Identifying Aggression and Offensive Language in Code-Mixed Tweets: A Multi-Task Transfer Learning Approach

Bharath Kancharla
k_bharath@cs.iitr.ac.in

Prabhjot Singh
prabhjot_s@cs.iitr.ac.in

Lohith Bhagavan Kancharla
k_lbhagavan@cs.iitr.ac.in

Yashita Chama
c_yashita@cs.iitr.ac.in

Raksha Sharma
raksha.sharma@cs.iitr.ac.in

Abstract

The widespread use of social media has contributed to the increase in hate speech and offensive language, impacting people of all ages. This issue is particularly difficult to address when the text is in a code-mixed language. Twitter is commonly used to express opinions in code-mixed language. In this paper, we introduce a novel Multi-Task Transfer Learning (MTTL) framework to detect aggression and offensive language. By focusing on the dual facets of cyberbullying, *viz.*, aggressiveness and offensiveness, our model leverages the MTTL approach to enhance the performance of the model on the aggression and offensive language detection. Results show that our Multi-Task Transfer Learning (MTTL) setup significantly enhances the performance of state-of-the-art pretrained language models, *viz.*, BERT, RoBERTa, and Hing-RoBERTa for Hindi-English code-mixed data from Twitter.

1 Introduction

Social media encompasses a variety of internet-based applications that enable people to connect globally and share user-generated content. Platforms like Twitter and Facebook are among the most popular applications on the internet today. However, there has been a significant rise in bullying behavior on these platforms, including snide remarks, abusive language, personal attacks, and even threats of rape and violence, impacting children, individuals, and communities. This situation underscores the need for technological advancements to automatically detect offensive content and create safer environments. Machine learning models, leveraging recent techniques in natural language processing, can be utilized to effectively identify such harmful behaviors.

In countries where English is not the native language, such as India, most social media users communicate using at least two languages, predominantly English and Hindi. These texts are classified as bilingual. In a bilingual context, an entire post may be written in the script of one language while incorporating words from both languages, a phenomenon known as code-mixed (or mixed-code) text.

In this paper, we introduce a pioneering Multi-Task Transfer Learning (MTTL) framework aimed at identifying aggression and offensive language in Hindi-English code-mixed tweets. Our method delves into the correlation between aggression and offensive language. As illustrated in Figure 1, it reveals that offensive language frequently accompanies expressions of aggression, suggesting an inherent connection between the two. We validate our MTTL framework using the dataset provided for the seventh Workshop on Online Abuse and Harms (WOAH) (Nafis et al., 2023). Derived from Twitter, this dataset classifies tweets based on two primary dimensions of cyberbullying: aggressiveness and offensiveness. Each tweet is annotated with the following labels.

- **Aggression** has been defined as any behavior enacted with the intention of harming another person who is motivated to avoid that harm. This label consists of 3 sub classes:
  1. (**OAG**) - overtly aggressive
  2. (**CAG**) - covertly aggressive
  3. (**NAG**) - not-aggressive
- **Offensiveness** has been described as any word or string of words which has or can have a negative impact on the sense of self or well-being of those who encounter it– that is, it makes or can make them feel mildly or extremely discomfited, insulted, hurt or frightened. This label consists of 2 sub classes:
  1. (**OFF**) - offensive
  2. (**NOT**) - not-offensive

Proceedings of the First Workshop on Natural Language Processing for Indo-Aryan and Dravidian Languages (IndoNLP2025), pages 148–154
January 20, 2025.

- **Codemixed:** this label specifies weather the tweet is codemixed or monolingual.

The **key contributions** of this work are the following:

- We have proposed a novel MTTL framework for aggression and offensive language detection tasks. We deploy state-of-the-art pretrained language models *viz.*, Hing-RoBERTa (Nayak and Joshi, 2022), BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), and XLM-RoBERTa (Conneau et al., 2019) using Multi-Task Transfer Learning (MTTL) with the aim of optimizing the model's performance in detecting aggression and offensive language within the dataset.

- Extensive experiments were conducted on each sub-task independently, using monolingual, code-mixed, and combined texts. The results highlight significant improvements in detecting both tasks with the MTTL approach. Notably, MTTL-Hing-RoBERTa, MTTL-BERT, and MTTL-RoBERTa demonstrate superior performance across various categories, as depicted in the table 2.

The rest of the paper is organized as follows. Section 2 presents the associated literature. Section 4 describes the proposed MTTL approach and associated loss function. Section 3 describes the dataset. Section 5 presents the experimental setup. Section 6 elaborates the results and Section 7 concludes the paper.

2 Related work

Previous research on aggression/hate speech detection has explored various approaches. These include a unified multi-modal deep learning architecture that integrates Deep Pyramid CNN, Pooled BiLSTM, and Disconnected RNN (Khandelwal and Kumar, 2020). Additionally, studies have investigated the utilization of word-level semantic information and sub-word knowledge to counter character-level adversarial attacks (Mou et al., 2020). Another approach involves a Tabnet classifier-based model trained on features extracted by MuRIL from transliterated code-mixed data, which has demonstrated efficacy even with Devanagari text (Chopra et al., 2023). Moreover, techniques such as data balancing using Generative Pre-trained Transformer (GPT-2) have been explored due to its contextual understanding and capability for more realistic data generation (Shrivastava et al., 2021).

Recent studies on offensive language detection have explored different machine learning algorithms and n-gram feature sets to identify offensiveness in social media messages (Pathak et al., 2021). Additionally, researchers have combined various multilingual transformer-based embedding models with machine learning classifiers to detect hate speech and offensive language in code-mixed text in Dravidian languages (Sreelakshmi et al., 2024). Furthermore, leveraging LSTM architecture, Zypherand, openchat-3.5, along with prompt engineering and QLoRA, has shown promising potential in addressing the challenges of hate and offensive comment classification (Shaik et al., 2024).

Research on Multi-Task Learning and Transfer Learning has explored various methodologies. These include proposing an unsupervised multi-task learning network that estimates bullying likelihood using a Gaussian Mixture Model (Cheng et al., 2020), utilizing cross-lingual contextual word embeddings and transfer learning for predictions in low-resource languages (Ranasinghe and Zampieri, 2021), enhancing AraBERT with Multi-task learning to effectively learn from limited Arabic data (Djandji et al., 2020), employing Multinomial Naive Bayes for textual data and ResNet50 for pictorial data, and integrating the results from both to identify misogynistic memes (H et al., 2024). Additionally, combining AdapterFusion with language adapters on a multilingual Large Language Model (LLM) has been explored for classifying code-mixed and code-switched social media text (Rathnayake et al., 2024). Moreover, a multi-task model based on the shared-private scheme has been proposed to capture both shared and task-specific features (Kapil and Ekbal, 2020).

In this paper, we also introduce a multi-task transfer learning approach, leveraging the intrinsic relationship between aggression and offensive language.

3 Dataset and Preprocessing

The dataset (Nafis et al., 2023) consists of 10000 tweet IDs, each labeled with offensiveness labels (OFF or NOT) and aggressiveness labels (OAG,CAG,or NAG) in addition with codemixed labels (codemixed or monolingual). We were able to retrieve text from 8281 tweets from the tweet

IDs provided in the dataset,the remaining tweets were most probably deleted. We partitioned this data randomly into an 80% training set, 10% validation set, and 10% evaluation set. Table 1 shows the distribution of the different labels across each data split.

Split	Class	OAG	CAG	NAG	OFF	NOT
Train	Codemixed	757	882	1400	1136	1903
	Monolingual	729	1137	1719	850	2735
	Combined	1486	2019	3119	1986	4638
Validation	Codemixed	83	118	197	142	256
	Monolingual	90	123	217	98	332
	Combined	173	241	414	240	588
Evaluation	Codemixed	93	118	177	140	248
	Monolingual	94	144	203	108	333
	Combined	187	262	380	248	581

Table 1: Dataset distribution

Among the 8281 instances, 4368 instances are labelled as aggressive(OAG + CAG) and 2474 instances are labelled as offensive (OFF). Of the 2474 offensive instances, 2150 overlap with the aggressive instances, as shown in Figure 1. The Venn diagram indicates that generally offensive language is used when people are aggressive(i.e., most of the offensive tweets are aggressive), highlighting a strong relationship between aggression and offensive language in the dataset.

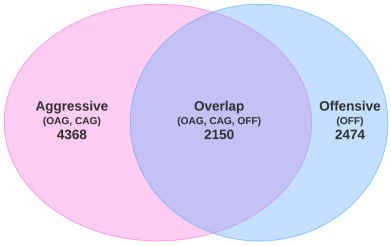

Figure 1: Overlap in aggressive and offensive instances

3.1 Preprocessing

In the preprocessing phase, we masked all the user mentions and retweet mentions with the token '@user' (e.g., `@narendramodi` → `@user`) to ensure the model does not learn features based on user-IDs. We further tokenized this data using the tokenizer corresponding to the selected pretrained language model to make sure the input would be compatible with the common layers input. We precisely applied all these preprocessing steps to each experiment conducted for both the sub tasks.

4 Proposed Model

We based our approach on the multi-task model based on the shared-private scheme that captures the shared-features and task-specific features (Kapil and Ekbal, 2020) and leverage the pretrained language models that have achieved a state-of-the-art performance in multiple Hindi-English NLP tasks. Our best model is based on augmenting the pretrained language model with task-specific layers and sharing the knowledge between them through transfer learning to achieve multi-task learning. We chose this approach to explore the relationship between aggressiveness and offensiveness of the text, and the results are more impressive than the models that achieved state-of-the-art performance in detecting aggression and offensive language from the text. [1]

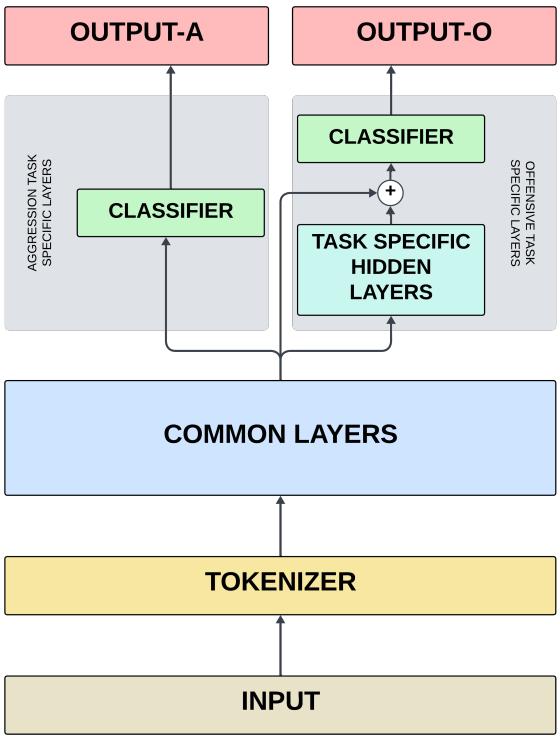

Figure 2: Model architecture

4.1 Multi-Task Transfer Learning (MTTL)

Multi-Task learning (MTL) is an approach in machine learning where a model is trained simultaneously on multiple tasks. By sharing representations between related tasks, the model can often improve performance on individual tasks compared to training separate models for each task. The core

[1] https://github.com/opius005/Aggression-and-Offensive-Language-Detection

idea is that learning to perform multiple related tasks can help a model generalize better because it captures commonalities and differences among the tasks. Transfer Learning (TL) is a technique where a model developed for a particular task is reused as the starting point for a model on a second task. It leverages the knowledge gained while solving one problem and applies it to a different but related problem. The key idea behind Multi-Task Transfer Learning (MTTL) is to combine the ideas of multi-task learning and transfer learning. This approach transfers the knowledge learned from multiple source tasks to improve learning for one or more target tasks. The aim is to leverage the shared information between the tasks to enhance the learning efficiency and performance of the target tasks. In our case, we have two sub-tasks, Aggressiveness and Offensiveness of the text; we employ the MTTL approach to augment the pretrained language model such that it can learn both tasks simultaneously, and we mainly focus on optimizing the performance of the model on both tasks by sharing the task-specific knowledge. Our MTTL model architecture consists of two components, as can be seen in Figure 2.

1. Common Layers: These layers include the pretrained language model, which is fine-tuned based on the combined weighted loss of both tasks to extract general features representing shared information between the tasks.

2. Task-Specific Layers: These layers consist of task-specific hidden layers and classification heads, designed to capture unique features for each task. They are fine-tuned based on the individual loss associated with each specific task.

From Figure 1, we can see that the number of aggression instances is almost the same as the combined task instances, while the number of offensive instances is nearly half of the combined task instances. This explains why adding task-specific hidden layers to the offensive task model helps capture task-specific features effectively, whereas adding such layers to the aggression task model leads to overfitting.

4.2 Loss Function

We need two different loss functions to efficiently tune the task specific layers and common layers to capture task specific features and common features respectively.

4.2.1 Individual Loss Function:

Cross-entropy loss is useful in classification tasks, weighted cross-entropy loss is an extension of the standard cross-entropy loss that applies different weights to different classes. This is particularly useful in scenarios where the class distribution is imbalanced, allowing the model to pay more attention to underrepresented classes. The mathematical formulation of weighted cross-entropy loss of a class i with weight W_i is given in Equation 1, the weight vector W_i is given in Equation 2.

$$L_{task}(x_i) = -W_i log\left(\frac{exp(x_i)}{\sum_j exp(x_i)}\right) \quad (1)$$

$$W_i = \frac{N°samples}{N°classes \times Count_i} \quad (2)$$

4.2.2 Overall Loss Function:

After deriving the individual losses of each task, we defined a custom loss function to compute the overall loss as weighted sum of the individual losses L_{agg}(loss of aggression task) and L_{off}(loss of offensive task) with parameter $w_l \in (0, 1)$.

$$Loss(x_i) = [w_l \times L_{agg}(x_i)] + [(1 - w_l) \times L_{off}(x_i)] \quad (3)$$

By adjusting the parameter w_l, we can direct the model to prioritize learning a specific task. Since our primary focus is on optimizing the model to detect offensiveness in the text, we will set the value of w_l accordingly.

5 Experimental Setup

We fine-tune the two tasks using the following pretrained language models: BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) which are trained on English data, XLM-RoBERTa (Conneau et al., 2019) which is trained over multilingual data, Hing-RoBERTa (Nayak and Joshi, 2022) a multilingual language model specifically built for Hindi-English code-mixed language as seen in the Indian context. These are the state-of-the-art models chosen by the authors of the dataset to evaluate their dataset.

We perform the experiments using the Huggingface Transformers library (Wolf et al., 2020). We monitor the validation set's macro-F1 scores to find the best hyper-parameter values, using the following range of values for selecting the best hyper-parameter:

MODEL	Offensive Language Detection			Aggression Detection		
	Combined	Codemixed	Monolingual	Combined	Codemixed	Monolingual
$BERT_{base}$	75.63	75.77	71.61	57.95	52.29	50.36
MTTL-$BERT_{base}$	79.03(+3.40)	80.78(+5.01)	**79.29(+7.68)**	64.10(+6.15)	63.32(+11.03)	61.48(+11.12)
$RoBERTa_{base}$	76.31	77.66	67.30	60.70	62.44	60.65
MTTL-$RoBERTa_{base}$	79.08(+2.77)	79.15(+1.49)	76.63(+9.33)	63.76(+3.06)	64.60(+2.16)	**64.68(+4.03)**
XLM-R_{base}	76.38	77.91	74.21	60.58	61.25	47.51
MTTL-XLM-R_{base}	76.45(+0.07)	73.61(-4.30)	74.91(+0.70)	64.29(+3.71)	62.07(+0.82)	60.44(+12.93)
Hing-RoBERTa	78.61	77.45	70.92	64.85	61.88	57.77
MTTL-Hing-RoBERTa	**82.03(+3.42)**	**81.61(+4.16)**	76.02(+5.10)	**67.01(+2.16)**	**69.10(+7.22)**	63.99(+6.22)

Table 2: Macro F1-scores obtained from pretrained language models on the dataset and the models augmented with MTTL approach are represented with 'MTTL' as the prefix. The values inside (.) represent the change in Macro-F1 score and the values in **bold** highlight represent the best-performing language model on each category of the dataset.

- w_l: [0.3, 0.4, 0.5, 0.6, 0.7, 0.8]
- No. of task specific hidden layers: [1, 2, 3, 4]
- Batch size: [4, 8, 16, 32]
- Learning rate: [1e-6, 2e-5, 2e-6, 5e-5, 5e-6]
- Number of training epochs: [2, 3, 4]

6 Results

The individual performance of these models on the two tasks, corresponding with codemixed (Hindi+English), monolingual (only English), and combined data (codemixed+monolingual) as input is shown in Table 2 with Macro-F1 as the metric. The performance of the pretrained language models fine-tuned with the MTTL approach is represented with 'MTTL' as the prefix is also shown in Table 2. We only show the results of our best MTTL model on the evaluation set in Table 2. We observed that the MTTL approach shows consistent improvement in almost all cases with MTTL-Hing-RoBERTa outperforming other models with Macro-F1 scores of 82.03%, 81.61% and 76.02% with an improvement of 3.42%, 4.16% and 5.10% respectively on combined, codemixed and monolingual data on offensive language detection and 67.01%, 69.10% and 63.99% with an improvement of 2.16%, 7.22% and 6.22% respectively on combined, codemixed and monolingual data on aggression detection. The results show that not only Hing-RoBERTa but also BERT-base, RoBERTa-base, and XLM-RoBERTa-base models show significant improvements in their performance with the MTTL approach.

6.1 Parameter Analysis

The parameter w_l plays a significant role in the model's performance on each task. The optimal performance of the MTTL model on the aggression task is observed when $0.5 < w_l < 1$, and on the offensive task, is observed when $0 < w_l < 0.5$ because the value of the w_l is indirectly the proportion of importance given to specific task. Note when the value of w_l is not optimal at the extreme value (i.e, 0 and 1) because the model completely learns only one task, nullifying the MTTL effect. We have only shown the results of our best MTTL model on each task with w_l tuned for that specific task in the given range. We explored the use of different numbers of task-specific hidden layers for each independent task to enhance the learning of task-specific features. However, we found that adding these layers to the aggression task led to overfitting on this dataset. Note that we are proposing to not to add any aggression task-specific layers to mitigate the overfitting issues for the given dataset. The model may perform better with task-specific layers for each task on other datasets depending on the dataset's class distribution.

7 Conclusion

Cyberbullying on social media platforms is a significant issue affecting many individuals, with the diverse languages and dialects in India posing a substantial challenge for automated offensive language detection systems. In this paper, we propose a Multi-Task Transfer Learning (MTTL) framework enhanced with pretrained language models like Hing-RoBERTa to efficiently learn multiple tasks and improve performance in detecting aggression and offensive language in Hindi-English code-mixed text. We explored the use of individual weighted loss functions for training task-specific layers and a custom overall loss function for training common layers. Our results demonstrate signif-

icant improvements with the MTTL approach over single-task learning across various pretrained language models, including Hing-RoBERTa, BERT, RoBERTa, and XLM-RoBERTa. Notably, MTTL-Hing-RoBERTa outperformed other models on non-monolingual data, while MTTL-BERT and MTTL-RoBERTa showed the best performance on monolingual data.

Limitations

The dataset primarily focuses on Hindi-English code-mixed tweets. While this is appropriate for the specific application, it limits the generalizability of the findings to other code-mixed languages or purely monolingual datasets. The proposed framework relies on pretrained language models such as BERT, RoBERTa, XLM-RoBERTa, and Hing-RoBERTa. These models may carry inherent biases or limitations from their original training data, which could influence their ability to accurately classify aggression and offensive language in a diverse range of contexts.

References

Lu Cheng, Kai Shu, Siqi Wu, Yasin N. Silva, Deborah L. Hall, and Huan Liu. 2020. Unsupervised cyberbullying detection via time-informed gaussian mixture model. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 185–194, New York, NY, USA. Association for Computing Machinery.

Abhishek Chopra, Deepak Kumar Sharma, Aashna Jha, and Uttam Ghosh. 2023. A framework for online hate speech detection on code-mixed hindi-english text and hindi text in devanagari. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(5).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Marc Djandji, Fady Baly, Wissam Antoun, and Hazem Hajj. 2020. Multi-task learning using AraBert for offensive language detection. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 97–101, Marseille, France. European Language Resource Association.

Shaun H, Samyuktaa Sivakumar, Rohan R, Nikilesh Jayaguptha, and Durairaj Thenmozhi. 2024. Quartet@LT-EDI 2024: A SVM-ResNet50 approach for multitask meme classification - unraveling misogynistic and trolls in online memes. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 221–226, St. Julian's, Malta. Association for Computational Linguistics.

Prashant Kapil and Asif Ekbal. 2020. A deep neural network based multi-task learning approach to hate speech detection. *Knowledge-Based Systems*, 210:106458.

Anant Khandelwal and Niraj Kumar. 2020. A unified system for aggression identification in english code-mixed and uni-lingual texts. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, CoDS COMAD 2020, page 55–64, New York, NY, USA. Association for Computing Machinery.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

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*, CIKM '20, page 1145–1154, New York, NY, USA. Association for Computing Machinery.

Nazia Nafis, Diptesh Kanojia, Naveen Saini, and Rudra Murthy. 2023. Towards safer communities: Detecting aggression and offensive language in code-mixed tweets to combat cyberbullying. In *The 7th Workshop on Online Abuse and Harms (WOAH)*, pages 29–41, Toronto, Canada. Association for Computational Linguistics.

Ravindra Nayak and Raviraj Joshi. 2022. L3Cube-HingCorpus and HingBERT: A code mixed Hindi-English dataset and BERT language models. In *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 7–12, Marseille, France. European Language Resources Association.

Varsha Pathak, Manish Joshi, Prasad Joshi, Monica Mundada, and Tanmay Joshi. 2021. Kbcnmujal@hasoc-dravidian-codemix-fire2020: Using machine learning for detection of hate speech and offensive code-mixed social media text. *Preprint*, arXiv:2102.09866.

Tharindu Ranasinghe and Marcos Zampieri. 2021. Multilingual offensive language identification for low-resource languages. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 21(1).

Himashi Rathnayake, Janani Sumanapala, Raveesha Rukshani, and Surangika Ranathunga. 2024.

Adapterfusion-based multi-task learning for code-mixed and code-switched text classification. *Engineering Applications of Artificial Intelligence*, 127:107239.

Zuhair Shaik, Sai Kartheek Reddy Kasu, Sunil Saumya, and Shankar Biradar. 2024. IIITDWD-zk@DravidianLangTech-2024: Leveraging the power of language models for hate speech detection in Telugu-English code-mixed text. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 134–139, St. Julian's, Malta. Association for Computational Linguistics.

Adarsh Shrivastava, Rushikesh Pupale, and Pradeep Singh. 2021. Enhancing aggression detection using gpt-2 based data balancing technique. In *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 1345–1350.

K. Sreelakshmi, B. Premjith, Bharathi Raja Chakravarthi, and K. P. Soman. 2024. Detection of hate speech and offensive language codemix text in dravidian languages using cost-sensitive learning approach. *IEEE Access*, 12:20064–20090.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.