DOSA: Dravidian Code-Mixed Offensive Span Identification Dataset

Manikandan Ravikiran*
OMSCS Program
Bangalore, Karnataka, India
mravikiran3@gatech.edu

Subbiah Annamalai*
OMSCS Program
Columbus, Ohio, USA
sannamali@gatech.edu

Abstract

This paper presents the Dravidian Offensive Span Identification Dataset (DOSA) for under-resourced Tamil-English and Kannada-English code-mixed text. The dataset addresses the lack of code-mixed datasets with annotated offensive spans by extending annotations of existing code-mixed offensive language identification datasets. It provides span annotations for Tamil-English and Kannada-English code-mixed comments posted by users on YouTube social media. Overall the dataset consists of 4786 Tamil-English comments with 6202 annotated spans and 1097 Kannada-English comments with 1641 annotated spans, each annotated by two different annotators. We further present some of our baseline experimental results on the developed dataset, thereby eliciting research in under-resourced languages, leading to an essential step towards semi-automated content moderation in Dravidian languages. The dataset is available in https://github.com/manikandan-ravikiran/DOSA.

1 Introduction

Fighting offensive content is imperative for social media companies and other entities involved in content moderation. Currently, much of moderation is relatively limited on most community platforms (Jhaver et al., 2019) with most of them relying on detection of repeatedly used words[1] and block-lists (Jhaver et al., 2018). Additionally, most social media companies employ human content moderators, who are frequently swamped by offensive mentions and their volume (Arsht and Etcovitch, 2018). On the other hand, precise moderation leads to content delay leading to user attrition. Furthermore, smaller entities cannot utilize human moderators on a large scale due to their sheer cost. As a result, they shut down their comments sections entirely. Although content moderation to some degree has utilized semi-automated approaches (Jhaver et al., 2019), most of them are not yet available for Non-English languages and code-mixed texts.

Code-switching or code-mixing is a mixing of linguistic units from two or more languages in a single conversation or sometimes even a single utterance and is widely used across the world (Sitaram et al., 2019). In India, due to widely employed educational and cultural guidelines, English largely influences all the Indian spoken languages, including Dravidian languages like Kannada and Tamil (Chakravarthi et al., 2020). However, with the advent of social media, code-switching has permeated to mediums with informal contexts like forums and messaging platforms. As a result, code-switching is part and parcel of offensive conversations in social media.

Despite many recent NLP advancements, handling code-mixed offensive content is still a challenge in Dravidian Languages (Sitaram et al., 2019). The primary reason is data scarcity, as it appears relatively less in standard textual resources and instead spread across the World Wide Web. However, recently the research of offensive code-mixed texts in Dravidian languages has seen traction (Chakravarthi et al., 2020; Hande et al., 2020). However, these are restricted to the whole comment's classification for offensiveness and do not identify the spans that make a text offensive. But emphasizing such offensive spans can assist human moderators who often deal with lengthy comments and prefer attribution instead of just a system-generated unexplained score per post. Accordingly, the contributions of this paper are as follows

- We first present DOSA, a code-mixed Tamil-English, and Kannada-English dataset annotated for offensive spans. We describe our

*Equal Contribution
[1] https://www.reddit.com/wiki/automoderator

Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pages 10–17
April 20, 2021 ©2021 Association for Computational Linguistics

annotation scheme in the due process and examine the dataset properties, and brief about annotator-related information[2].

- We also provide an experimental baseline over state-of-the-art multilingual language models of BERT (Devlin et al., 2019), DistillBERT (Sanh et al., 2019), and XLM-RoBERTA (Conneau et al., 2020) on the developed offensive span identification dataset.

The rest of the paper is organized as follows. In section 2, we discuss literature on offensive language and span identification. Following this in section 3, we present the dataset collection and annotation process. Section 4 offers the experimental setting used for the baseline creation. In section 5, we discuss our results and errors so identified. Finally, in section 6, we conclude with a summary and possible directions for future work.

2 Related Work

2.1 Offensive language & Span Identification:

Offensive language identification (OLID) problem is widely investigated in the literature via multiple facets of works ranging from hierarchical OLID annotation scheme (Zampieri et al., 2019a,b), release of large-scale semi-supervised training dataset with over nine million English tweets (Rosenthal et al., 2020), extension of OLID to languages of Arabic (Mubarak et al., 2020), Danish (Sigurbergsson and Derczynski, 2020), Greek (Pitenis et al., 2020), and Turkish (Çöltekin, 2020) and development of multiple systems (Zampieri et al., 2020; Ravikiran et al., 2020). In parallel, there are more course-grained works on Hate Speech Identification (Kumar et al., 2018), Aggressiveness Detection (Aroyehun and Gelbukh, 2018), Bullying Detection (Xu et al., 2012) etc. In this work, we restrict ourselves to offensive comments only[3]. Unlike OLID, span identification is still in the nascent stage. To the best of our knowledge work by Pavlopoulos et al. 2021 which introduces a toxic span dataset and shared task with 10k comments is the only work in this line. Our work extends span identification to Youtube comments in code-mixed Dravidian languages.

2.2 Code-Mixing in Offensive Language and Span Identification:

Offensive language identification with code mixed texts have seen most works in Hindi-English (Srivastava et al., 2020; Bohra et al., 2018; Santosh and Aravind, 2019; Rajput et al., 2020; Chopra et al., 2020). Recently there are works in Bangla (Jahan et al., 2019), Kannada (Hande et al., 2020) and Tamil (Chakravarthi et al., 2020). To the best of our knowledge, there are no works on span identification with Dravidian code-mixed datasets. Our work addresses this gap, by emphasizing the creation of code-mixed offensive span identification inline with Pavlopoulos et al. 2021.

3 Dataset Collection and Annotation

In this work, we reuse TamilMixSentiment (Chakravarthi et al., 2020), and KanCMD (Hande et al., 2020) datasets consisting of 15k and 7k YouTube code-mixed comments respectively in Tamil-English and Kannada-English languages. Reusing the existing dataset is beneficial. It encourages the development of multitask models with span identification as one of the tasks, analysis of model interpretability during offensive language identification, and developing a unified benchmark dataset for multiple NLP tasks in code-mixed Dravidian languages.

In this work, we considered only a subset of the comments that were already annotated as offensive[4] for our span annotation process. Out of this subset, we rechecked and removed non-code mixed comments resulting in 9049 and 1311 comments in Tamil-English and Kannada-English, respectively. For the final annotation process, we considered all of the code-mixed Kannada-English comments and 5000 Tamil-English comments.

3.1 Annotation Setup

For annotation, we follow earlier works on span identification (Pavlopoulos et al., 2021) where two annotators annotated every comment according to the guidelines from section 3.2.

[2] **Disclaimer**: This paper contains examples that may be considered profane, vulgar, or offensive. These contents do not reflect the authors' views or the graduate schools/employed organization with which they are associated and exclusively serve to explain linguistic research challenges.

[3] The relationship between offensiveness, Hate-speech, agreesiveness etc. is presented in https://link.springer.com/article/10.1007/s10579-020-09502-8

[4] Released as part of https://competitions.codalab.org/competitions/27654

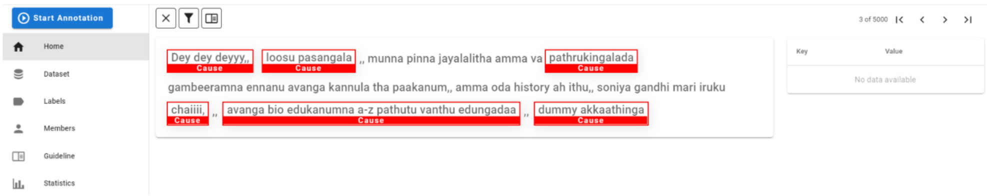

Figure 1: Annotation of offensive spans using Doccano.

Since the original comments are from the public domain, we anonymized all the personal information and user-related tags to protect actual users' privacy during our annotation process. Besides, no personal information of annotators was collected except their educational background and expertise in the language they volunteered to annotate.

Moreover, all the annotators were informed that the contents to be annotated are profane, vulgar, or offensive and can withdraw from the annotation process if necessary. For annotation, we use doccano (Nakayama et al., 2018) which was locally hosted by each individual annotator, and the annotations were finally merged separately once all annotations were obtained. Within doccano, all the annotators were explicitly asked to create a single label called **CAUSE** with label id of 1, thus maintaining consistency of annotation labels. (See Figure 1)

3.2 Annotation Guidelines

The annotators have explained the meaning of offensiveness with illustrative examples. Annotators who agreed that they understood this were given the following instructions:

- Extract the offensive word sequences (spans) of the comment by highlighting each such span and labeling them as **CAUSE** as shown in Figure 1.

- If the comment is not offensive or if the offensiveness is context-dependent, do not highlight any span.

- If the whole comment should be annotated, then annotate the whole comment and convey the annotation verifier about the same after completion.

3.3 Annotators

To start with, we selected a total of 15 annotators, all of whom had minimal education of Bachelors Degree with either medium of schooling to be one of the English, Tamil, and Kannada languages or proficient in both speaking and writing of one or both the Dravidian languages. Further, the annotation was done iteratively in a cycle of 500 sentences where each of the annotators was asked to report back to verify the quality of annotations and receive their next batch of 500 sentences. Each batch was manually verified by an annotation verifier, which allowed us to control the quality of annotations. This, in turn, permitted us to remove annotators who did not annotate well or had a significant delay in annotations. At the end of this process, we had six annotators, out of which all of the annotators were native speakers or writers of either Kannada or Tamil or both. Also, two of the annotators acted as annotation verifiers. Table 1 shows details of the annotators with educational qualification, gender diversity, Medium of instruction in schooling, miscellaneous qualities, including knowledge of multiple accents of Kannada/Tamil. Each YouTube comment was initially sent to two annotators for span annotation without revealing that the comment was offensive. If there was a disagreement in annotation, then the comment was sent to the third annotator. If all the three disagreed, then we skipped the annotation of that particular comment. Overall this leads to the annotation of each comment by two annotators.

Annotator Identity	**Educational Background**	**Medium of Schooling**	**Bilingual**	**Accent Knowledge**
1	Masters ♂☼	English	✗	✗
2	Masters ♂☼	English	✓	✗
3	Master (Tamil) ♀	Tamil	✗	✓
4	Bachelors ♀	Kannada	✓	✗
5	Masters (Kannada) ♂	Kannada	✓	✓
6	Masters ♂	English	✗	✗

Table 1: Annotators and their characteristics. ☼ indicates annotation verifiers.

3.4 Ground Truth Creation

For ground truth creation, we follow a strategy in line with works of Pavlopoulos et al. 2021 where for each comment, we obtain character offset of the identified span using doccano. We then retained only the overlapping annotations, i.e., both annotators must have included each character offset in their spans for the offset to be included in the ground truth. The annotation verifiers resolved any discrepancy in considering the non-overlapping part of the annotations.

3.5 Corpus Statistics

Language-Pair	Tamil-English	Kannada-English
Number of Sentences	4786	1097
Number of unique tokens	22096	7781
Number of annotated spans	6202	1641
Average size of spans (# of characaters)	21	20
Min size of spans (# of characaters)	4	2
Max size of spans (# of characaters)	82	160
Number of unique tokens in spans	10737	3742

Table 2: DOSA corpus statistics

Corpus statistics is given in the Table 2. Compared to Tamil-English, we can see Kannada-English has a significantly lesser number of samples. This is because of the inherent nature of the KanCMD dataset (Hande et al., 2020) which consists of only 1472 comments annotated as offensive. While the dataset is minimal, we release this along with Tamil-English to empower more annotation and subsequently build better offensive span identification models for the Kannada-English language. Moreover, we can see that the maximum size of the annotation is 82 and 102, respectively, across the datasets, but it can be seen from Figure 2 and 3 that these have very few occurrences.

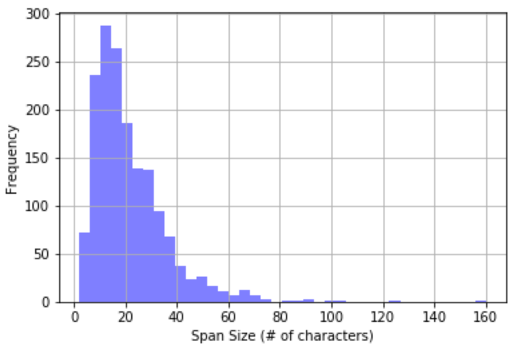

Figure 2: Histogram of annotated Span size in Kannada-English dataset.

3.6 Inter annotator Agreement

Since two annotators annotated each sentence, and the focus is only on the offensive contents, the annotation quality is validated using Cohens Kappa on annotated tokens only. In our case, we saw this value to be 0.6.

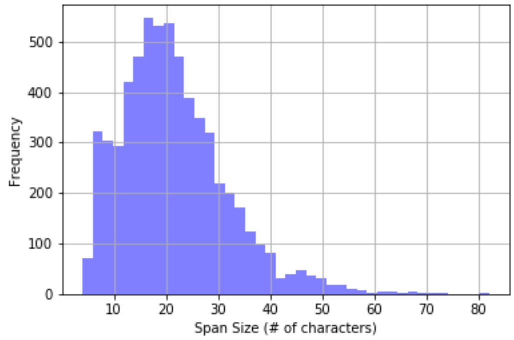

Figure 3: Histogram of annotated Span size in Tamil-English dataset.

4 Experimental Settings

To establish a baseline performance, we applied multiple state-of-the-art multilingual language models to determine the span of offensive comments. In this section, we present various models, hyperparameters, and other experimental settings used as part of the baseline estimation.

4.1 Models

Since the task focuses on identifying spans of offensive word sequences, we treat the problem of identification of span as the task of sequence labeling where we tag words that contribute to offensiveness. In this work, we use the following language models available through HuggingFace's Transformer Library (Wolf et al., 2019).

- **Mulitlingual BERT:** Multilingual BERT (M-BERT) is a language model pre-trained from monolingual corpora in 104 languages where task-specific annotations in one language are used to finetune the model for evaluation in another language. We use two variants of BERT, namely **BERT-M1**[5] which is trained on Wikipedia corpus and **BERT-M2**[6] which is original BERT finetuned first on XQUAD and Tydi QA dataset.
- **Mulitlingual DistilBERT:** We also use a smaller general-purpose language representation model, DistilBERT, which upon finetuning offers better performance on downstream tasks. Again we use two variants of DistilBERT namely DBERT-M1 [7] which is the original model developed as part of (Sanh et al.,

[5] bert-base-multilingual-uncased
[6] bert-multi-cased-finedtuned-xquad-tydiqa-goldp
[7] distilbert-base-multilingual-cased

2019) and **DBERT-M2** [8] which is similar to the earlier case of BERT where the model is again finetuned on XQUAD and Tydi QA dataset.

- **Mulitlingual XML-RoBERTA:** This is a masked language model trained on a multilingual language modeling objective using only monolingual data. Here again we use two variants namely **XBERT-M1**[9] and **XBERT-M2** [10] with former being the base model released as part of (Conneau et al., 2020) and later being a larger model which is finetuned on multiple NLI datasets.

4.2 Hyperparameters

For our experiments, we trained all of our models under a common setting. The various parameter setting is as shown in Table 3. Considering the effect of the presence of specific offensive terms and the size of the overall dataset, rather than creating a random train-test split in this work, we employed 3-fold cross-validation for all the experiments.

Parameters	Values
Learning Rate	4x10^{-5}
Maximum Sequence Length	128
Batch Size	16
Epochs	100
Weight Decay	0.01
Adam ϵ	1x10^{-8}

Table 3: Hyperparameters used across experiments.

5 Experiments, Results, and Discussion

The experimental results for various state-of-the-art multilingual language models are as shown in Tables 4-9. Since the focus of these experiments is to just establish baselines and provide some starting pointers for further exploration, we restrict ourselves from in-depth error analysis and instead focus on unique errors which we came across during the experiments. To start with, we compute results for each of the fold where we identify span/entity level Precision (P), Recall (R), and F1-Score (F1) inline with past works (Wang and Iwaihara, 2019; Yamada et al., 2020). Computing entity level P, R, and F1 measures consider only those word sequences which precisely match the annotation, thus eliminating partially identified offensive spans. This measure is also in line with Pavlopoulos et al. 2021.

[8] distilbert-multi-finetuned-for-xqua-on-tydiqa

[9] xlm-roberta-base

[10] xlm-roberta-large-xnli-anli

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
BERT-M1	1	0.369	0.387	0.377	0.374	0.397	0.385
	2	0.381	0.432	0.406	0.309	0.356	0.331
	3	0.397	0.419	0.408	0.400	0.416	0.408
	Average	**0.382**	**0.413**	**0.397**	**0.361**	**0.390**	**0.375**

Table 4: Results of BERT-M1 for offensive span identification.

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
BERT-M2	1	0.394	0.394	0.394	0.382	0.391	0.387
	2	0.397	0.441	0.418	0.349	0.397	0.372
	3	0.386	0.408	0.396	0.387	0.406	0.396
	Average	**0.392**	**0.414**	**0.403**	**0.373**	**0.398**	**0.385**

Table 5: Results of BERT-M2 for offensive span identification.

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
DBERT-M1	1	0.380	0.412	0.395	0.408	0.420	0.414
	2	0.349	0.364	0.356	0.363	0.417	0.389
	3	0.413	0.417	0.415	0.393	0.436	0.414
	Average	**0.381**	**0.398**	**0.389**	**0.388**	**0.425**	**0.405**

Table 6: Results of DBERT-M1 for offensive span identification.

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
DBERT-M2	1	0.372	0.391	0.381	0.378	0.387	0.382
	2	0.295	0.365	0.328	0.382	0.440	0.409
	3	0.370	0.378	0.374	0.396	0.434	0.414
	Average	**0.346**	**0.378**	**0.361**	**0.385**	**0.420**	**0.402**

Table 7: Results of DBERT-M2 for offensive span identification.

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
XBERT-M1	1	0.405	0.432	0.418	0.379	0.395	0.387
	2	0.364	0.397	0.380	0.395	0.420	0.407
	3	0.407	0.415	0.411	0.374	0.391	0.382
	Average	**0.392**	**0.415**	**0.403**	**0.383**	**0.402**	**0.392**

Table 8: Results of XBERT-M1 for offensive span identification.

Model		Kannada-English			Tamil-English		
	Fold #	P	R	F1	P	R	F1
XBERT-M2	1	0.365	0.381	0.372	0.249	0.308	0.275
	2	0.379	0.438	0.405	0.216	0.254	0.234
	3	0.336	0.408	0.369	0.263	0.317	0.289
	Average	**0.360**	**0.409**	**0.382**	**0.243**	**0.293**	**0.266**

Table 9: Results of XBERT-M2 for offensive span identification.

To start with, for both Tamil-English and Kannada-English code-mixed text, all the models perform poorly with the best average F1 of 0.403 for BERT-M2 and XBERT-M2, respectively for Kannada-English. Meanwhile, for Tamil-English comments, we found the maximum average F1 of 0.405 for DBERT-M1. Besides, across all the folds on each language model, the results are in similar ranges. Such poor performance can be attributed to two reasons: the training process and the complexity of the code-mixed text.

During our experiments, for Kannada-English, we found most models tend to overfit, and in some cases, the model training saturated at a training loss of 0.1. For the former case, we employed precision control of the learning rate to control overfitting; however, the net effect was relatively limited due to the small dataset size. However, the latter case tends to be more challenging to handle even after significant learning rate changes. Additionally, since the experiment was a baseline, we didn't perform any hyperparameter tuning for other parameters, leaving them for future work.

Furthermore, unlike Kannada-English experiments, we found overfitting for Tamil-English only in the case of XLM-RoBERTA models. Moreover, for both Kannada-English and Tamil-English, we found that comments with only one or more profane words causing offensiveness were correctly identified in their spans. However, for both Tamil-English and Kannada-English, we can see overall lower results for which one of the reasons we thought was the nature of code-mixed text themselves. To verify this, we cross-check the errors and found the following issues.

Complete Sentence annotations - For cases where the complete comment or more than 70% of the characters are accounted in offensive span, we found the errors to be highest where one or more words are not tagged as offensive, leading to a drop in span level F1. Example sentences with ground truth and predicted spans are as shown below.

Ground Truth: *Sir intha cinema madida mele dodda mattadalli prachara madbekittu. nNodi prem avru dabba film madudru hype build-up create madodralli etthida kai.*

Prediction: *Sir intha cinema madida mele dodda mattadalli prachara madbekittu. nNodi prem avru dabba film madudru hype build-up create madodralli etthida kai.*

Translation: *Sir, after making this kind of movie, there was need of more publicity. Please see Mr. Prem, even after making useless movie still he is king in creating hype.*

Word Pronunciation - Another unique case of errors involves words that are the same except the texts are written differently. These are again not correctly identified as offensive. In the example below both Devidya and Thevidya translates to *whore*, which is often used as an abuse word.

Example 1: *...Dai unga ammayepdi da unna petha* ***Devidiya*** *intha comments ku.*
Example 2: *Otha* ***Thevidiya*** *Pasangala Neenga Nalla Padam Edukkanumnu Yenda...*

Noisy texts - Occasionally, we also found span errors where the sentence is full of hashtags that were annotated as offensive, but the model only identifies part of them. Examples are shown below.

Example 1: *#BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali #BoycottComali ....*
Example 2: *Disaster.... disaster .... disaster.....Disaster.... disaster .... disaster.....Disaster.... disaster.*

Sarcastic Sentences: Sarcastic comments where the complete sentence is annotated for spans were also cases where the model fails to work. One such example is as shown with ground truth and prediction.

Ground Truth: *puratchi thalavi kamika sona tun tun aunty kamikerenga.*
Prediction: *puratchi thalavi kamika sona tun tun aunty kamikerenga*
Translation: Rather than showing purtachi thalavi (Honorable Nick Name given to former Tamil Nadu CM), why are you showing Tun Tun Aunty (A Cartoon character in Chota Bheem).

6 Conclusion

In this paper, we presented DOSA, a dataset for offensive span identification for Tamil-English and Kannada-English code mixed texts. We also achieved an inter-annotator agreement of 0.6 for the annotations. We also created baselines and reported P, R, and F1 for span identification using the developed dataset and state-of-the-art multilingual language models. In the due process, we presented some of the challenges in the training language model-based baselines, which is a possible future work. Moreover, in this work, we did not present results on simpler models like LSTM-CRF and its variants, which is also a possible exploration. Most importantly, we could see cases where the complete comment was annotated either due to their sarcastic nature or because they had only offensive terms. In this regard, a possible question to explore includes evaluating the need for such annotations by considering larger datasets and improving models' performance under such conditions. Additionally, we think this resource is useful for multitask learning and interpretability for code-mixed offensive language models.

Acknowledgments

We thank our anonymous reviewers for their valuable feedback. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors only and does not reflect the view of their employing organization or graduate schools. The work is the result of continuous study group discussions during and after the CS7646-ML4T course (OMSCS Program, Georgia Tech) done at MLSG - An online machine learning study group formed to discuss interesting papers and open research problems in areas of NLP focus on South Indian Languages. Further, MLSG is a random name generated for the representation purpose of our reading-discussion group. We would also like to thank all of our annotators for their effort in annotation and review.

References

Segun Taofeek Aroyehun and Alexander F. Gelbukh. 2018. Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying, TRAC@COLING 2018, Santa Fe, New Mexico, USA, August 25, 2018*, pages 90–97. Association for Computational Linguistics.

Andrew Arsht and Daniel Etcovitch. 2018. The human cost of online content moderation. *Harvard Journal of Law & Technology*.

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, PEOPLES@NAACL-HTL 2018, New Orleans, Louisiana, USA, June 6, 2018*, pages 36–41. 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 and Collaboration and Computing for Under-Resourced Languages, SLTU/CCURL@LREC 2020, Marseille, France, May 2020*, pages 202–210. European Language Resources association.

Shivang Chopra, Ramit Sawhney, Puneet Mathur, and Rajiv Ratn Shah. 2020. Hindi-english hate speech detection: Author profiling, debiasing, and practical perspectives. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 386–393. AAAI Press.

Çagri Çöltekin. 2020. A corpus of turkish offensive language on social media. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 6174–6184. 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, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Adeep Hande, R. Priyadharshini, and Bharathi Raja Chakravarthi. 2020. Kancmd: Kannada codemixed dataset for sentiment analysis and offensive language detection. In *PEOPLES*.

Maliha Jahan, Istiak Ahamed, Md. Rayanuzzaman Bishwas, and Swakkhar Shatabda. 2019. Abusive comments detection in bangla-english code-mixed and transliterated text. *2019 2nd International Conference on Innovation in Engineering and Technology (ICIET)*, pages 1–6.

Shagun Jhaver, Iris Birman, Eric Gilbert, and Amy S. Bruckman. 2019. Human-machine collaboration for content regulation: The case of reddit automoderator. *ACM Trans. Comput. Hum. Interact.*, 26(5):31:1–31:35.

Shagun Jhaver, Sucheta Ghoshal, Amy S. Bruckman, and Eric Gilbert. 2018. Online harassment and content moderation: The case of blocklists. *ACM Trans. Comput. Hum. Interact.*, 25(2):12:1–12:33.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying, TRAC@COLING 2018, Santa Fe, New*

Mexico, USA, August 25, 2018, pages 1–11. Association for Computational Linguistics.

Hamdy Mubarak, Ammar Rashed, Kareem Darwish, Younes Samih, and Ahmed Abdelali. 2020. Arabic offensive language on twitter: Analysis and experiments. *CoRR*, abs/2004.02192.

Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. 2018. doccano: Text annotation tool for human. Software available from `https://github.com/doccano/doccano`.

John Pavlopoulos, Lo Laugier, Jeffrey Sorensen, and Ion Androutsopoulos. 2021. Semeval-2021 task 5: Toxic spans detection (to appear). In *Proceedings of the 15th International Workshop on Semantic Evaluation*.

Zeses Pitenis, Marcos Zampieri, and Tharindu Ranasinghe. 2020. Offensive language identification in greek. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 5113–5119. European Language Resources Association.

Kshitij Rajput, Raghav Kapoor, Puneet Mathur, Hitkul, Ponnurangam Kumaraguru, and Rajiv Ratn Shah. 2020. *Transfer Learning for Detecting Hateful Sentiments in Code Switched Language*, pages 159–192. Springer Singapore, Singapore.

Manikandan Ravikiran, Amin Ekant Muljibhai, Toshinori Miyoshi, Hiroaki Ozaki, Yuta Koreeda, and Sakata Masayuki. 2020. Hitachi at semeval-2020 task 12: Offensive language identification with noisy labels using statistical sampling and post-processing. pages 1961–1967.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2020. A large-scale semi-supervised dataset for offensive language identification. *CoRR*, abs/2004.14454.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

T. Y. S. S. Santosh and K. V. S. Aravind. 2019. Hate speech detection in hindi-english code-mixed social media text. pages 310–313.

Gudbjartur Ingi Sigurbergsson and Leon Derczynski. 2020. Offensive language and hate speech detection for danish. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 3498–3508. European Language Resources Association.

Sunayana Sitaram, Khyathi Raghavi Chandu, Sai Krishna Rallabandi, and A. Black. 2019. A survey of code-switched speech and language processing. *ArXiv*, abs/1904.00784.

Abhishek Srivastava, Kalika Bali, and Monojit Choudhury. 2020. Understanding script-mixing: A case study of hindi-english bilingual twitter users. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching, CodeSwitch@LREC 2020, Marseille, France, May, 2020*, pages 36–44. European Language Resources Association.

Qianwen Wang and Mizuho Iwaihara. 2019. Deep neural architectures for joint named entity recognition and disambiguation. In *IEEE International Conference on Big Data and Smart Computing, BigComp 2019, Kyoto, Japan, February 27 - March 2, 2019*, pages 1–4. IEEE.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore. 2012. Learning from bullying traces in social media. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 3-8, 2012, Montréal, Canada*, pages 656–666. The Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, H. Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention. *ArXiv*, abs/2010.01057.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1415–1420. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, June 6-7, 2019*, pages 75–86. Association for Computational Linguistics.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çagri Çöltekin. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 1425–1447. International Committee for Computational Linguistics.