hate speech sequences were an issue for all three LLMs, since Mistral and DeepSeek assign hardly any ‘B’ labels. However, Mistral correctly identified more bad words than the other two models on both datasets, but DeepSeek obtained the highest precision possible on the rap dataset. The results indicate that the single domain rap dataset was labeled more successfully by LLMs than the cross-domain held-out dataset. All XLM-R-based models across both datasets show higher recall than precision results, indicating that they perform better on identifying hate speech sequences than accurately differentiating them from other sequences. This difference between the two metrics is lower for the cross-domain held-out dataset than for the rap dataset, which contains more creative uses of code-mixed hate speech and neologisms.

The distribution of gold standard categories for deduplicated swear words in the rap dataset compared to missed swear words in the models’ predictions is depicted in Figure 1. It shows that ethnicity, gender, and sex are the most challenging categories for the CMM model, with both ethnic slurs not accurately labeled, 8 out of 19 gender-specific swear words missed, and 10 out of 24 sex-related hate speech sequences incorrectly labeled. Sex and gender seem to be the most challenging categories for all models on this comparatively small dataset. For instance, *blowt* in the sex category as grammatical mixture of the English *blow* and German *blast* as third person verb is inaccurately labeled by the DFT and CMM model, but correctly detected by the DEFT model. From the LLMs, only GPT-4o correctly detects *blowt*. Even etymologically more obscure swear words, such as *punanis*, are detected by the DFT and DEFT model as well as Mistral and GPT-4o, but not by the CMM model. However, the DEFT model does not detect the sex-related *gebumst* (fucked in the sexual sense), while the other two models and GPT-4o accurately assign labels to this word. Additionally, it could be expected that the German-only DFT model does not detect the English *motherfucking*, even though the underlying base model was trained on multilingual data, while the CMM model correctly classifies this example. However, it is curious that the code-mixing fine-tuned multilingual CMM model fails to detect its literal translation *mutterficker*, while the other two models label it correctly. All three LLMs correctly labeled the English and German version of the swear word.

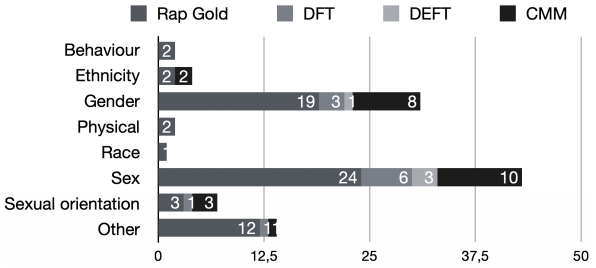

Figure 1: Category distribution of gold and predicted hate speech sequences in the rap domain

Neologism Detection. In order to compare the models’ ability to handle neologisms, their prediction for novel swear words is contrasted with a gold standard list shown in Table 5. Some examples represent intra-compound code-mixing, such as *Bitchbart* (bitch beard), while for others the language identification is less obvious, such as *Disco-Hoes*, since the components of the compound exist in English and in German. For simplicity, all examples with English components in German rap lyrics are marked as code-mixed in Table 5. Furthermore, the novelty of the identified swear words might be debatable, as for some, such as *Disco-Hoes*, entries in urban and slang dictionaries exist. Nevertheless, such sequences are considered novel as they are not easily found in conventional search engines or previous datasets and publications on swear words and/or hate speech. The results show that all models perform reasonably well on detecting such creative uses of hate speech, correctly classifying all but two or three examples. For the LLMs, DeepSeek only correctly detected 5 out of the 13 neologisms, Mistral only missed the two neologisms *Facedrive* and *Kanackenfreestyle*, and GPT-4o only missed *Facedrive*, but labeled many incorrectly as ‘B’. None of the LLMs or models classified *Facedrive* correctly apart from the fine-tuned CMM. This example might refer to the name of a Canadian tech company, however, in its contextual use clearly has a sexual and derogatory meaning. All fine-tuned models failed to identify *Disco-Hoes*, which was found by GPT-4o and Mistral. While a *hoe* might have traditionally referred to gardening equipment, its contextual use with *disco* and the whole sentence is clearly derogatory and refers to promiscuous women. Interestingly, the DFT and DEFT model correctly identify *Hoe* as hate speech, but not in plural and combination with *Disco*. The example *Hurentochter* (daughter of a whore) is an adaptation from *Hurensohn*

	Rap			Held-Out		
Model	Prec	Rec	F1	Prec	Rec	F1
DFT	63.95	83.34	72.37	60.40	71.43	65.45
DEFT	73.81	93.94	**82.67**	76.51	87.02	**81.43**
CMM	55.26	63.64	59.15	57.05	68.55	62.27
GPT-4o	55.70	66.67	60.69	61.11	56.67	58.82
DeepSeek	1.00	45.45	62.50	71.60	38.67	50.22
Mistral	95.83	69.70	80.70	49.56	74.67	59.57

Table 4: Zero-shot evaluation scores of precision (Prec), recall (Rec), and F1 score on two code-mixed test datasets for all three fine-tuned models and three LLMs

Neologism	**DFT**	**DEFT**	**CMM**
Bitchbart	✓	✓	✓
Bitchrap	✓	✓	✓
Bitchtits	✓	✓	✓
Boss-Cock	✓	✓	✓
Broke-Ass-Bitch	✓	✓	✓
Disco-Hoes	✗	✗	✗
Dreckstoys	✓	✓	✓
Facedrive	✗	✗	✓
Hurenschleimer	✓	✓	✓
Hurensohnköpfen	✓	✓	✓
Hurentochter	✓	✓	✗
Kanackenfreestyle	✓	✓	✗
Muschibattle	✓	✓	✓

Table 5: List of neologisms with code-mixed variants highlighted in bold and identification performance of the three models

(son of a whore), which is not correctly classified by the CMM model but all others including the LLMs. The final problematic neologism for the CMM model is *Kanackenfreestyle*, where the first component represents an ethnic slur with an alternative spelling to the more common *Kanaken*. In conclusion, the neologism list represents a short test set for this phenomenon that indicates a strong performance of token classification hate speech detection models on this specific subtask.

6 Discussion

This is, to the best of our knowledge, the first proposal of datasets and approaches to combine word-level hate speech detection and code-mixing. As real-world and (mostly) manually curated datasets, a benchmark for word-level hate speech detection is provided that also considers code-mixing and neologisms. Rap lyrics are particularly prone to novel profanities as well as code-mixing within words, phrases and compounds, making them an especially interesting testbed for hate speech detection and zero-shot domain and language transfer.

As can be seen from the results, the inclusion of the languages involved in the code-mixing improves the overall performance of the token classifiers. Adding English data to fine-tuning XLM-R on word-level hate speech detection, improved the overall performance on two unseen code-mixed test datasets, one of them even considering several domains. In contrast to the best performing model trained on code-mixed data in Jagdale et al. (2024), the German-English token classifier outperformed an identical base model specifically trained on German-English code-mixed data in the presented experiments. This indicates a strong zero-shot domain transfer and a better performance when fine-tuning the same base model on separate English and German examples than on code-mixed examples. However, two performance-relevant factors are the quality and the size of the datasets. The code-mixed model was trained on a subset of OCD, which draws example from Twitter and Gab, while the other two models were trained on curated datasets from different online domains. The quality of the latter might indeed be high, however, the size of the former is considerably lower than the other two datasets.

It is particularly interesting that the XLM-R model trained on multilingual hate speech without code-mixing not only performed better than the code-mixing-specific multilingual model, but also better than three state-of-the-art LLMs tested in a zero-shot scenario. While two of the LLMs correctly identified the majority of the novel swear words, their overall detection performance lagged behind the zero-shot performance of the model fine-tuned on German-English hate speech, indicating a positive impact of language-specific fine-tuning on

a code-mixing task.

The code-mixed CMM model in this experiment uses the same base model as Salaam et al. (2022) and a subset of their dataset, since we needed to annotate the dataset on word-level. Even though this smaller subset was used for fine-tuning, the overall F1 scores are better for DE-EN code-mixed hate speech detection than the previous approach, which could be an indication of better hate speech detection when treating the task as token rather than sentence classification.

The results also show that specific categories of hate speech are more challenging for the fine-tuned models than others, especially sex- and gender-related hate speech sequences. In contrast to previous categorizations (e.g. Silva et al., 2021), explicit hate speech sequences related to sexual intercourse and sex organs, categorized as sex in this work, turned out to be prevalent in the rap dataset. Such sex-related instances might relate to gender in the sense of the intended deformation of specific groups, however, are kept separate for a generally more violent note. The second most frequent category relates to gender-specific instances. Hate speech tends to disproportionately target vulnerable groups, such as women, people of color, LGBTQ+ individuals, and other marginalized communities, exacerbating social inequalities. Addressing this issue is essential to mitigating its impact and ensuring a balance between technological advancement and the protection of these groups.

In this approach, the focus is on explicit, code-mixed hate speech, which potentially disregards sequences of implicit hate speech that do not explicitly contain swear words. For instance, *weißer Völkermord is real* (white genocide is real) appears in the context of a reference to the world-wide Asian population and represents an instance of implicit hate speech. Taking implicit hate speech into consideration, especially with a word-level annotation, is an interesting endeavor for future work.

To the best of our knowledge, rap lyrics have not been considered for this task. The recognition of rap lyrics as a domain for code-mixed word-level hate speech detection holds substantial significance, not only for German and English, but also for its potential applicability to other languages for future work. This is exemplified by artists, such as Capital Bra, who incorporate up to 14 different languages in their lyrics, and Olexesh, who integrates 11 (Tikhonov, 2020), highlighting the complex and multilingual nature of this domain. Thus, one future direction would be to consider the multiplicity of languages in such openly available datasets.

7 Conclusion

Hate speech detection has most commonly been treated as a sentence labeling task. Instead, this paper addresses it as a word-level annotation and token classification task, providing cross-domain training and test datasets as well as a test set of novel swear word creations. The most central contribution is the consideration of code-mixed dataset for German-English sequences, which in combination with word-level hate speech detection is a novel task and seeks to consider all marginalized groups in this important topic. As a major finding, the inclusion of languages involved in the code-mixing for fine-tuning a word-level hate speech detection model improves the results as compared to a model trained on code-mixed data. Furthermore, the token classification method improves the results as compared to previous hate speech detection methods on larger datasets with sentence labels. We would, thus, propose word-level hate speech detection as an improved variant of the sentence-level task, especially for detecting code-mixed hate speech. Since multilingualism and code-mixing are increasingly common in online contents, it is high time to consider this aspect in hate speech detection tasks. As regards neologisms, all fine-tuned models perform surprisingly well, only missing two or three novel uses of (code-mixed) swear words.

As a main endeavor for future work, we consider the domain of rap lyrics across languages as a valuable resource for code-mixed hate speech datasets. Given its fast-paced and dynamic nature, it naturally contains intra-sentential, intra-compound, and intra-word code-mixed hate speech sequences. In addition, this approach only considers explicit hate speech and it would be interesting to extend the approach to annotate and include implicit hate speech, especially in combination with code-mixing.

Limitations

While this paper originally combines word-level hate speech annotation with code-mixing, the code-mixed examples are not tagged with the language identifier. This would permit for further and more detailed analyses of the language-specific ratio of code-mixed hate speech. Nevertheless, it should be considered that this assignment of a language tag is not without controversies due to cognates and

loanwords, e.g. *rapper*, used both in English and German.

To the best of our knowledge, this paper presents the first approach to word-level hate speech classification fine-tuning using code-mixed data, marking an initial contribution to this area despite the limited availability of data. The small dataset size presents challenges for fine-tuning consistency and hyperparameter optimization, making the model more sensitive to minor variations in initialization, batch composition, and data ordering. This sensitivity can lead to inconsistencies across runs and potential reproducibility issues, especially in low-resource settings. As a result, further investigating the stability of results and fine-tuning performance in low-resource data scenarios of word-level hate speech classification is an important direction for future research, with the potential to offer additional valuable insights to the findings presented here. Furthermore, the present approach only considers more recent LLMs in a zero-shot scenario, where few-shot and fine-tuning would present interesting further experiments.

Ethical Considerations

The provided datasets and examples contain offensive and upsetting language. Apart from the negative impact of such language on human annotators being confronted with profanities over an extended period of time, the contents might be upsetting to readers. Nevertheless, such datasets provide a promising approach to enhance the automated detection of the dynamics of profane language, since users tend to code-mix and invent new examples, particularly in the proposed domain of rap lyrics.

Furthermore, the specific types of hate speech are not further delimited by their specific type of insult as regards marginalized communities. For instance, all LGBTQ+ instances of hate speech are classified under the same category of sexual orientation or gender. In the future, given the political and general relevance of the topic, it might be better to provide a more fine-grained annotation of hate speech sequences. Already in this proposed paper a previous and commonly used categorization was extended to instances relating to sexual intercourse and sex organs being used in this specific sense, since this represented a predominant category of swear words in rap lyrics. Thus, real-world data and a consideration of all marginalized groups might require a more fine-grained typology of hate speech. The proposed dataset may serve as a benchmark to test hate speech detection approaches on code-mixed and novel offensive sequences, for which we also provide a baseline approach.

A general risk might be that the proposed datasets could be used to create novel swear words, given its word-level annotation. However, since new swear words keep on appearing on social networks, the need of neologism-aware word-level, multilingual and ideally code-mixing-aware language models is higher than the potentially malignant use of the dataset. At times, context tends to be minimal, especially in rap lyrics. However, placing *Facedrive* alongside *69*, *pussy* and *suck* is usually sufficient for a human to catch a general sexual connotation.

A positive potential ethical impact of this work is that it considers word-level code-mixed language data in a domain specifically known for its derogatory, wide-spread and recent use of language. Further assisting and enabling language models to better detect creative uses of offensive language in the very dynamic domain of rap lyrics should be a general concern for research, since this domain reaches a very wide population.

References

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. LinCE: A centralized benchmark for linguistic code-switching evaluation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1803–1813, Marseille, France. European Language Resources Association.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2623–2631. ACM.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A dataset of Hindi-English code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020. Corpus creation for sentiment analysis in code-mixed Tamil-English text. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for*

Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL), pages 202–210, Marseille, France. European Language Resources association.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, and Wangding Zeng. 2024. Deepseek-v3 technical report. *CoRR*, abs/2412.19437.

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.

LK Dhanya and Kannan Balakrishnan. 2024. Integrating hybrid neural networks and domain-specific embeddings for detecting hate content in code mixed social media comments. *Journal of Internet Services and Information Security (JISIS)*, 14(3):316–329.

Areej Alhothali Hind Saleh and Kawthar Moria. 2023. Detection of hate speech using bert and hate speech word embedding with deep model. *Applied Artificial Intelligence*, 37(1):2166719.

Sanne Hoeken, Sina Zarrieß, and Özge Alacam. 2024. Hateful word in context classification. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 172–186, Miami, Florida, USA. Association for Computational Linguistics.

Shruti Jagdale, Omkar Khade, Gauri Takalikar, Mihir Inamdar, and Raviraj Joshi. 2024. On importance of code-mixed embeddings for hate speech identification. *CoRR*, abs/2411.18577.

Oshadhi Liyanage and Krishnakripa Jayakumar. 2021. Hate speech detection in sinhala-english code-mixed language. In *2021 21st International Conference on Advances in ICT for Emerging Regions (ICter)*, pages 225–230. IEEE.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14867–14875.

Mistral AI. 2024. Mistral large.

Sandip Modha, Thomas Mandl, Gautam Kishore Shahi, Hiren Madhu, Shrey Satapara, Tharindu Ranasinghe, and Marcos Zampieri. 2021. Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in english and indo-aryan languages and conversational hate speech. In *Proceedings of the 13th Annual Meeting of the Forum for Information Retrieval Evaluation*, pages 1–3.

Sepideh Mollanorozy, Marc Tanti, and Malvina Nissim. 2023. Cross-lingual transfer learning with Persian. In *Proceedings of the 5th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 89–95, Dubrovnik, Croatia. Association for Computational Linguistics.

Wajid Hassan Moosa and Najiba. 2022. Multi-lingual hatespeech dataset.

Guanyi Mou, Pengyi Ye, and Kyumin Lee. 2020. Swe2: Subword enriched and significant word emphasized framework for hate speech detection. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1145–1154.

Daniel Nkemelu, Harshil Shah, Michael Best, and Irfan Essa. 2022. Tackling hate speech in low-resource languages with context experts. In *Proceedings of the 2022 International Conference on Information and Communication Technologies and Development*, pages 1–11.

Debora Nozza and Dirk Hovy. 2023. The state of profanity obfuscation in natural language processing scientific publications. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3897–3909, Toronto, Canada. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Cesa Salaam, Franck Dernoncourt, Trung Bui, Danda Rawat, and Seunghyun Yoon. 2022. Offensive content detection via synthetic code-switched text. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6617–6624, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Bhavani Shankar, Preethi Jyothi, and Pushpak Bhattacharyya. 2024. In-context mixing (ICM): Code-mixed prompts for multilingual LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4162–4176, Bangkok, Thailand. Association for Computational Linguistics.

Leandro Silva, Mainack Mondal, Denzil Correa, Fabrício Benevenuto, and Ingmar Weber. 2021. Analyzing the targets of hate in online social media. *Proceedings of the International AAAI Conference on Web and Social Media*, 10(1):687–690.

Aleksej Tikhonov. 2020. Multilingualism and Identity: Polish and Russian Influences in German Rap. *Multiethnica: Journal of the Hugo Valentin Centre*, 40:55–66.

Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, and Amit P. Sheth. 2014. Cursing in english on twitter. In *Computer Supported Cooperative Work, CSCW '14, Baltimore, MD, USA, February 15-19, 2014*, pages 415–425. ACM.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. 2018. Inducing a lexicon of abusive words – a feature-based approach. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1046–1056, New Orleans, Louisiana. Association for Computational Linguistics.

Quirin Würschinger, Mohammad Fazleh Elahi, Desislava Zhekova, and Hans-Jörg Schmid. 2016. Using the Web and Social Media as Corpora for Monitoring the Spread of Neologisms. The case of 'rapefugee', 'rapeugee', and 'rapugee'. In *Proceedings of the 10th Web as Corpus Workshop*, pages 35–43, Berlin. Association for Computational Linguistics.

Nicolas Zampieri, Carlos Ramisch, Irina Illina, and Dominique Fohr. 2022. Identification of multiword expressions in tweets for hate speech detection. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 202–210, Marseille, France. European Language Resources Association.