

Detecting Hate Speech and Offensive Content in English and Indo-Aryan Texts

Mohammadmostafa Rostamkhani¹, Sauleh Eetemadi²

¹*Iran University of Science and Technology (IUST University), University of Science and Technology of Iran, University St., Hengam St., Resalat Square, Tehran, Iran*

²*Iran University of Science and Technology (IUST University), University of Science and Technology of Iran, University St., Hengam St., Resalat Square, Tehran, Iran*

Abstract

In this paper, we address hate speech and offensive content detection for English and Indo-Aryan languages. It is a shared task in hate speech detection for Sinhala and Gujarati (Task 1A and 1B [1, 2, 3]), and English hateful span identification in a text already detected as hateful (Task 3 [4, 5, 6]). The study compares multilingual models on translated text against source language-specific fine-tuned models on source text and evaluates DistilBERT and XLM-RoBERTa for hateful span identification. Results show that fine-tuned source language models excel in hate speech detection, especially with ample high-quality source data. Language models with good pre-training data (languages like Sinhala, and English) have superior performance but limited models in languages like Gujarati emphasize XLM-RoBERTa's advantage against source-language models. This shows the importance of good pre-training data which language models are pre-trained on, for superior hate speech detection. Moreover, XLM-RoBERTa surpasses DistilBERT in identifying hateful spans for Task3. In Task 1A, we ranked 6th out of 16 teams, for Task 1B, we stood 13th among 17 teams, for Task 3, our method achieved 5th place in the public leaderboard (on 30% of test data) and ranked 2nd place in the private leaderboard (70% of test data) among 12 teams. For task 1, our team goes by the name "NAVICK," and for task 3, we are identified as "Mohammadmostafa78". In Task 1A, our highest achieved metrics include an Accuracy of 83.24, Precision of 84.03, Recall of 83.24, and an F1-score of 82.90. Turning to Task 1B, our best performance stands at a Precision of 70.38, Recall of 73.64, and an F1-score of 69.46. For Task 3, we attain peak results with a Precision of 48.81, Recall of 55.39, F1-score of 51.89, an impressive Accuracy of 90.09, along with a Public F1-score of 44.17 and a Private F1-score of 51.38.

Keywords

Hate speech, Offensive content detection, English, Indo-Aryan languages, Sinhala, Gujarati, Multilingual models, Translated text, Source language-specific fine-tuned models, DistilBERT, XLM-RoBERTa, Hateful span identification

1. Introduction

The proliferation of hate speech and offensive content on digital platforms has underscored the urgency of developing effective methods for their detection across languages. We can protect people from offensive content and detect offensive parts of content and censor it. Different

Forum for Information Retrieval Evaluation, December 15-18, 2023, India

✉ mo_rostamkhani97@comp.iust.ac.ir (M. Rostamkhani); sauleh@iust.ac.ir (S. Eetemadi)

🌐 <https://www.sauleh.ir/> (S. Eetemadi)

🆔 0009-0007-0529-6831 (M. Rostamkhani); 0000-0003-1376-2023 (S. Eetemadi)

© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).


 CEUR Workshop Proceedings (CEUR-WS.org)



Figure 1: systems architecture

variety of methods have been used for hate speech detection tasks such as traditional classifiers [7, 8, 9, 10], deep learning-based classifiers [11, 12] or the combination of both approaches [13]. There are also some research for investigation on the importance of initial fine-tuning multilingual models on English hate speech and subsequently fine-tuning them with labeled data in the target language [14].

This paper addresses investigating hate speech identification in both low-resource Indo-Aryan languages (Sinhala and Gujarati) and English. The study encompasses three key tasks: classifying tweets as hate/offensive or not (Task 1A and 1B) and detecting hateful spans within sentences (Task 3). To tackle Task 1, we use two approaches: leveraging translation services in combination with multilingual models and utilizing language models fine-tuned on the source languages.

The code and data associated with our research are made openly available to the community for further exploration and validation in this [GitHub Repository](#).

2. Background

For Task 1 our model receives tokenized text as input and generates its corresponding class for hate speech. For task 3 our model receives tokenized text as input and generates corresponding labels for each token as output.

3. System overview

For task 1 [1, 2, 3], we use different models including English-only, multilingual, and source-language models. We just translate training and test data and then use the model to predict its label.

For the task of hate speech classification (Task 1A and 1B), our methodology begins with an examination of the label distribution for each class. Task 1A distribution displays a class ratio of approximately 4:3, while Task 1B presents a balanced 1:1 ratio. While a minor class imbalance exists in the dataset for task 1A, the adoption of oversampling or undersampling techniques is deemed unnecessary. For this task, we compare two different strategies. The first, uses a translation-based technique, employing the Google Translate API to convert content in the source language into English, subsequently subjecting translations to multilingual models (as an English text), An English-only model which pre-trained on hate speech corpus. the second approach uses models fine-tuned on the source languages.

For Task 3 [4, 5, 6], which pertains to identifying hateful spans in English sentences, we use the BIO notation for sequence labeling. For labeling words with more than one token, we assign

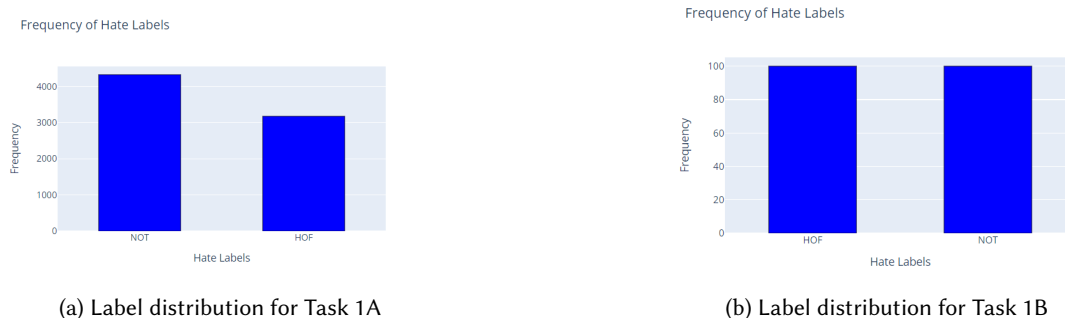


Figure 2: Label distribution

the label of the word to the first token and ignore the remaining tokens. This approach prevents model bias towards lengthy words that have more than one token. The remaining tokens of a word except the first one, are assigned a value of -100, for ignoring them in the loss function and mitigating their impact. we use DistilBERT [15] and XLM-RoBERTa [16] Base models for this task and compare their results. In our system overview for task 3, we outline the key steps and components of our approach for hate span detection.

3.1. Tokenization

We start by tokenizing the input text, and breaking it down into individual tokens.

3.2. Label Assignment for Multi-Token Words

To handle words consisting of more than one token, we adopt a strategy where we assign the label of the entire word to its initial token. Any subsequent tokens from the same word are assigned a special value of -100, effectively excluding them from the loss function during training to mitigate their impact.

3.3. Model Selection

We employ two distinct transformer-based models: DistilBERT and XLM-RoBERTa , to explore their performance in hate span detection.

3.4. Data Split

We use HASOC Task3 dataset [17] as our main dataset. For validation purposes, we partition 10% of our dataset, reserving the remaining 90% for model training.

3.5. Performance Metrics

Throughout the training process, we monitor and report key evaluation metrics for each epoch, including precision, recall, F1 score, and accuracy. These metrics help us assess the model's progress and effectiveness.

3.6. Training Procedure

We utilize the Hugging Face Trainer, a powerful training framework, to facilitate the training and evaluation of our models. This framework streamlines the training process and provides insights into model performance.

3.7. Model Selection Based on Validation Loss

At the conclusion of training, we select the model with the lowest validation loss as our final model. This ensures that we choose the model configuration that optimizes performance.

3.8. Test Data Preparation

Prior to making predictions on the test dataset, we apply the same preprocessing steps as used during training to ensure consistency and fairness in our evaluation.

3.9. Prediction Generation

We employ the selected model to generate predictions for our test data, detecting hateful spans within the test sentences.

4. Experimental setup

In our experimental setup, we carefully configured the parameters to train and evaluate our hate span detection models. The following details provide insight into our setup:

4.1. Data Split

We partitioned our dataset into two subsets: 90% for training and 10% for validation. This allowed us to train our models on a substantial portion of the data while reserving a separate subset for assessing their performance.

4.2. Training Configuration

We conducted training for a total of 5 epochs. Through experimentation, we determined that the optimal model performance was achieved at epoch 2, which we identified as the "best epoch."

4.3. Learning Rate

The learning rate used during training was set to $2e-5$. This rate helps govern the step size taken during the optimization process, influencing the model's convergence and performance.

4.4. Weight Decay

To control overfitting and fine-tune model parameters, we applied a weight decay of 0.01. This regularization technique helps prevent excessive parameter updates during training.

4.5. Batch Sizes

During training, we utilized a batch size of 16 for both the training and evaluation phases. Batch sizes influence the efficiency of the training process and can impact memory usage.

Hyperparameter	Value
Train-Test Split	90% - 10%
Max Epoch	5
Best Epoch	2
Learning Rate	2×10^{-5}
Weight Decay	0.01
Batch Size	16

Table 1
Hyperparameter Settings

By carefully configuring these parameters and splitting our data into training and validation sets, we aimed to ensure a robust and well-tuned training process, ultimately leading to the selection of the best-performing model for hate span detection.

5. Results

In Task 1A, experiments demonstrated that utilizing models fine-tuned on the source languages outperformed the translation-based approach. This can be attributed to the preservation of linguistic nuances and contextual understanding inherent in language-specific models, as well as the absence of a proficient language model for correct and accurate translations also some issues present within the translated sentences. For example, "@USER" in some translations changed and in some others did not. To prevent this issue, we could remove "@USER" totally. The fine-tuned models on source language have higher precision and recall and F1-score in all cases for identifying hate speech and offensive content.

Language	Model	Loss	Test Accuracy	Test Precision	Test Recall	Test F1
En- glish	distilbert-base-uncased	0.6252	0.6408	0.6231	0.6132	0.6144
	twitter-xlm-roberta-base-sentiment	0.5484	0.6416	0.6282	0.6278	0.6280
	roberta-hate-speech-dynabench-r4-target	0.6317	0.6568	0.6422	0.6223	0.6228
Sin- hala	SinhalaBERTo	0.4169	0.8228	0.8216	0.8075	0.8126
	xlm-t-hasoc-hi	0.4199	0.8244	0.8267	0.8244	0.8208
	xlm-t-hasoc-hi-sold-si	0.4116	0.8324	0.8403	0.8324	0.8272
	xlm-t-sold-si	0.4284	0.8316	0.8326	0.8316	0.8290

Table 2
Results of Task 1A (hate speech detection for Sinhala)

In the evaluation of hateful span detection, we assessed both DistilBERT and XLM-RoBERTa models. The outcomes highlighted XLM-RoBERTa's effectiveness in identifying hateful spans within sentences, attaining superior F1 scores. This underscores the significance of harnessing pre-trained models explicitly engineered for cross-lingual and contextual comprehension tasks.

Language	Model	Precision	Recall	F1-Score
Gujarati	gujarati-bert	0.6917	0.6835	0.6870
	Gujarati-Model	0.3383	0.3843	0.3577
	Gujarati-XLM-R-Base	0.3428	0.5000	0.4067
English	twitter-xlm-roberta-base-sentiment	0.7038	0.7364	0.6946

Table 3
Results of Task 1B (hate speech detection for Gujarati)

Model	Val Loss	Val Precision	Val Recall	Val F1-Score	Val Accuracy	Public F1	Private F1
distilbert-base-uncased	0.3184	0.4712	0.5221	0.4953	0.8906	0.4389	0.4941
xlm-roberta-base	0.2877	0.4881	0.5539	0.5189	0.9003	0.4417	0.5138

Table 4
Results of Task 3 (hate span identification)

The organizers have implemented two sets of metrics and leaderboards: the public leaderboard, which evaluates performance using roughly 30% of the test data, and the private leaderboard, which utilizes approximately 70% of the test data for evaluation. On the public leaderboard, we achieved a ranking of 5th, whereas on the private leaderboard, we secured the 2nd position.

6. Conclusion

This paper investigates hate speech detection in English and Indo-Aryan languages, showcasing results from Sinhala and Gujarati tasks (Task 1A and 1B), as well as English hateful span identification (Task 3). Comparing translation-based multilingual models and language-specific fine-tuned models, it evaluates DistilBERT and XLM-RoBERTa for hateful span identification. Fine-tuned source language models excel in hate speech detection, particularly with ample high-quality source data, benefiting languages like Sinhala. The scarcity of models in languages like Gujarati highlights XLM-RoBERTa’s advantage. This underscores tailored data and language models’ significance. Moreover, XLM-RoBERTa outperforms DistilBERT in identifying hateful spans, accentuating language-specific models’ importance in advancing cross-lingual processing.

References

- [1] T. Ranasinghe, I. Anuradha, D. Premasiri, K. Silva, H. Hettiarachchi, L. Uyangodage, M. Zampieri, Sold: Sinhala offensive language dataset, arXiv preprint arXiv:2212.00851 (2022).
- [2] S. Satapara, H. Madhu, T. Ranasinghe, A. E. Dmonte, M. Zampieri, P. Pandya, N. Shah, M. Sandip, P. Majumder, T. Mandl, Overview of the hasoc subtrack at fire 2023: Hate-speech identification in sinhala and gujarati, in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, Goa, India. December 15-18, 2023, CEUR Workshop Proceedings, CEUR-WS.org, 2023.

- [3] T. Ranasinghe, K. Ghosh, A. S. Pal, A. Senapati, A. E. Dmonte, M. Zampieri, S. Modha, S. Satapara, Overview of the HASOC subtracks at FIRE 2023: Hate speech and offensive content identification in assamese, bengali, bodo, gujarati and sinhala, in: Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Goa, India. December 15-18, 2023, ACM, 2023.
- [4] S. Masud, M. A. Khan, M. S. Akhtar, T. Chakraborty, Overview of the HASOC Subtrack at FIRE 2023: Identification of Tokens Contributing to Explicit Hate in English by Span Detection, in: Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, CEUR, 2023.
- [5] K. Ghosh, A. Senapati, A. S. Pal, Annihilate Hates (Task 4, HASOC 2023): Hate Speech Detection in Assamese, Bengali, and Bodo languages, in: Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, CEUR, 2023.
- [6] S. Satapara, S. Masud, H. Madhu, M. A. Khan, M. S. Akhtar, T. Chakraborty, S. Modha, T. Mandl, Overview of the HASOC subtracks at FIRE 2023: Detection of hate spans and conversational hate-speech, in: Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Goa, India. December 15-18, 2023, ACM, 2023.
- [7] M. W. M. Thomas Davidson, Dana Warmesley, I. Weber, Automated hate speech detection and the problem of offensive language, CoRR abs/1703.04009 (2017). URL: <http://arxiv.org/abs/1703.04009>.
- [8] Z. Waseem, D. Hovy, Hateful symbols or hateful people? predictive features for hate speech detection on Twitter, in: Proceedings of the NAACL Student Research Workshop, Association for Computational Linguistics, San Diego, California, 2016, pp. 88–93. URL: <https://aclanthology.org/N16-2013>. doi:10.18653/v1/N16-2013.
- [9] Y. H. R. Y. E. R. K. G. N. MacAvaney, S., O. Frieder, Hate speech detection: Challenges and solutions, PloS one (2019).
- [10] E. W. Pamungkas, V. Basile, V. Patti, Misogyny detection in twitter: a multilingual and cross-domain study, Information Processing Management 57 (2020) 102360. URL: <https://www.sciencedirect.com/science/article/pii/S0306457320308554>. doi:https://doi.org/10.1016/j.ipm.2020.102360.
- [11] S. Agrawal, A. Awekar, Deep learning for detecting cyberbullying across multiple social media platforms, CoRR abs/1801.06482 (2018). URL: <http://arxiv.org/abs/1801.06482>. arXiv:1801.06482.
- [12] P. Badjatiya, S. Gupta, M. Gupta, V. Varma, Deep learning for hate speech detection in tweets, CoRR abs/1706.00188 (2017). URL: <http://arxiv.org/abs/1706.00188>. arXiv:1706.00188.
- [13] Z. Mossie, J.-H. Wang, Vulnerable community identification using hate speech detection on social media, Information Processing Management 57 (2020) 102087. URL: <https://www.sciencedirect.com/science/article/pii/S0306457318310902>. doi:https://doi.org/10.1016/j.ipm.2019.102087.
- [14] P. Röttger, D. Nozza, F. Bianchi, D. Hovy, Data-efficient strategies for expanding hate speech detection into under-resourced languages, 2022. arXiv:2210.11359.
- [15] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter, CoRR abs/1910.01108 (2019). URL: <http://arxiv.org/abs/1910>.

01108. arXiv:1910.01108.

- [16] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, CoRR abs/1911.02116 (2019). URL: <http://arxiv.org/abs/1911.02116>. arXiv:1911.02116.
- [17] S. Masud, M. Bedi, M. A. Khan, M. S. Akhtar, T. Chakraborty, Proactively reducing the hate intensity of online posts via hate speech normalization, in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 3524–3534. URL: <https://doi.org/10.1145/3534678.3539161>. doi:10.1145/3534678.3539161.