# Emojis Trash or Treasure: Utilizing Emoji to Aid Hate Speech Detection

**Tanik Saikh, Soham Barman, Harsh Kumar, Saswat Sahu, Souvick Palit**
School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT)
Deemed to be University, Bhubaneswar-751024, Odisha, India.
{tanik.saikhfcs,20051423,20051146,20051413,20051178}@kiit.ac.in

## Abstract

In this study, we delve into the fascinating realm of emojis and their impact on identifying hate speech in both Bengali and English languages. Through extensive exploration of various techniques, particularly the integration of Multilingual BERT (MBert) and Emoji2Vec embeddings, we strive to shed light on the immense potential of emojis in this detection process. By meticulously comparing these advanced models with conventional approaches, we uncover the intricate contextual cues that emojis bring to the table. Ultimately, our discoveries underscore the invaluable role of emojis in hate speech detection, thereby providing valuable insights for the creation of resilient and context-aware systems to combat online toxicity. Our findings showcase the potential of emojis as valuable assets rather than mere embellishments in the domain of hate speech detection. By leveraging the combined strength of MBert and Emoji2Vec, our models exhibit enhanced capabilities in deciphering the emotional subtleties often intertwined with hate speech expressions.

## 1 Introduction

In the rapidly evolving landscape of online communication, social media platforms have emerged as crucial channels for individuals to share their ideas and perspectives. However, the unbridled nature of this digital freedom has occasionally given rise to a concerning issue: the rampant spread of hate speech. This perilous outcome has the potential to inflict profound harm and sow discord among communities. Detecting and effectively addressing hate speech in such a diverse and globalized space has become an urgent digital- age imperative.

This write-up sets out to present a holistic and innovative solution that harnesses the power of advanced natural language processing (NLP) techniques to counter the challenge of hate speech in the Bengali language. By integrating the capabilities of Multilingual BERT and Emoji2Vec models, we hope to achieve a multi-dimensional approach that goes beyond superficial analysis. Through an in-depth examination of the data ecosystem, we uncover the nuances that surround hate speech. The incorporation of emoji embeddings, a rich source of emotional context in digital conversations, is expected to provide valuable cues for detecting subtle instances of hate speech that might evade traditional approaches.

Central to our exploration is the Multilingual BERT model, a pioneering development in NLP. The model's architecture, rooted in pre-training and fine-tuning phases, equips it with the capacity to understand and contextualize the intricacies of language. We will dissect this architecture to reveal how it enables the model to discern the fine line between genuine expression and hate-driven content, especially when the medium is Bengali.

The culmination of this paper is the meticulous analysis of the results obtained through the proposed approaches. We will gauge the efficacy of our method in accurately identifying hate speech and dissect the cases in which the model excelled or faced challenges. Ultimately, this comprehensive approach not only detects hate speech but also paves the way for more inclusive and respectful digital interactions, transforming social media platforms into safer and more conducive spaces for open dialogue and shared understanding.

## 2 Related Work

In recent years, hate speech detection has received significant attention, with the majority of work dedicated to monolingual hate speech detection. These skewed in attention to monolingual language has opened ample room to carry research forward further in other languages too. The task defined in (Deng et al., 2022) presented a benchmark for Chinese offensive language analysis named *COLD*. They provided a dataset called *COLDATASET* and a system, namely *COLDETECTOR* adopting transformer based BERT architecture. They showed that the *COLD* benchmark contributes to Chinese offensive language detection which is challenging for existing resources. They then deploy the *COLDETECTOR* and conduct extensive analysis on popular Chinese pre-trained language models. They first assess the offensiveness of existing generative models and show that these models necessarily expose varied degrees of offensive concerns. Furthermore, they analyzed the aspects that influence the offensive generations, and concluded that anti-bias contents and terms referring to certain groups or exhibiting unfavorable attitudes activate offensive outputs faster. The task of (Vanetik and Mimoun, 2022) presented an automated system to detect offensive language and racism in French. This kind of task has been a pressing need as the surge in internet usage has led to a notable rise in toxic online content, posing a significant issue in recent years. The system tested on the novel *French Twitter Racist (FTR)* speech dataset, for racist speech detection. The experiments were carried out to achieve three primary objectives, *viz.* (1). the assessment and comparative analysis of diverse models and textual representations within the French language context; (2). the execution of cross-lingual experiments designed to ascertain the efficacy of transfer learning employing the proposed methodology, particularly in scenarios characterized by limited linguistic resources; and (3). the conduct of multilingual experiments investigating the potential enhancement of classification accuracy through the incorporation of an additional language (e.g., English) into the training dataset. The investigation and analysis were conducted as part of information 2022, with a specific focus on addressing the aforementioned research goals. They evaluated the dataset with the use of alternative text representations and supervised learning approaches for racist text detection in social media. They showed that extending the FTR dataset with extra French data containing hate speech is beneficial because it leads to better scores for practically all models, like random forest, logistic regression, extreme gradient boosting, BERT Transformer, and text representations. They also run cross-lingual and multilingual experiments for evaluating a theory about transfer learning. The study conducted by (Velankar et al., 2021) proposed various deep learning-based approaches (like LSTM, CNN and BERT-based models) on Hindi and Marathi datasets released as a part of HASOC-2021 (Modha et al., 2022). They performed binary and multi-class classification tasks. For the binary classification task, they employed CNN and LSTM incorporating random and FastText embeddings. Among these, the LSTM + non-trainable FastText configuration exhibited optimal performance for Marathi, while for Hindi, BiLSTM with non-trainable FastText demonstrated superior efficacy. Additionally, they explored transformer-based BERT models such as IndicBERT, mBERT, and RoBERTa-base for Marathi, as well as RoBERTa base and Neural Space BERT for Hindi. IndicBERT outperformed other models in the case of Marathi, while RoBERTa yielded the best results for Hindi. The same RoBERTa model was utilized in a hierarchical approach. The findings underscore the superior performance of transformer-based models in binary tasks, although even fundamental models demonstrated competitive performance. In the context of Hindi Task 2, the study establishes that the CNN + non-trainable FastText model slightly outperforms the RoBERTa Hindi model. The task of (Dhanya and Balakrishnan, 2021) presented a brief survey of hate speech detection in Asian languages. The survey's primary goal is to promote the development of an automated hate speech detection system tailored specifically for Malayalam. Messages on social media that have negative social impacts, touching on subjects such as sex, caste, religion, politics, race, etc., are classified as hateful messages, posing a significant detection challenge. The study exclusively considers language-specific research on hate speech detection, analyzing the methodologies employed in each instance. Three parameters are used to assess the overall landscape of this issue across Asian languages. The study seeks to identify the optimal classification algorithm for this task and explore the relationship between classification approach, dataset type and size, and accuracy. The article demonstrated diverse language-specific papers to automate the detection of hate speech in Asian languages. The researchers employed a spectrum of machine learning (ML) classification techniques and deep learning algorithms to address this issue. The findings indicate that the support vector machine (SVM) emerged as a widely preferred algorithm for binary-level categorization in this context. However, the accuracy of each study was found to be contingent upon factors such as the nature of the dataset (balanced or imbalanced) and its size. The survey underscores a positive correlation between accuracy and dataset size, revealing that accuracy tends to increase with larger datasets. Additionally, certain studies focusing on multi-label categorization demonstrated superior performance. The task defined in (Saha et al., 2021) demonstrated a comprehensive exploration of different transformer-based models. Additionally, a genetic algorithm-based technique for assembling diverse models has been introduced. The ensembled models, trained separately on each language, achieved the first, second, and third positions in Tamil, Kannada, and the Malayalam sub-task, respectively. The research conducted involved the exploration of ensemble methodologies, recognizing the potential for improved predictive capabilities through the amalgamation of diverse models, as opposed to reliance on a singular classifier. Acknowledging the limitations of conventional prediction averaging ensembles, especially when weaker models are present, the researchers adopted a strategy incorporating model weights based on individual performances. This approach aligns with established genetic algorithm (GA) techniques for determining optimal weights within the ensemble. It is observed that, among the individual transformer models, optimal performance is achieved by employing XLM-RoBERTa-large (XLMR-large) for the Tamil dataset and Custom XLM-RoBERTa-base (XLMR-C) for the Kannada dataset. In the case of the Malayalam dataset, both aforementioned models demonstrate comparable performance. The heightened effectiveness of XLM-RoBERTa models can be attributed to their pretraining using a parallel corpus, which is consistent across different languages. Subsequent pretraining with the specific dataset further contributes to performance improvement in the Kannada dataset. The utilization of the larger XLM-R model was precluded due to limitations in GPU space. Furthermore, the study addresses the performance of fusion models, emphasizing their nearly identical performance across various combinations. In the task of (Khan et al., 2021) approximately 100,000 tweets originating from South Asia, with potential hate-speech content, were collected through web scraping. Subsequently, the tweets were manually parsed to identify those in Roman Urdu. An iterative approach was then employed to formulate guidelines for hate-speech detection, which were subsequently used to create a hate-speech corpus specifically for Roman-Urdu. The resulting corpus comprises over 5000 tweets, predominantly containing hate speech. The initial categorization involved classifying tweets into hostile and neutral, followed by a secondary categorization of hostile tweets into offensive and hate speech. The evaluation of this developed corpus was conducted using various supervised learning techniques. The research sought to illustrate the utility of the curated dataset for the hate-speech detection task. In pursuit of this objective, a series of experiments were conducted employing five distinct supervised ML techniques encompassing both classic and deep learning methodologies. This dichotomy between classic and deep learning approaches has been extensively explored in the existing literature, as evidenced by the works of various scholars. Noteworthy observations in the literature, as exemplified in certain instances, suggest that classic techniques may outperform deep learning methodologies under specific conditions. For instance, when confronted with a relatively small training dataset, classic techniques have demonstrated superior performance. Conversely, on larger datasets, deep learning methodologies have exhibited a propensity to surpass the efficacy of classical techniques. These nuanced findings underscore the contextual dependence and interplay between dataset size and the comparative performance of classic and deep learning approaches in hate-speech detection applications. Various supervised learning techniques were employed to assess their efficacy in identifying objectionable language, discerning between basic and intricate sentences, and categorizing sentences as offensive or involving hate speech. Additionally, rules were devised to distinguish between hate speech and offensive speech. The experimental results yielded the following insights: a). Logistic Regression emerged as the most effective technique for discerning between neutral and hostile sentences, with Count Vectors proving to be the most efficient features; b). Logistic Regression, when coupled with Count Vectors, was also identified as the most effective technique for differentiating between hateful and offensive sentences; c). Both Character n-grams and Word n-grams exhibited suboptimal performance on the developed corpus, primarily attributed to issues related to spelling variations in Roman Urdu; d) The detection of sarcasm proved challenging, as there were instances where no explicitly hateful or offensive words were used in the comments, resulting in the classification of comment polarity as neutral. The shared task in association with *the Forum for Information Retrieval Evaluation (FIRE)* 2020 on Detecting Hate Speech and Offensive Content in German has been described in (Que et al., 2020).

## 3 Dataset

We initially intended to utilize well-known benchmark hate speech datasets to conduct a comprehensive analysis and effectively compare our proposed methodologies with existing works. Regrettably, the paucity of studies focusing on speech inclusive of emojis has constrained our options, leaving us with only two datasets that align with our requirements. For the English dataset, the **emoji2vec** dataset comprises a collection of tweets, meticulously stored in a pickle file and thoughtfully divided into both training and test sets. This dataset encompasses a total of 64,599 tweets, with 51,679 tweets allocated to the training set and 12,920 tweets assigned to the test set. Each of these tweets has been diligently labeled as positive (1), neutral (0), or negative (-1), providing a comprehensive understanding of the sentiment. For the Bengali, the dataset consists of a substantial collection of 30,000 comments, revealing a significant disparity as 10,000 comments are categorized as hate speech. It is particularly noteworthy that this bias is more prevalent in areas such as celebrity and politics, where instances of hate speech are relatively infrequent. However, accurately labeling hate speech poses challenges due to the intricate contextual nuances, often resulting in misclassification. Analyzing the average length of the text reveals interesting variations, with meme comments being notably concise, while comments related to celebrities tend to be more extensive. Comparing this dataset with state-of-the-art collections emphasizes the remarkable linguistic diversity present, underscoring the necessity of considering different categories for a comprehensive linguistic analysis.

## 4 Model

The proposed model incorporates the sophisticated encoder-decoder architecture, a powerful framework that enhances the understanding and interpretation of sentences. By embedding words into a sequence of vectors, the encoder skillfully captures the essence of the sentence, enabling a comprehensive comprehension. The decoder subsequently utilizes this sequence of word embeddings to generate accurate predictions, elevating the overall performance of our system. We chose the most popular transformer-based encoders and emoji2vec word embeddings to help us understand and analyze emojis. Emoji2vec word embeddings were created by assigning numbers to emojis and their official descriptions. We also matched each emoji with a sequence of word embeddings from Google word2vec that represents its official description. For each emoji *i*, we define a trainable vector of float numbers $x_i$ and a vector of float numbers $v_i$ that represents its official textual description. Also, we match each emoji with its official description that is expressed as a sequence of vectorized word embeddings $w_1$, ..., $w_n$ from Google word2vec as follows: $v_i = \sum_{k=1}^{n} w_k$. To train the vector representation $x_i$ for emoji i, we generate emoji-description pairs that are either a true match (the emoji i matches the description j) or a false match (the emoji i does not match the description j, but is paired with a description of another emoji). Then, we train the vectors xi's with log-likelihood loss: Here, $y_{ij}$ represents the binary label indicating whether the pair is a true match ($y_{ij}$=1) or a false match ($y_{ij} = 0$). The optimization process aims to minimize this loss function, contributing to the learning of accurate vector representations for the given emojis. In the realm of transformer encoders, a crucial component is the self-attention mechanism, which empowers the calculation of attention for each word embedding towards the context. This process unfolds through a relatively straightforward mathematical operation, as elucidated below. Imagine you have a set of data represented as X, where each element is a word in a sequence. Think of B as the number of sets, S as the length of each sequence, and D as the dimension of the initial word representations. Now, there are three operations called query (Q), key (K), and value (V) for the word sequences. These operations are calculated by multiplying the input data X with trainable weight matrices named $W_Q$, $W_K$, and $W_V$. In simpler terms: Q (query) is obtained by multiplying X with $W_Q$. K (key) is obtained by multiplying X with $W_K$. V (value) is obtained by multiplying X with $W_V$. These operations form the basis of the self-attention mechanism, enabling each word's representation to adapt and consider its surrounding context. This adaptability is a key factor contributing to the transformer encoder's effective processing capabilities. Attention-based word embeddings ($X_{att}$) can then be calculated by the scaled dot product attention formula: A formula needs to be introduced here.

### 4.1 Leveraging Emoji2Vec Embeddings: Unveiling Emotional Context

Emojis, the vibrant and universally understood visual symbols, have transcended their origins as mere embellishments in digital communication. They serve as emotional punctuation marks, imbuing the written text with a spectrum of sentiments. Recognizing the power of emojis, our methodology introduces Emoji2Vec embeddings. These embeddings transform emojis into high-dimensional numerical vectors, creating a bridge between textual content and the realm of emotions. By integrating Emoji2Vec embeddings into the model, it gains access to the emotional nuances that often elude traditional text analysis techniques. Emoji2Vec is a concept drawn from the same lineage as Word2Vec—an embedding technique that transforms words into numerical vectors that capture their semantic relationships. However, while Word2Vec aims to capture the semantic meaning of words, Emoji2Vec focuses on representing the emotional nuances and context carried by emojis. Emojis hold immense potential to enhance the understanding of complex text characteristics. Here's how Emoji2Vec enables us to decode the depth of emotions and context within textual content:

***Nuanced Emotion Detection:*** Emojis are often used to convey emotions ranging from joy and sadness to anger and surprise. Emoji2Vec's embeddings allow algorithms to identify these emotional cues, enabling systems to gauge the underlying sentiment even when words alone may not indicate it.

***Contextual Sensitivity:*** The meaning of an emoji can dramatically change based on its context within a sentence or conversation. Emoji2Vec's ability to capture these contextual nuances empowers algorithms to discern between different interpretations of the same emoji, refining their understanding of the conversation.

***Understanding Irony and Sarcasm:*** Emojis can be used ironically or sarcastically, where their intended meaning contradicts their literal depiction. Emoji2Vec's contextual embeddings assist algorithms in recognizing such subtleties, enabling a more accurate interpretation of the intended message.

***Enhanced Sentiment Analysis:*** Emojis contribute significantly to the overall sentiment of a text. By embedding emojis using Emoji2Vec, algorithms can more accurately analyze and classify the sentiment of a piece of text.

### 4.2 The Synergy of Architecture

The architecture of the proposed model ingeniously blends Multilingual BERT's profound contextual comprehension with the perceptive insights of Emoji2Vec, culminating in a comprehensive framework. This construct elegantly amalgamates the realms of linguistic intricacies and emotional depths, empowering the model to scrutinize instances of hate speech from a more panoramic vantage point.

***Text Encoding with Multilingual BERT:*** Multilingual BERT adeptly encodes textual content, encapsulating intricate word relationships and their contextual implications. This encoding assumes paramount importance in comprehending the latent semantics of the text, especially in languages such as Bengali, wherein context exerts a pervasive influence on interpretation. Emoji2Vec embeddings in Action: Emojis, as non-verbal messengers, embellish communicative context through sentiment projection. The incorporation of Emoji2Vec embeddings augments the model's proficiency in processing these symbolic cues and deciphering emotional subtleties. This infusion, synergized with Multilingual BERT's encodings, begets a more holistic articulation of textual significance. Fusing Features for Insight: The marriage of Multilingual BERT's contextual grasp and Emoji2Vec's emotional perspicacity is consummated via feature concatenation. This assimilated feature vector becomes a conduit for both linguistic disposition and affective expression, furnishing an opulent reservoir of information for subsequent layers. Neural Network: Extracting Patterns: The neural network strata play a pivotal role in transmuting the concatenated feature vector into actionable insights. These strata undertake the mantle of identifying patterns, interconnections, and correlations within the dataset. The network's depth is instrumental in ferreting out subtle hate speech indicators, thus elevating the model's discrimination acumen.

### 4.3 Data-processing model

This model comprises several sub-modules *viz.* 1. Control Model 2. Translated Model 3. Added

|  | F1-Score (%) |  |  |  |
|---|---|---|---|---|
|  | Control | Translated | Added | Average |
| English | 72.8 | 68.03 | 69.67 | 78.84 |
| Bengali | 69.47 | 67.34 | 66 | 75.53 |

Table 1: Results obtained in terms of F1-scores on Control, Translated, Added and Average models in English and Bengali languages.

Model and 4. Average Model

***Control Model:*** In this module, the hate speech detection relies exclusively on the textual content processed by the Multilingual BERT-base model. The approach involves encoding the input text "X" using Multilingual BERT, represented as $Y_Control = M_BERT(X)$.

***Translated Model:*** The translated model introduces a step to convert emojis into their corresponding English meanings before processing the text with the Multilingual BERT-base model. Denoting the set of emojis as "E" and the translation function as $T_emoji2text$, the output is obtained as $Y_Translated = M_BERT(X + T_emoji2text(E))$.

***Added Model:*** This model enhances the information by summing up all Emoji2Vec embeddings in a sentence and concatenating the result with the Multilingual BERT-base context. If $V_Emoji2Vec$ represents Emoji2Vec embeddings, the output is given by $Y_Added = M_BERT(X + \sum V_{Emoji2Vec})$.

***Average Model:*** In the average model, we take the average of Emoji2Vec embeddings of contained in a sentence, creating a 30x30 vector. This vector is then concatenated with the Multilingual BERT-base context, expressed as $Y_Average = M_BERT(X + mean(V_Emoji2Vec))$.

## 5 Results and Analysis:

The model has been trained and evaluated using the datasets that are split into training, validation, and test sets. The training process involved optimizing model parameters to minimize the binary cross-entropy loss. After ten epochs of training each model and the two datasets, the results are shown in the following Table 1. The Control model achieves a relatively high F1 score of 72.8% f1 in English,indicating that relying solely on textual evidence with Multilingual BERT yields effective hate speech detection. In Bengali it is 69.47%) showing slightly lower performance compared to English, suggesting potential linguistic nuances impacting hate speech detection.

The Translated Model produces a lower F1 score (68.03%) compared to the Control Model in English. The translation of emojis into English meanings might introduce noise or loss of contextual information. Whereas, in Bengali with 66% f1-score experiencing a more noticeable decrease compared to English, suggesting potential challenges in incorporating emoji embeddings for Bengali. The Average model in English with 78.84% demonstrates a substantial increase in f1-score, outperforming other models. Averaging Emoji2Vec embeddings proves effective, indicating a strong contextual correlation between emojis and hate speech in English. In Bengali with 75.53% maintaining a high F1 score, suggests that averaging emoji embeddings is a robust strategy for hate speech detection in Bengali.

The Average Model stands out as the most effective approach for hate speech detection, demonstrating superior performance across both English and Bengali datasets. This model's success lies in its distinctive strategy of averaging Emoji2Vec embeddings, creating a nuanced and contextually rich representation that significantly enhances the discrimination between hate speech and non-hate speech instances. Averaging Emoji2Vec embeddings offers a multifaceted advantage. Firstly, it captures the collective emotional essence conveyed by emojis in a sentence, resulting in a comprehensive and contextually rich vector representation. This approach is particularly effective in hate speech detection, where understanding the emotional context is pivotal. Moreover, averaging helps mitigate potential noise introduced by individual emojis, allowing the model to focus on the overarching emotional context rather than relying heavily on specific emoji instances. One noteworthy aspect of the Average Model's success is its cross-linguistic applicability. The model demonstrates robust performance in both English and Bengali datasets, showcasing its adaptability and effectiveness in diverse linguistic contexts. This indicates that the model's ability to leverage universal emotional signals from emojis contributes significantly to its effectiveness. The inclusion of emojis in the Average Model holds strategic significance. Emojis, as universal symbols, contribute to a shared emotional language. The model's capability to integrate both linguistic and emotional features enables a more comprehensive understanding of textual content. Emojis, in this context, serve as complementary signals to the textual information, enriching the model's understanding of emotional nuances. The Average Model's proficiency highlights the practical application of considering emojis as valuable contextual cues in NLP tasks.

## 6 Conclusion:

The development of effective hate speech detection models is hindered by the challenge of obtaining a large and diverse dataset that accurately labels instances of hate speech. The limited availability of comprehensive datasets restricts the model's exposure to the various nuances of hate speech, which can lead to decreased effectiveness in detecting hate speech instances that deviate from the limited training data. Hate speech is known for its nuanced expressions, often conveyed through the use of emojis to convey emotions and sentiments. However, the inclusion of emojis in the model introduces complexities, as the model must decipher the emotional subtleties conveyed by these visual symbols. This challenge is particularly demanding when hate speech instances heavily rely on emojis for expression, requiring a model that can adeptly capture these nuanced emotional cues. The constraint of limited computational resources further compounds the challenges. The depth and complexity of neural network architectures necessary for discerning subtle patterns in hate speech may be curtailed due to resource constraints. This limitation hinders the model's capacity to achieve a nuanced understanding of hate speech instances, especially those that require intricate pattern recognition. An additional challenge is the lack of annotations for emojis in existing datasets labeled for hate speech. This absence of annotated emojis in the training data impedes the model's ability to learn and generalize from hate speech instances where emojis play a pivotal role in conveying context and emotion.

Furthermore, relying on free GPU resources can add further constraints to model training. The availability and capacity of these resources may be limited, leading to slower model training times and potential restrictions on the complexity of the neural network. These limitations can impact the model's overall efficiency and its ability to achieve optimal performance in hate speech detection tasks.

The proposed approach combines the strengths of Multilingual BERT and Emoji2Vec embeddings to create a robust model for detecting hate speech in the Bengali language. By incorporating both textual and emoji-based features, the model gains a deeper understanding of context and sentiment. The results demonstrate promising accuracy in identifying hateful content, although further improvements can be made to address the false positives.

## References

Jiawen Deng, Jingyan Zhou, Hao Sun, Chujie Zheng, Fei Mi, Helen Meng, and Minlie Huang. 2022. COLD: A benchmark for Chinese offensive language detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11580–11599, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

L K Dhanya and Kannan Balakrishnan. 2021. Hate speech detection in asian languages:a survey. In *2021 International Conference on Communication, Control and Information Sciences (ICCISc)*, volume 1, pages 1–5.

Muhammad Moin Khan, Khurram Shahzad, and Muhammad Kamran Malik. 2021. Hate speech detection in roman urdu. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 20(1).

Sandip Modha, Thomas Mandl, Gautam Kishore Shahi, Hiren Madhu, Shrey Satapara, Tharindu Ranasinghe, and Marcos Zampieri. 2022. 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*, FIRE '21, page 1–3, New York, NY, USA. Association for Computing Machinery.

Qinyu Que, Ruijie Sun, and Shasha Xie. 2020. Simon@ hasoc 2020: Detecting hate speech and offensive content in german language with bert and ensembles. In *FIRE (Working Notes)*, pages 283–289.

Debjoy Saha, Naman Paharia, Debajit Chakraborty, Punyajoy Saha, and Animesh Mukherjee. 2021. Hate-alert@DravidianLangTech-EACL2021: Ensembling strategies for transformer-based offensive language detection. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 270–276, Kyiv. Association for Computational Linguistics.

Natalia Vanetik and Elisheva Mimoun. 2022. Detection of racist language in french tweets. *Information*, 13(7).

Abhishek Velankar, Hrushikesh Pramod Patil, Amol Gore, Shubham Salunke, and Raviraj Joshi. 2021. Hate and offensive speech detection in hindi and marathi. *ArXiv*, abs/2110.12200.