Identification of Offensive Language in Social Media Comments and Posts

JISHNU.B, SUWINKUMAR.T, TANUSH.K, UDHAYARAJAN.M
Department of Computer Science and Engineering Karpagam College Of Engineering, Othakkalmandapam Coimbatore-641032

1. ABSTRACT:

The rapid proliferation of online content has given rise to an urgent need for automated systems capable of identifying offensive language in comments and posts across various digital platforms. This paper presents a comprehensive approach to address this challenge. We propose a multi-faceted model that combines natural language processing techniques, machine learning algorithms, and deep neural networks to accurately detect offensive language.

Our approach begins with data preprocessing, including tokenization, stemming, and lemmatization, to enhance the model's understanding of text. We employ a wide range of linguistic features, such as word embeddings and syntactic analysis, to capture the nuances of offensive language usage. Additionally, we curate a large and diverse dataset of offensive and non-offensive text to train and fine-tune our model.

To tackle the dynamic nature of offensive language, we incorporate continuous learning techniques that adapt the model over time to evolving language trends and user behaviors. This ensures the model's long-term effectiveness in identifying offensive content.

Evaluation of our approach against benchmark datasets demonstrates its superior performance in terms of precision, recall, and F1-score. Furthermore, we provide insights into the interpretability of the model's decisions, addressing concerns related to transparency and accountability.

Overall, our comprehensive approach offers a robust solution for identifying offensive language in comments and posts, promoting safer and more inclusive online communities.

Keywords: Offensive language identification, Dravidian languages, Code-mixing, Deep learning, MP Net, CNN

2. Problem Definition :

Now a days with the increasing usage of social media platforms like Facebook and Twitter, we often see people to misuse this freedom of speech. Some of them try to use this platform to post the offensive or abusive things about a person or a group. This in turn can negatively impact the mental health of a community/ group or an individual. Considering the sensitivity of the topic, we aim to tackle this problem of detecting the offensive language in social media through cutting edge techniques in Machine Learning, Deep Learning and Natural Language Processing.

To proceed further with our topic we choose the Offensive Language Identification Dataset (OLID) which was released in 2019. But the question is that, why we choose this particular dataset when there exists various datasets in the problem domain. The previous datasets only aims to capture a particular type of offense such as hate speech or cyber bullying, but OLID is a well diverse dataset which covers all the offense types. Therefore we choose this recent novel dataset which opened up various new research opportunities to contribute in.

There are three sub-tasks for this new dataset which involved the detection, predicting the type and the target of the offensive tweet respectively. We particularly focus on predicting whether the post is offensive or not which is the first and the most crucial sub-task for this dataset making it a binary classification problem.

3. INTRODUCTION:

The advent of social media has aided in bridging both political borders and paved the path for individuals to interact with others and express themselves more readily than at any prior point in human history (Edosomwan, Prakasan, Kouame, Watson, & Seymour, 2011). Through the usage of social media platforms, such as Twitter, Facebook, YouTube, Instagram, WhatsApp, Snapchat, and LinkedIn, enormous amounts of information are generated, which allow for data mining and simulation modelling. While microblogging is a relatively new communication medium in comparison with traditional media, it has garnered significant interest from users, organisations, and experts from a variety of sectors (Ye, Dai, an Dong, & Wang, 2021).

The appeal of microblogging originates from its unique characteristics, such as portability, instant messaging, and user-friendliness; these capabilities enable real-time communication with little or no content limitations. However, these platforms have also become spaces where people are targeted, defamed, and marginalised based on their physical appearance, religion, sexual orientation, and many other factors (Benikova, Wojatzki, Zesch, 2018, Keipi, Näsi, Oksanen, Räsänen, 2016, Pamungkas, Basile, Patti, 2020). Social media has developed into a specialised instrument for verbally threatening and cornering people, not based on their actions but on their identities (Maitra, McGowan, 2012, Patton, Eschmann, Butler, 2013, Zinovyeva, Hrdle, Lessmann, 2020). The depth and breadth of this ‘digital marvel’ have enabled previously ‘invisible and socially paralysed’ populations to participate in social discourses (Barnidge, Kim, Sherrill, Luknar, & Zhang, 2019).

At present, the COVID-19 virus is wreaking havoc all over the world. Several studies have revealed the age distribution of users who used offensive phrases on social media during the COVID-19 pandemic, with the 18–24 and 25–34 age groups

accounting for 49 percent of all users (Lyu, Chen, Wang, & Luo, 2020). Thus, it can be said the public fear sparked by rumours and offensive comments on social media is more concerning than the virus's impact (Depoux et al., 2020). Nonetheless, the complexity of event recognition algorithms has hampered the effectiveness of most offensive language detection approaches. While the categorisation of offensive languages using social media data has remained a dynamic area of study, little attention has been paid to the creation of a data, threshold settings, and models for low-resource languages (Ravikiran & Annamalai, 2021). The detection of abusive language in social media sources relies on a variety of approaches from several domains, including machine learning (ML), natural language processing (NLP), data mining, content extraction and retrieval, and text mining. However, social media streams from multilingual nations such as India contain a high proportion of mixed languages; this has a detrimental effect on the effectiveness of categorisation algorithms (Jose, Chakravarthi, Suryawanshi, Sherly, & McCrae, 2020).

In recent years, there have been substantial improvements in research on hate speech identification and offensive language detection (Mandl, Modha, Kumar M, Chakravarthi, 2020, Zampieri, Nakov, Rosenthal, Atanasova, Karadzhov, Mubarak, Derczynski, Pitenis, Çöltekin, 2020) utilising NLP. However, there is still a dearth of research on under-resourced languages. For instance, under-resourced languages such as Tamil, Malayalam, and Kannada lack NLP tools and datasets (Thavareesan, Mahesan, 2019, Thavareesan, Mahesan, 2020, Thavareesan, Mahesan, 2020). Recently, (Chakravarthi et al., 2021c) sentiment analysis as well as language identification for Tamil and Malayalam have paved the way for further studies on Dravidian languages. Tamil, Malayalam, and Kannada are Dravidian languages spoken by approximately 220 million people in India, Singapore, and Sri Lanka (Krishnamurti, 2003). It is critical to develop NLP systems, such as hate speech identification and offensive language detection, for indigenous languages, as the majority of user-generated content is in these languages. Deep learning approaches offer considerable potential in the process of classification detection. However, only a few existing studies demonstrate the value of ensemble approaches in detecting offensive language in low-resource Dravidian languages.

The rest of our work is organised as follows. Section 2 discusses the literature on offensive language identification, while Section 3 introduces the dataset used for the task at hand. Section 4 focuses on the several models and approaches for identifying offensive languages in Dravidian languages. Section 4.9 discusses the details of the implementation of detection methods. Section 5 comprises a detailed analysis on the behaviour and results of the models, which are also compared with other approaches. Finally, Section 6 concludes our work and discusses the potential directions for future work on offensive language identification in Dravidian languages.

The increasing number of online platforms for user-generated content enables more people to experience freedom of expression than ever before. In addition, users of these platforms have the option of being anonymous and hiding their personal identities, which can increase the chance that they will misuse these technical features. Offensive language online creates an exclusive environment and, in more severe cases, it can foster real-world violence [Sapetal. 2019]. The use of offensive language has become one of the most common problems on social networking platforms. According to a study by the Pew Research Center in 2015, 67% of people in the U.S. agree they should be able to publicly make offensive statements against minority groups [Laub 2019]. Some countries have issued laws to ban hate speech on social networking platforms. For example, in 2017, Germany passed the Network Enforcement Act, a law that requires social media

In the era of social computing, the interaction between individuals becomes more striking, especially through social media platforms and chat forums. Microblogging applications opened up the chance for people worldwide to express and share their thoughts instantaneously and extensively. Driven, on one hand, by the platform's easy access and anonymity. And, on the other hand, by the user's desire to dominate debate, spread / defend opinions or argumentation, and possibly some business incentives, this offered a fertile environment to disseminate aggressive and harmful content. Despite the discrepancy in hate speech legislation from one country to another, it is usually thought to include communications of animosity or disparagement of an individual or a group on account of a group characteristic such as race, color, national origin, sex, disability, religion, or sexual orientation [100].

Benefiting from the variation in national hate speech legislation, the difficulty to set a limit to the constantly evolving cyberspace, the increased need of individuals and societal actors to express their opinions and counter-attacks from opponents and the delay in manual check by internet operators, the propagation of hate speech online has gained new momentum that continuously challenges both policy-makers and research community. With the development in natural language processing (NLP) technology, much research has been done concerning automatic textual hate speech detection in recent years. A couple of renowned competitions (e.g., SemEval-2019[158] and 2020 [159], GermEval-2018 [150]) have held various events to find a better solution for automated hate speech detection. In this regard, researchers have populated large-scale datasets from multiple sources, which fueled research in the field. Many of these studies have also tackled hate speech in several non-English languages and online communities. This led to investigate and contrast various processing pipelines, including the choice of feature set and Machine Learning (ML) methods (e.g., supervised, unsupervised, and semi-supervised), classification algorithms (e.g., Naives Bayes, Linear Regression, Convolution Neural Network (CNN), LSTM, BERT deep learning architectures, and so on).

The limitation of the automatic textual-based approach for efficient detection has been widely acknowledged, which calls for future research in this field. Besides, the variety of technology, application domain, and contextual factors require a constant up-to-date of the advance in this field in order to provide the researcher with a comprehensive and global view in the area of automatic HT detection. Extending existing survey papers in this field, this paper contributes to this goal by providing an updated systematic review of literature of automatic textual hate speech detection with a special focus on machine learning and deep learning technologies.

We frame the problem, its definition and identify methods and resources employed in HT detection. We adopted a systematic approach that critically analyses theoretical aspects and practical resources, such as datasets, methods, existing projects following PRISMA guidelines [90]. In this regards, we have tried to answer the following research questions:

- Q1: What are the specificities among different HS branches and scopes for automatic HS detection from previous literature?
- Q2: What is the state of the deep learning technology in automatic HS detection in practice?
- Q3: What is the state of the HS datasets in practice?
- Q4: How can offensive language be detected in real-time, especially in fast-paced online environments like social media platforms?

- Q5: What are the challenges and opportunities in integrating text-based and non-textual content analysis?
- Q6: How can offensive language detection systems adapt to the constantly evolving nature of language and emerging trends in online communication?
- Q7: What are the challenges in scaling up real-time detection to handle large volumes of user-generated content?

The above-researched questions will examine barriers and scopes for the automatic hate speech detection technology. A systematic review-based approach is conducted to answer Q1 and Q2, where we will try to depict and categorize the existing technology and literature. The third research question Q3, will be answered by critically examining the scope and boundaries of the dataset identified by our literature review, highlighting the characteristics and aspects of the available resources.

This review paper is organized as follows: section 2 will include a brief theoretical definition of HS. Section 4 examines the previously identified review papers of HS detection. Section 5 details the systematic literature review document collection methodology. Section 6 presents the results of this literature review, including the state of deep learning technology. Section 7 emphasizes on the available resources (datasets and open-source projects). After that, in section 8, an extensive discussion is carried out. Finally, we have highlighted future research directions and conclusions at the end of this paper.

3.1 Motivation and Background :

Social media has become one of the most important environments for communication among people. As user-generated content on social media increases significantly, so does the harmful content such as offensive language. Aggressiveness in social media is a problem that especially affects vulnerable groups (Hamm et al., 2015), (Kowalski and Limber, 2013). Within this context, the need for automatic detection of offensive content gains a lot of attraction. Traditional methods to detect offensive language include use of blacklisted keywords and phrases based on profane words, regular expressions, guidelines and human moderators to manually review and detect unwanted content. However, these methods are not sufficient, particularly considering the users that tend to use more obfuscated and implicit expressions. Automatic identification of offensive language is essentially considered as a classification task.

Previous research on the topic include approaches from different perspectives, utilizing different data sets and focusing on various contents such as abusive language (Waseem et al., 2017) (Chu, Jue, and Wang, 2016), hate speech (Davidson et al., 2017) (Schmidt and Wiegand, 2017) (Fortuna and Nunes, 2018) and cyberbullying (Van Hee et al., 2018). Where machine learning approaches are of concern, (Davidson et al., 2017) indicated using certain terms and lexicons are useful. (Zhang, Robinson, and Tepper, 2018) compared different approaches and pointed out that a deep neural network model combining convolutional neural network and long short-term memory network, performed better than state of the art, including classifiers such as SVM.

There are several previous shared tasks similar to offensive language detection. The shared task on Aggression Identification called 'TRAC' provided participants a dataset containing annotated Facebook posts and comments in English and Hindi (Kumar et al., 2018). Aiming to classify the text among three classes including

nonaggressive, covertly aggressive, and overtly aggressive. The best-performing systems in this task used deep learning approaches based on convolutional neural networks (CNN), recurrent neural networks and LSTM (Majumder, Mandl, and others, 2018). The Spanish language has also been considered. For example, in the recent shared task, MEX-A3T 2018, regarding aggression detection in Mexican Spanish; among the methodologies proposed by participants, there were content based (bag of words, word n-grams, dictionary words, slang words etc.) and stylistic-based features (frequencies, punctuations, POS etc.) as well as approaches based on neural networks (CNN, LSTM and others); baselines were outperformed by the most participants (Alvarez- ´ Carmona et al., 2018).

Furthermore, other shared tasks focusing on aggression in other languages include Italian, German (Bosco et al., 2018),(Wiegand, Siegel, and Ruppenhofer, 2018). One of the most recent shared task on the topic is "Categorizing Offensive Language in Social Media" (SemEval 2019 - Task 6) (Zampieri et al., 2019b). Referring to the problem in a hierarchichal scheme including the target type of the offense. To classify offensive text, about 70 % of the participants used deep learning approaches. Among the top-10 teams, seven used BERT (Devlin et al., 2018).

3.2 LITERATURE REVIEW:

A significant increase in the use of offensive language on social media platforms has been observed in recent years (Zampieri et al., 2020). Offensive language and hate speech, which exist at the junctures of various social and political tensions as an expression of conflicts between different groups within and beyond civilisations, comprise a pervasive phenomena on social media (Caselli, Basile, Mitrović, Kartoziya, & Granitzer, 2020). This clearly illustrates how technologies are rife with both opportunities and difficulties. Consequently, some organisations have developed automatic systems that block inappropriate or offensive language from being shown on their platforms (Poletto, Basile, Sanguinetti, Bosco, & Patti, 2021). The research community has extensively examined hostile language over a long period. One of the early studies on this topic (Chen, Zhou, Zhu, & Xu, 2012a) attempted to detect offensive users using lexical syntactic characteristics extracted from their posts. While it established an effective framework for future research, the dataset provided insufficient proof. The authors of a later study (Davidson, Warmsley, Macy, & Weber, 2017) gathered one of the most comprehensive datasets of offensive and hate speech.

To improve abusive language detection in English social media communications, (Felbo, Mislove, Søgaard, Rahwan, & Lehmann, 2017) used the 'deepmoji' technique, which was first announced in 2017. This strategy is primarily based on pretraining a neural network model for offensive language classification using emojis as poorly supervised training labels. A lexical syntactic feature architecture was proposed by (Chen, Zhou, Zhu, & Xu, 2012b) to strike a balance between identifying offensive content and potentially offensive users in social media. The authors argued that the content's source should be emphasised instead of considering messages as separate instances. Xiang & Zhou (2014) used a topic-based mixture model integrated into the framework of semi-supervised training, which was trained on a substantial amount of un-annotated Twitter data to detect offensive tweets. To detect objectionable content on Twitter, the authors of (Xiang, Fan, Wang, Hong, & Rose, 2012) focused on the Twitter social media platform and suggested a semi-supervised strategy in conjunction with statistical subject modelling.

OffensEval 2019 (Zampieri et al., 2019c) and GermEval (Struß et al., 2019) are two large collaborative initiatives centred on offensive language identification. Other

projects involving the identification of offensive speech include HASOC-19 (Mandl et al., 2019), which focused on hate speech and offensive material in Indo-European languages, and TRAC-2020 (Kumar, Ojha, Malmasi, & Zampieri, 2020), which focused on aggression detection in Bangla, Hindi, and English. While HASOC-19 and TRAC 2020 focused on identifying offensive speech in the Indo-Aryan languages of Bangla and Hindi, DravidianLangTech (Chakravarthi et al., 2021a) is the first shared task to focus on identifying offensive speech in Dravidian languages.

Researchers have been working on creating methods to identify inappropriate language in Arabic, Danish, English, Greek, and Turkish (Zampieri et al., 2020), and several other languages for some time. They have identified objectionable language using a variety of methods. Recently, Ranasinghe & Zampieri (2021) reported on the use of the XLM-RoBERTa model for identifying offensive language in Indian languages such as Bengali and Hindi. They demonstrated that the XLM-R model outperforms all existing techniques for detecting inflammatory language. A Marathi offensive language dataset was released by Gaikwad, Ranasinghe, Zampieri, & Homan (2021). However, there has been minimal research into text categorisation in Dravidian languages, and only a few studies have been conducted on offensive speech identification in Dravidian languages. Our work fills a gap in the research on methods for identifying offensive speech in Dravidian languages; moreover, the systems proposed can be extended to other Indian languages as well as foreign languages.

To address offensive language identification in Dravidian languages, several manually annotated datasets for Tamil, Malayalam, and Kannada (Chakravarthi et al., 2021c) were created for sentiment analysis and offensive language identification. In multilingual nations such as India, where the majority of speakers are polyglots, code-mixed utterances are unavoidable, given that the videos were gathered from social media. Chakravarthi et al. (2021a) carried out a collaborative task on offensive language identification in Tamil, Malayalam, and Kannada for user-generated comments. It utilised DravidianCodeMix1, a multilingual code-mixed dataset that has been carefully annotated for sentiment analysis and offensive language identification. Around 44,000 comments in code-mixed Tamil-English, 20,000 comments in Malayalam-English, and 7700 comments in Kannada-English were included in the dataset.

The majority of the works on offensive language identification focus more on model improvements using pre-trained embedding. Dowlagar & Mamidi (2021) used a pre-trained multilingualBidirectional Encoder Representations from Transformers (BERT) transformer model with transliteration and class balancing loss for offensive content identification. Many participants in the shared task used a form of BERT from multilingual BERT, XLM-R, m-BERT, and Indic-BERT (Ghanghor, Chakravarthi, Priyadharshini, Thavareesan, Krishnamurthy, 2021, Huang, Bai, 2021, Li, 2021, Vasantharajan, Thayasivam, 2021, Yasaswini, Puranik, Hande, Priyadharshini, Thavareesan, Chakravarthi, 2021). However, some participants proposed new methods, including (Zhao, 2021), who proposed a system based on the multilingual model XLM-RoBERTa and DPCNN. Chen & Kong (2021) used multilingual BERT and TextCNN for semantic extraction and text classification. ML techniques (LR, SVM), three deep learning techniques (LSTM, LSTM+Attention), and three transformers-based methods (m-BERT, Indic-BERT, XLM-R) were proposed by Sharif, Hossain, & Hoque (2021). Sharma, Kandasamy, & Kandasamy (2021) have used MPNet (Song, Tan, Qin, Lu, & Liu, 2020) on (Zampieri et al., 2019a) dataset which consists of tweets conatining english language text which are being classified as Offensive or Not Offesinve.

In this paper, we propose an approach that aims to address the lack of annotated data for low-resource Dravidian languages, using fusion of MPNet and CNN. First, we examine the performance of existing state-of-the-art baseline classical ML models, Ensemble of ML models for which genetic algorithms are used to select the best among multiple classifiers to enhance the ensemble's performance on the training set, and multilingual language models on these languages. Second, we explain how we may increase our classification performance by utilising MPNet and CNN fusion model and analyze the model further to understand it's working.

4. BACKGROUND :

In this section, we highlighted important concepts related to the domain of our survey. Deciding whether or not a text is offensive can be difficult, owing to the subjective nature of the endeavor. Therefore, we started with some descriptions and definitions of the meaning of offensive language and listed the common types of offensive language that appeared in our survey.

4.1 Offensive Language :

Providing a discrete definition of offensive language is a very complex task. In general, culture and personal experience are two crucial factors in describing what is considered offensive language and what is not [Vandersmissen 2012]. A text that contains some form of abusive behavior, exhibiting actions with the intention of harming others, causing hurt, and making others angry, is known as offensive language [Vandersmissen 2012]. Wiedemann et al. [2018] described offensive language as "threats and discrimination against people, swear words or blunt insults"(p.1). Hate speech, aggressive content, cyberbullying, and toxic comments are all different forms of offensive language [Schmidt and Wiegand 2017]. This abusive behavior could cause disturbance, disrespect, harm, insult, and anger, affecting the harmony of conversations. Moreover, this form of online behavior reduces user trust in the online platform [Founta et al. 2018].

Hate Speech : Text that is targeted towards a group of people, with the intent to cause harm, violence, or social chaos is known as hate speech [Sigurbergsson and Derczynski 2019]. Davidson et al. [Davidson et al. 2017] defined hate speech as language used to express "hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group" (p.1). Common forms of hate speech usually contain racial and homophobic slurs [Davidson et al. 2017]. Fortuna and Nunes [2018] defined a list of rules to identify hate speech, which consists of: using stereotypes to refer to individuals based on the groups to which they belong, using negative statements about minority groups, using racial and disparaging terms to cause harm, using racial and sexist slurs, using language that shows pride in a specific group, supporting entities that encourage hate speech, discriminating based on nationalities or religions, and using language that suggests the superiority of in-group individuals.

Cyberbullying : Generally, cyberbullying is characterized by online harassment against an individual. Cyberbullying can have more severe effects than physical or verbal bullying, owing to the nature of online materials that can spread the harassment faster and make it viewable to wider audience [Dadvar et al. 2013]. Zampieri et al. [2019] defined cyberbullying as targeted insults or threats toward an individual. Hee et al. [2015] provided guidelines for analyzing cyberbullying; they mentioned three

indicators of cyberbullying, including the intention to cause harm, repetitiveness, and an imbalance of power. Haidar, Chamoun, and Serhrouchni [Haidar et al. 2017] defined eight categories for cyberbullying: flaming, masquerade, impersonation, harassment, outing, trickery, exclusion, and cyberstalking.

Violence : Johnston and Weiss [2017] defined violence in terms of terrorist groups (e.g., al-Qaeda and Islamic State of Iraq and Syria ("ISIS")) and extremist groups (e.g., radical leftist ("Antifa"), white supremacist). In Abdelfatah et al. [2017], seven classes of social media violence were mentioned, including: crime, violence, human rights abuse, political opinion, crisis, accidents, and conflict. Figure 1.1 illustrates randomly selected examples of some types of offensive language from the list of datasets used among the surveyed literature.

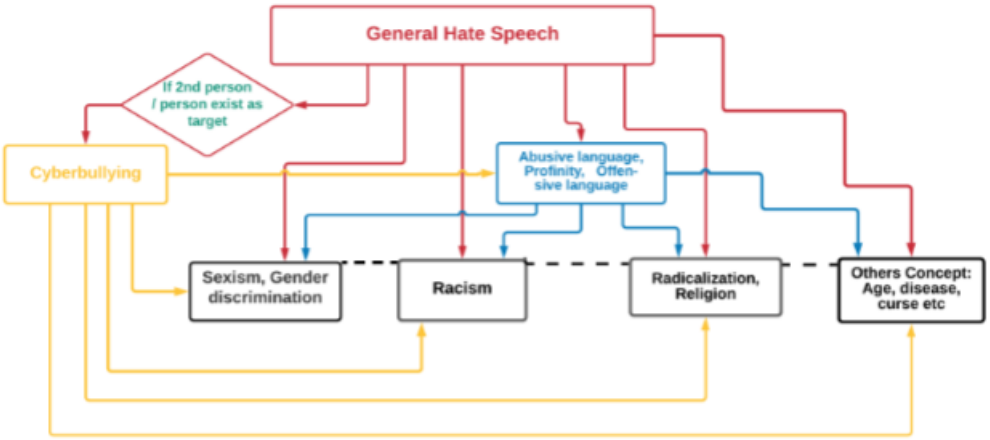

Figure 1.1 : Relational diagram between different type of hate speech concepts

4.2 Tamil language:

Tamil is one of the oldest languages in the world, with a rich literary tradition dating back over 2,000 years. It is primarily spoken in the Indian state of Tamil Nadu and the union territory of Puducherry, as well as in Sri Lanka, Singapore, Malaysia, and other Tamil diaspora communities worldwide.

Key points about Tamil language:

1. Script: Tamil has its own unique script, which is syllabic and consists of 18 consonants and 12 vowels. It is written from left to right.

2. Classical Language: Tamil is classified as a classical language of India due to its ancient literary heritage and history.

3. Literature: Tamil literature includes ancient Sangam poetry, epics like the "Silappathikaram" and "Manimekalai," and contributions to various literary genres.

4. Official Status: Tamil is one of the official languages of India and holds official status in the state of Tamil Nadu and Puducherry.

5. UNESCO Recognition: The United Nations Educational, Scientific, and Cultural Organization (UNESCO) recognizes Tamil as one of the world's classical languages.

6. Diverse Dialects: Tamil has several dialects, with variations in pronunciation and vocabulary across different regions.

7. Influence: Tamil has influenced other South Indian languages, and it has also borrowed words from Sanskrit and other languages over the centuries.

8. Modern Usage: In addition to being spoken, Tamil is used in literature, cinema, music, and other forms of art and media.

9. Global Community: The Tamil diaspora has contributed to the spread of the language and culture in various countries, particularly in Southeast Asia and the West.

Tamil language and culture have a rich and vibrant history, making it an integral part of South Indian and global heritage.

4.3 Kannada Language :

Kannada is a Dravidian language primarily spoken in the Indian state of Karnataka. It's one of the 22 officially recognized languages of India. Kannada has a rich literary tradition with a history dating back over a thousand years. It has its own script, which is an abugida, where each character represents a consonant with an inherent vowel sound that can be modified with diacritics.

Kannada is not only spoken in Karnataka but also by smaller communities in neighboring states and among the Kannadiga diaspora around the world. It has a significant body of classical and modern literature, making it an essential part of Karnataka's cultural heritage. If you have more specific questions or need information on a particular aspect of the Kannada language, please let me know.

4.4 Malayalam Language :

Malayalam is a Dravidian language spoken predominantly in the Indian state of Kerala and in the Union Territory of Lakshadweep. It's one of the 22 officially recognized languages in India. Malayalam has a unique script, which is an abugida, meaning each character represents a consonant with an inherent vowel sound that can be altered using diacritics to indicate different vowel sounds.

Key points about Malayalam:

1. Script: The Malayalam script is derived from ancient Brahmi and has its own distinctive characters. It is written from left to right.

2. History: Malayalam has a rich literary history dating back to the 9th century. It has a vast body of classical literature, including poems and prose, as well as a vibrant contemporary literary tradition.

3. Geographic Distribution: While primarily spoken in Kerala, Malayalam is also used in some neighboring regions and by the Malayali diaspora around the world.

4. Culture: The Malayalam language is deeply embedded in Kerala's culture, and it plays a significant role in the state's arts, films, music, and daily life.

5. Influence: Malayalam has been influenced by Sanskrit and other languages due to its long history and cultural exchanges.

5. RELATED WORKS AND METHODOLOGY:

Authors in [8] applied the standard pre-processing techniques like stemming, and removal of stop-words and unimportant characters from the textual data. They experimented with three classifiers Naive Bayes, SVM and Decision Tree with 10-fold cross validation. They divided their feature space into general features and label specific features. For the general features they used TF-IDF and for the specific features they used unigrams and bigrams.

In [9] the authors trained the model to detect and analyze the cyber-bullying on the social media platforms. They divided their work into 4 subtasks. For the subtask A, after the regular pre-processing techniques they applied three techniques for the featurization namely unigram, unigram+bigram and POS colored unigram+bigram. For the classification task they experimented with Naive Bayes, SVM with linear kernel, SVM with RBF kernel and Logistic Regression with 5 fold cross validation using the WEKA implementation. In subtask B, for the author's role they used the same classifiers as the subtask A with 10-fold cross validation and also tuned the best model jointly with 5-fold cross validation with grid search CV. For categorizing the person mention role into respective categories they used the named entity recognition and trained the linear CRF and SVM respectively with 10-fold cross validation. In subtask C, they used the same feature representation, classifiers and parameter tuning as for the previous 2 subtasks with 10-fold cross validation. They used LDA as well as its variational inference implementation as their exploratory tool to discover the relevant topics from the bullying trace in the subtask D.

Authors in [10] used three feature sets to train the cyber-bullying classifier which were content-based, cyber-bullying based and user-based features. For the pre-processing they removed all the stop words and applied stemming to their dataset. They trained a Support Vector Machine to classify the bullying comments and non-bullying comments with 10-fold cross validation.

To deal with the problem of hate against black community on twitter the authors in [11] trained a Na¨ıve Bayes classifier to able to classify the new tweet as racist or nonracist. They pre-processed the dataset by eliminating the URL's, mentions, stop words and punctuation along with lowercasing and replacing the wrong spellings with the correct ones. Authors found that 86% of the tweets that were racist only because they contained the offensive words so they preferred unigram model to featurize the training data.

Work in [12] was a binary classification task of predicting whether the comment is hateful or not. They followed a pipeline starting from Data collection and annotation, Feature selection, Data pre-processing, Feature preparation and finally Model selection. They realized that the offensive words from the tweet could be the important features so they utilized the frequency of occurring of unigram and bigram. As the offensive tweet contain the certain instances following a particular pattern therefore for the extraction of typed dependencies within the tweet text they employed a Stanford lexical parser along with a context free lexical parsing model which

represented the syntactic grammatical relationship in a sentence that are used a important features for the classifier. They came out with more common sense type of reasoning approach for this feature extraction phase. For the pre-processing phase they followed a generalized pipeline of tokenization, lowercase conversion, removal of stop words and alphanumeric characters, stemming. To preserve the context of words and the surrounding they employed unigrams to trigrams. They experimented with the 2 approaches of n-grams and collection of derogatory or hateful terms to check the contribution of other terms in determining the strong predictors. They ran a Bayesian Logistic Regression using all the typed dependencies features and came up with the vector representation of the tweet containing list of n grams that included words, typed dependencies or combination of both. They used the three classifiers Bayesian Logistic Regression, Random Forest Decision Tree and Support Vector Machine for this binary classification task. They also employed the meta voting ensemble classifier made from these classifiers.

After applying the standard pre-processing techniques, they [13] divided their work into two parts for the detection of hate speech from the user comments. First they employed paragraph2vec to learn the distributed representation of comments and words using the neural language model of the continuous BOW (CBOW). This produced a low dimensional embeddings where the semantically similar comments resided in the same part of the space. Secondly a logistic regression classifier was trained on these embeddings to classify the type of user comment as hateful or clean.

Authors in [14] used the Vowpal Wabbit's regression model to measure the different aspect of the user comments using NLP features. They divided their features into 4 categories which were N-grams, Linguistics, Syntactic and Distributional Semantics. Due to noise found in the data they performed some mild-preprocessing for the first three features but did not performed any normalization for the fourth feature.

In [15], the pre-processing part was undertaken as to convert the tweet into lowercase and performed stemming through the porter stemmer. After that they featurized the tweets as weighted TF-IDF unigrams, bigrams and trigrams followed by the construction of the POS tagging using NLTK. They used Flesch-Kincaid Grade Level and Flesch Reading Ease scores to capture the quality of each tweet and also assigned the sentiment scores to each of the tweet. For the hashtags, mentions, retweets and URL's, they included binary and count indicators and for the number of characters, words and syllables, they included features. They tried various models in Scikit-learn like Logistic regression, Na¨ıve Bayes, Random Forest, Decision Tree and Linear SVM to train the model using 5-fold cross validation along with L1 regularization to reduce the dimensionality of the text data. They also performed the grid search parameter tuning to find the optimal parameters.

Authors in [16] used a LIBLINEAR SVM implementation for this multi-class classification task which has proven to be very effective on Native language identification and temporal text identification. For the features they used character n-grams, word n-grams and word skip-grams.

In [17], the authors defined their topology based on the prior work in the field of detection of different types of abusive language. They considered a 2 fold approach where the first aspect is to analyse the target of the abuse and another aspect is to analyse the degree to which the abuse is explicit. They also laid the implications of this topology on the annotation and the modeling of this problem. They suggested that the data annotation strategies should be dependent on the type of the abuse that is intended to be identified. On the other hand to select the most relevant features for the

modeling, it is important to identify whether the abuse is directed, generalized, explicit or implicit.

In TRAC workshop proceedings [18], there were a total of 30 teams who submitted their systems for English and Hindi Language. Participants applied various techniques like LSTM, CNN, SVM, BiLSTM, Logistic Regression, Random Forest and many more to classify the English and Hindi Facebook comments.

The participants of the shared task of GermEval [20] used tokenization, POS-tagging, lemmatization and stemming and parsing as the methods for tweets pre-processing. They used SVM, Logistic Regression, Na¨ıve Bayes, CNN, LSTM, GRU and the combination for the classification of the tweets.

The authors in [21] compared word embeddings and CNN against the BOW approach with the classifiers such as SVM, Na¨ıve Bayes, k-NN and LDA that were applied on the Document Term Matrix.

Related work as evident in [1] applied various machine learning and deep learning techniques for each of the subtask within the problem domain. They first applied linear SVM trained on word unigrams followed by BiLSTM with the softmax activation function in the final layer with FastText embeddings. Finally they also experimented with CNN on this dataset.

5.1 RESEARCH METHODS:

The scientific study of Tamil offensive language, from an NLP point of view, is recent, and the number of studies in the field is relatively low in comparison to others from different languages.In this survey, each study was considered as a unit of our analysis. We found 35 studies during the process of the survey, which involved six main steps. We provided clear descriptions of these steps to support study replicability. The six steps started with keywords selection, followed by the search for studies, recursive search, filtering the studies, qualitative and quantitative analysis, and synthesizing data. Our steps are illustrated in Figure 1 and explained further in the following paragraphs.

Keywords Selection :

Some keywords were selected during the first step of the survey. The keywords list included “offensive language” and “Tamil NLP,” which had to be present anywhere in text. In addition, we considered terms referring to particular categories of offensive language. Four more optional keywords were used, including “Tamil abusive language,” “Tamil hate speech,” “Tamil violence detection,” and “Tamil Cyberbullying,” at least one of which had to be present anywhere in text.

Search for Studies :

We conducted the survey with the goal of collecting the largest possible number of literature from the domain of our study. Thus, we did not restrict our search to specific databases, rather, we started the search from the commonly used databases, such as the Association for Computing Machinery Digital Library (ACM DL), the Institute of Electrical and Electronics Engineers (IEEE) Xplore, and the Association for Computational Linguistics (ACL). We included more studies from other databases (e.g., SAGE Journal, Springer Link) during the next steps. Moreover, we attempted to

collect data from all available literature. Hence, our literature set included conference publications, workshop papers, conference posters, journal publications, dissertations, theses, and books.

Recursive Search :

We used Research Gate and Google Scholar to retrieve the references listed in literature and to retrieve other literature that cited the original work. Then, we recursively repeated the search with the new literature found. This process helped to ensure that as much literature as possible on the same topic was collected.

Filtering :

To ensure the quality of the data collected, studies were filtered using some inclusion and exclusion criteria before being included into the study. The inclusion criteria were not strict to ensure coverage of research; the survey included different types of studies (e.g., conference publications, workshop papers, journal publications), collected from various resources (e.g., ACM DL, IEEE Xplore, ACL), which covered several categories of offensive language (e.g., hate speech, cyberbullying, violence, abusive language), and involved the Tamil language in their datasets. The exclusion
criteria were needed to ensure quality of research; literature from the NLP domain (e.g., classification systems, linguistic resources, deep learning, machine learning, statistical learning) were included, while literature from other domains were excluded. Examples of excluded domains are computer vision, communication, and public policy. Moreover, literature that did not include any experiments was excluded. For example, the study “Detection of Hate Speech in Social Networks:

A Survey on Multilingual Corpus,” done by Al-Hassan and Al-Dossari [2019], was excluded from the survey, because it did not have an experiment or an implementation of a system. All studies published before June 2020 were evaluated, as there are very limited research and studies in this area.

Qualitative and Quantitative Analysis :

The first step in analyzing the studies was the data extraction task. Before starting with reviewing and analyzing the literature, we defined our data extraction techniques. Tables were created with the main attributes to be extracted as a template for the data extraction step. The extracted attributes included publication year, study goal, dataset size, dataset source, evaluation results, classifiers used, features implemented, and pre processing techniques used. Papers were coded into themes to be categorized based on the targeted offensive category. Studies that covered more than one offensive category were discussed based on the experiments within each offensive category. Hence, they appeared in multiple offensive categories based on their content. Quantitative analysis focused on analyzing literature quantitatively, such as accuracy results obtained and data set size. During the quantitative analysis, we focused on all experiments in each study if the number of the experiments was less than eight, otherwise, we selected the top experiments with best results to analyze. Qualitative analysis included tools and resources used, description of the general approach of studies, and categorization of studies based on the closest offensive form.

Synthesize Data :

After analyzing literature quantitatively and qualitatively, findings were aggregated into graphs and tables to identify limitations in previous research and gaps that need to be addressed. Synthesizing the data helped to highlight the future direction of research in Tamil offensive language detection.

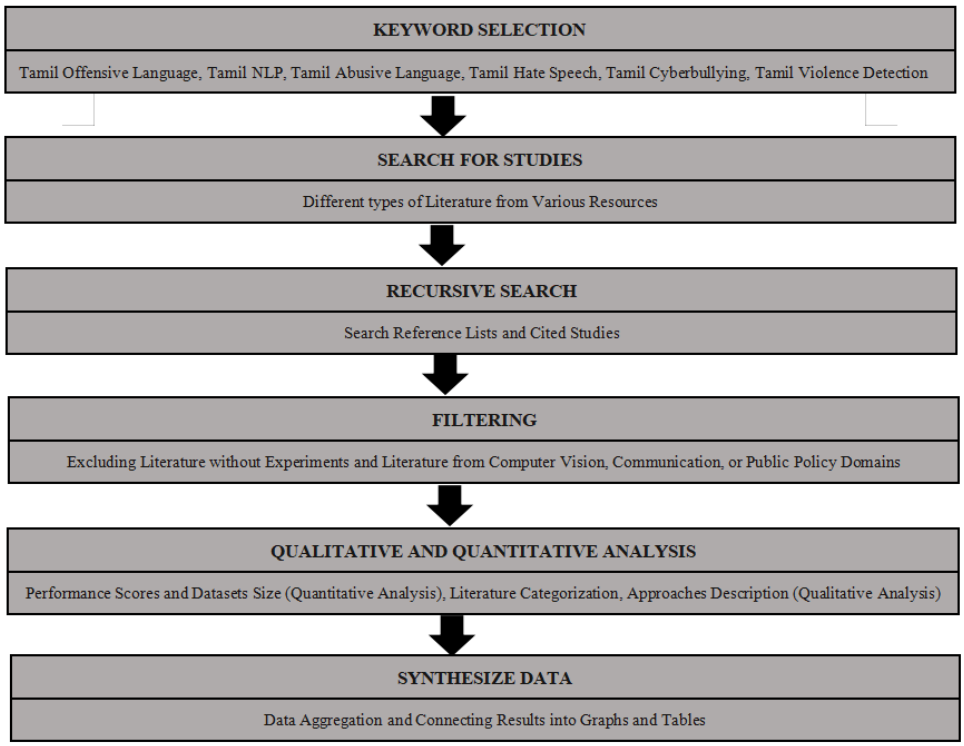

Figure 1.2 : Steps of this survey.

5.2 EXISTING METHODOLOGY :

Support Vector Machine :

Support vector machines (SVMs) are a strong, supervised ML method used mostly for classification problems; however, they may also be used for regression tasks in some cases. The purpose of an SVM is to locate the hyperplane in an N-dimensional space that best distinguishes between the data points. This implies that the decision boundary line between the data points that belong to a given category and those that do not is clearly drawn by this method. A vector may be used to encode any type of data; this is relevant to all vector data types. As a result, if we can generate adequate vector representations of the data in our possession, SVM can be employed to obtain the required outcomes. The input characteristics are the same as in LR, namely the TF-IDF values up to 3 g. L2 regularisation is used to assess the SVM model in this study.

Multinomial naive bayes (MNB) :

The Multinomial Naive Bayes classifier (MNB) is a Bayesian classifier that is based on the naive assumption of the conditional independence of features for performing its classification task. This implies that each input is completely independent from the others, which is impossible when real data is involved. Nonetheless, it simplifies a number of complicated tasks, validating the need for such a tool.

We investigated the performance of a naive Bayes classifier for multinomially distributed data, which is derived from Bayes' theorem and predicts the probability of a future occurrence given an observation of a past event. MNB is a customised variant of naive Bayes that is more suited for text documents than other types of data. Unlike basic naive Bayes, which models a text based on the presence and absence of specific words, MNB explicitly models word counts and alters the underlying computations to account for them. Thus, the incoming text data is seen as a bag of words, which considers only the frequency of the words in the text data and ignores their location.

Decision tree (DT) :

When used in conjunction with a tree structure, the decision tree (DT) may be used to create classification or regression models. At the same time, a dataset is split into smaller and smaller subgroups and an accompanying DT is generated incrementally. The end outcome is a tree with decision nodes and leaf nodes at its nodes. Thus, DT classification involves constructing a tree structure in which each node corresponds to a feature name and each branch relates to the values generated for the feature names. The categorisation labels are represented by the leaves of the tree. Following the selection of possible options in a sequential manner, each node is recursively divided, and eventually, the classifier establishes some rules to predict the outcome. DTs are capable of handling large amounts of data and performing categorisation without extensive processing. DT classifiers are generally considered to be reasonably accurate. Regarding their disadvantages, they are particularly sensitive to errors in classification issues with multiple classes and a fairly small number of training samples. Furthermore, it is computationally expensive to prepare. To identify the optimum split, it must first arrange each potential dividing region at each node before locating the optimal split. Some algorithms utilise combinations of fields, and it is necessary to seek the optimal combination weights before proceeding. Implementing pruning techniques can be time consuming and expensive since it is necessary to form and compare several candidate sub-trees. In this study, for DT, we tuned the hyper-parameters using grid-search, which resulted in Gini being used as the splitting criterion, maximum depth as -1 and minimum sample split of value 2.

Random forest (RF) :

Random forest (RF) is an ensemble classifier that generates predictions using a collection of distinct DTs trained on datasets the same size as the training set, called bootstraps, which are constructed through random resampling on the training set. Once the tree is formed, a collection of bootstraps is used as the test set. These bootstraps exclude any particular record from the original dataset (out-of-bag [OOB] samples). The classification error rate for all test sets is the OOB estimate of the generalisation error. In this study, RF demonstrated significant benefits over other approaches in terms of the capacity to handle extremely non-linearly correlated data, resistance to noise, ease of tuning, and ability to perform effective parallel processing. Additionally, RF has an essential feature: an intrinsic feature selection phase that is performed prior to the classification job to condense the variables' space by assigning a significance value to each feature. RF adheres to precise principles for tree growth, tree combination, self-testing, and post-processing; furthermore, it is resistant to overfitting and is considered more stable in the presence of outliers and in very large parameter spaces as compared to other ML methods. We analysed the RF model using the same criteria as the DT model.

LightGBM :

With the advent of ensemble techniques there are lots of new models that have come up such as XGBoost, AdaBoost, RandomForest and many more. But the main

disadvantage with these models was the unsatisfactory performance in terms of efficiency and scalability when the size of the data given as input to these models were large. This was because for each feature it was required to scan through all the data points to find the best split. Thus a very highly time consuming process. So two avoid this GOSS and EFB techniques were proposed. LightGBM Machado, Karray, & de Sousa (2019) is a Gradient Boosting Decision Tree(GBDT) with GOSS and EFB where it achieves almost the same accuracy of the conventional GBDT wherein the training time was 20 times faster than the conventional GBDT. As a part of this study, we examine LightGBM model by training it on TF-IDF vectors.

BERT :

BERT is a semi-supervised language representation model that employs both left- and right-context conditioning in conjunction with the masked language model training target (Devlin, Chang, Lee, & Toutanova, 2019). These extensive contextual representations may be extended to a classification head to fine-tune BERT's performance on downstream NLP tasks. We classified using BERT in conjunction with the classification head and fine-tuned all parameters end to end. We conducted tests using the Hugging Face library (https://huggingface.co/). BERT overcomes the limitation of previous language models (such as **word2vec** and **GloVe**) in interpreting context and polysemous words. Moreover, it performs well in monolingual and multilingual classifications, resulting in the greatest performance increase in NLP tasks, such as question answering (SQuAD v1.1) and natural language inference (MNLI). In contrast to unidirectional language models, transformer encoders read the whole input word sequence at once. Consequently, BERT is the superior bidirectional transformer as compared to bidirectional LSTMs in terms of accuracy. This behavior causes the transformer model to simultaneously learn the context of a word from left to right and right to left. For training the BERT model, we set the number of epochs to 3 and used an initial learning rate of $2e^{-5}$ with a decay factor of 0.01.

Ensemble Method :

An ensemble of models is a collection of learning models whose individual predictions are integrated in such a way that the component models' flaws are balanced (Rokach, 2010). The premise behind this technique is that various models display varying degrees of inductive bias. If the mistakes generated by these biases are uncorrelated, it is predicted that the models in the ensemble would correct each other's errors, thus reducing the total number of errors when the model results are aggregated. Ensemble approaches have been found to be successful at using this trait to minimise variance error without increasing bias error (Kazmaier & van Vuuren, 2022).

In the independent approach, the general foundation for any ensemble learning is to employ an aggregation function to combine a set of baseline classifiers, $c_1,c_2,\ldots,c_k$, for predicting a single output. Given a dataset of size n and features of dimension m, $D = \{(x_i,y_i)\}$, $1 \leq i \leq n, x_i \in R^m$ the prediction of the output based on this method is given by Eq. (1).

$$y_i = \emptyset(x_i) = G(c_1,c_2,\ldots,c_k)$$

where $y_i \in Z$ denotes classification.

Building an ensemble model, given this general framework, entails deciding how to train the baseline classifiers and finding a suitable process for aggregating the outputs of the baseline classifiers. For their successful improvement regarding predictive accuracy and to ensure they can be easily parallelised in training, several independent

ensemble methods have been proposed over the last few years. Genetic algorithm (GA)-based approaches are common for adjusting the weights of multiple models in an ensemble. For this study, we have used GA for electing the best models for the ensemble model. We noticed that GA dropped DT model from the ensemble which boosted the performance.

6. DATASET:

Online media, for example, Twitter, Facebook or YouTube, contain quickly changing data produced by millions of users that can drastically alter the reputation of an individual or an association. This raises the significance of programmed extraction of sentiments and offensive language used in online social media. YouTube is one of the popular social media platforms in the Indian subcontinent because of the wide range of content available from the platform such as songs, tutorials, product reviews, trailers and so on. YouTube allows users to create content and other users to comment on the content. It allows for more user-generated content in under-resourced languages. Hence, we chose YouTube to extract comments to create our dataset. We chose movie trailers as the topic to collect data because movies are quite popular among the Tamil, Malayalam, and Kannada speaking populace. This increases the chance of getting varied views on one topic. Figure 1 shows the overview of the steps involved in creating our dataset.

We compiled the comments from different film trailers of Tamil, Kannada, and Malayalam languages from YouTube in the year 2019. The comments were gathered using YouTube Comment Scraper tool7 . We utilized these comments to make the datasets for sentiment analysis and offensive language identification with manual annotations. We intended to collect comments that contain code-mixing at various levels of the text, with enough representation for each sentiment and offensive language classes in all three languages. It was a challenging task to extract the necessary text that suited our intent from the comment section, which was further complicated by the presence of remarks in other non-target languages.

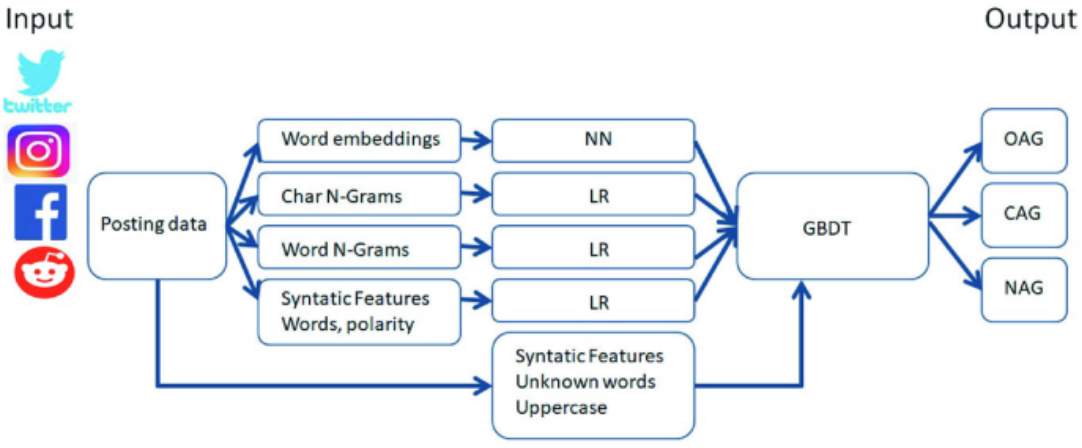

Figure 1 : Data collection process

As a part of the preprocessing steps to clean the data, we utilized langdetect library to tell different languages apart and eliminate the unintended languages. The Langdetect library, however, is a script detection library that filters out languages based on certain scripts. This has serious limitations as it misses out a number of languages written in non-conventional script. This explains why we still get data from other

languages despite using this library. Examples of code-mixing in Tamil, Kannada and Malayalam corpora are shown in Figs. 2, 3, and 4 along with their translations in English. By keeping data privacy in mind, we made sure that all the user-related information is removed from the corpora. As a part of the textpreprocessing, we removed redundant information such as URL.

Code - Switching Type	Example	Translation
No Code - Mixing Only Tamil (Written in Tamil Script)	இது மாதிரி ஒரு படத்தை தான் இத்தனை வருஷமா எதிர்பார்த்து கொண்டிருந்தேன் டிரெய்லர் பார்க்கும்போதே மனசுக்கு அவ்வளவு சந்தோசமா இருக்கு	How long I have been Waiting for this kind of Movie! Feels joyful just Watching this Trailer
Inter - Sentential Code - Mixing Mix of English and Tamil (Tamil Written only in Tamil Script)	ரெட்டியார் சமுகம் சார்பாக படம் வெற்றி பெற வாழ்த்துக்கள்… Mohan G All the Very Best we all behind you.	On behalf of Reddiyar Community, I am Wishing the Best for this Movie. Mohan G All the Very Best. We all behind you.
Only Tamil (Written in Latin Script)	Inga niraya perukku illatha kedda palakkam enkidda irukku. vassol mannan VIJAY anna.	'I have a bad habit which is not found here in others'. Brother Vijay is the king of blockbusters.
Code - Switching at morphological level (Written in both Tamil and Latin Script)	ஓ விஜய் படத்துக்கு இப்படிதான் viewers sekkurangala	Oh. So this is how you gather more viewers for Vijay's movie ?
Intra - Sentential mix of English and Tamil (Written in Latin Script Only)	Patti thotti engum pattaya kelaputha ne va thalaiva I am waiting	Rocking Performance that will be a hit among every type of audience. Come on, my star. I am Waiting
Inter - Sentential and Intra - Sentential Mix (Tamil Written in Both Tamil Script and Latin Script)	இந்த படத்த வர விட கூடாது... இந்த படத்த திரையரங்கில் ஓட விட கூடாதுனு எவனாது தடை பன்னா.. . Theatre la vera endha padam odunaalum screen கிழியும்	If anybody imposes a ban that this movie should not be released, that it should not be allowed to run on theaters, then the screens will be torn if any other movie is released.

Table 1 : Examples of code mixing in Tamil dataset

Comment Type	Example	Translation
Only English	Concentrate on hindi promotion.. sir	Concentrate on hindi promotion.. sir
Only Kannada (Written in Kannada Script Only)	ಉತ್ತಮ ಸಾಹಿತ್ಯ ಮತ್ತು ಸಂಗೀತದ ಗೆಳೆಯರೇ, ನನ್ನ ಮನೆಯಲ್ಲಿ ಎಲ್ಲರೂ ಈ ಹಾಡಿಗೆ ಮನಸೋತಿದ್ದಾರೆ.	Great Lyrics and Music mate, Everyone in my home are obsessed with this song.
Mix of English and Kannada	My Favorite song in 2019 is Taaja	My Favorite song in 2019 is

(Kannada written in Kannada Script Only)	samachara ಸಾಹಿತ್ಯಾಸಕ್ತರು ಕೇಳಿದರೆ ಮತ್ತೆ ಕೇಳಬೇಕೆನಿಸುತ್ತದೆ… Everybody watch this.	Taaja samachara. If it is heard by literary lovers, they would want to hear it again. Everybody watch this.
Only Kannada (Written in English)	Neevu varshkke ondu cinema madru supper 1 varshikke 3-4 cinema madobadalige intha ondu cinema saku.	If you make one movie a year it's super, instead of doing 3-4 movies a year, one movie of this type is enough.
Only Kannada (Written in both Kannada and English Script)	Nanage ಈ ವೀಡಿಯೊವನ್ನು ರಶ್ಮಿಕಾ ಮಂದಣ್ಣ ಅಭಿಮಾನಿಗಳು ಇಷ್ಟಪಡಲಿಲ್ಲ ಎಂದು ಭಾವಿಸುತ್ತಾರೆ.	I Feel that this video has been disliked by the fans of Rashmika Mandana.
Mix of English and Kannada (written in English only)	Wonderful song daily 5/6 kelalill Andre eno miss madakodante.	A Wonderful song, if I don't hear this song 5 - 6 times a day, I feel like I am Missing Something.
Mix of English and Kannada (Kannada Written in both English and Kannada Script)	ರಕ್ಷಿತ್ ಶೆಟ್ಟಿ ಅಭಿನಯದ ಬಗ್ಗೆ ನನಗೆ ಯಾಕೆ ಹುಚ್ಚು ಹಿಡಿದಿದೆಯೋ ಗೊತ್ತಿಲ್ಲ. ನಿಮ್ಮ ಚಲನಚಿತ್ರಕ್ಕಾಗಿ ಕಾಯುತ್ತಿದ್ದೇನೆ, ಅದು ಬ್ಲಾಕ್‌ಬಸ್ಟರ್ ಆಗುವ ನಿರೀಕ್ಷೆಯಿದೆ. All the Best your bright ಭವಿಷ್ಯ	Don't know why I am Obsessed with Rakshit Shetty's acting. waiting for your movie, expecting it to be a blockbuster. All the Best your bright Future.

Table 2 : Examples of code mixing in Kannada dataset

Code - Switching Type	**Example**	**Translation**
Only English	Very Good Movie - Making Skills in your Language.. Keep up the Good Work.	Very Good Movie - Making Skills in your Language.. Keep up the Good Work.
No - Code - Mixing: Only Malayalam (Written in Malayalam Script Only)	സ്ത്രീകൾ ശാപമാണെങ്കിൽ ഇത്തരക്കാരുടെ അമ്മമാർ പുരുഷന്മാരാണോ?	If women are a curse, are the mothers of such people men ?
Inter - sentential code - mixing: Mix of English and Malayalam (Malayalam written in Malayalam Script Only)	High Promising Trailer. ആ പഴയ മഞ്ജു ചേച്ചിയെ തിരികെ കിട്ടിയ പോലെ തോന്നുന്നു.	High Promising Trailer. It feels like got that old Manju sister back.
Only Malayalam (Written in Latin Script)	Ee onathinu nalloru kudumbhachithram pratheekshikkunnu.	Expecting a good family entertainer for this Onam.
Code - Switching in Morphological level (written in both Malayalam and Latin Script)	കുറച്ച് കാലത്തിന് ശേഷം സിദ്ദിഖ് വീണ്ടും Comedy യിത്ത് Sajeevamayi.	After some time Siddique became active again in Comedy.
Intra - Sentential mix of English and Malayalam (Written in Latin Script)	Video song kaanathe unlike adikkunath nallakaaryam alla	It is not fair to unlike a video song without watching the same.

Inter - sentential and Intra - sentential mix. (Malayalam written in both Latin and Malayalam Script	ഈ Success സിനിമയിലൂടെ vichariche. എന്നാൽ nice സുഖമാണ്.Full Comedy with കോമഡിയും Climax pratheekshikunnu.. I am Waiting,	Through this movie will not be a success. But it is nice now. Expecting full comedy and an awesome climax, I am Waiting.

Table 3 : Examples of code mixing in Malayalam dataset

7. RESULT AND ANALYSIS :

The first step in analyzing the studies was the data extraction task. Before starting with reviewing and analyzing the literature, we defined our data extraction techniques. Tables were created with the main attributes to be extracted as a template for the data extraction step. The extracted attributes included publication year, study goal, dataset size, dataset source, evaluation results, classifiers used, features implemented, and pre processing techniques used. Papers were coded into themes to be categorized based on the targeted offensive category. Studies that covered more than one offensive category were discussed based on the experiments within each offensive category. Hence, they appeared in multiple offensive categories based on their content. Quantitative analysis focused on analyzing literature quantitatively, such as accuracy results obtained and data set size. During the quantitative analysis, we focused on all experiments in each study if the number of the experiments was less than eight, otherwise, we selected the top experiments with best results to analyze. Qualitative analysis included tools and resources used, description of the general approach of studies, and categorization of studies based on the closest offensive form.

We used the offensive language data from DravidianCodeMix (Chakravarthi et al., 2021c). The dataset consists of numerous code-mixed comments on Tamil, Malayalam, and Kannada movie trailers on YouTube. The dataset is separated into training, development, and test sets, all of which have a comparable distribution across the three languages. The Malayalam dataset has five distinct labels, whereas the Tamil and Kannada datasets have six distinct labels, including the 'Offensive-Targeted-Insult-Other' category. We detected a massive class imbalance in the dataset, with 'Not-Offensive' accounting for the majority of the sample data and 'Offensive-Targeted-Insult-Other' accounting for a small portion of the sample data for all three languages.

The corpora statistics of DravidianCodemix is tabulated in Table 1. The class-wise distribution of the training and test set are tabulated in Table 2 and Table 3 as given by DravidianLangTech organizers. The distribution the dataset Chakravarthi et al. (2021 a) was created by adapting the work of (Zampieri et al., 2019b). The organizers of DravidianLangTech 2022 reduced the three-level hierarchical annotation scheme into a flat scheme with five labels to account for the types of offensiveness in the comments (Table 4).

Not-Offensive (NO): "Not offensive" refers to content, language, or behavior that does not cause discomfort, insult, or harm to individuals or groups. It adheres to social norms and values, promoting a respectful and inclusive environment. It avoids provoking negative emotions, and its intent is to maintain a positive, non-confrontational atmosphere.

Offensive-Targeted-Insult-Individual (OTI): "Offensive targeted insult individual" refers to a deliberate act of using hurtful or derogatory language, actions, or content with the specific aim of demeaning, belittling, or harming a particular person. It often involves personal attacks or discriminatory behavior meant to insult and degrade an individual, causing emotional or psychological harm.

Offensive-Targeted-Insult-Group (OTG): "Offensive targeted insult group" refers to the deliberate use of derogatory language, actions, or content aimed at demeaning or harming a particular collective or community. It involves prejudiced or discriminatory behavior intended to insult and degrade a specific group, causing emotional or psychological harm.

Offensive-Targeted-Insult-Other (OTO): "Offensive targeted insult other" describes a deliberate act of using derogatory language, actions, or content with the specific intent to demean or harm an entity or category that doesn't fit into individual or group classifications. This behavior often involves personal attacks or discriminatory actions meant to insult and degrade, leading to emotional or psychological harm.

Offensive-Untargeted (OU): "Offensive untargeted" refers to content, language, or behavior that is hurtful, inappropriate, or objectionable but is not directed at a specific individual, group, or category. It may involve general offensive content, such as profanity or crude humor, without singling out anyone.

Not in indented language: "Not" is a negation word used to indicate the absence or negation of something. It is employed to express the opposite or the inverse of a condition or statement. For example, "not happy" means lacking happiness or being unhappy. "Not" is a fundamental term in logic and language for expressing negation.

LANGUAGE	TAMIL	MALAYALAM	KANNADA
Number of Words	511,734	202,134	65,702
Vocabulary Size	94,772	40,729	20,796
Number of Comments	43,919	20,010	7771
Number of Sentences	52,617	23,652	8586
Average Number of Words per Sentence	11	10	8
Average Number of Sentences per Comment	1	1	1

Table 1 : Corpus statistics for Offensive Language Identification.

CLASS	TAMIL	MALAYALAM	KANNADA
Not Offensive	31,808 (72.42%)	17,697 (88.44%)	4336 (55.79%)
O - Untargeted	3630 (8.26%)	240 (1.19%)	278 (3.57%)
O - Targeted Individual	2965 (6.75%)	290 (1.44%)	628 (8.08%)
O - Targeted Group	3140 (7.14%)	176 (0.87%)	418 (5.37%)
O - Targeted Others	590 (1.34%)	-	153 (1.96%)
Not in Indented Language	1786 (4.06%)	1607 (8.03%)	1898 (24.42%)
Total	43,919	20,010	7771

Table 2 : Offensive language Identification Dataset Distribution. O-Offensive. O-Untargeted: Offensive Untargeted.

	TAMIL	MALAYALAM	KANNADA
Training	35,139	16,010	6217
Development	4388	1999	777
Test	4392	2001	777
Total	43,919	20,010	7771

Table 3 : Train-Development-Test Data Distribution with 90%-5%-5% train-dev-test for Offensive Language Identification provided by DravidianLangTech shared task organizers.

	TAMIL	MALAYALAM	KANNADA
Training	33,685	14,721	4695
Development	4216	1836	586
Test	4232	1844	592
Total	42,133	18,401	5873

Table 4 : Train-Development-Test Data Distribution for Offensive Language Identification after removing not-indented language labels.

CLASSIFIER	SHORT NAME
Random Forest	RF
Support Vector Machine	SVM
Multinomial Naive Bayes	MNB
Decision Tree	DT
Light Gradient Boosting Tree	LGBM
Ensemble With Decision Tree	EWDT
Ensemble Without Decision Tree	EWODT
Bidirectional Encoder Representations from Transformers	BERT
Masked and Permuted Network	MPNet
Convolutional Neural Network	CNN

Table 5 : Classification models used for comparison.

7.1 Qualitative Analysis :

Ref	Preprocessing	Features	Dataset	Classifier	Accuracy	F1	Recall	Precision
[Alakrot et al. 2018a]	Stemming, removing stopwords and non-alphabetic characters, normalizing letters, replacing Persian	Word-level features	YouTube, 15,050 comments	SVM	90.05%	82%	77%	88%
[Mohaoucha	Adopt Alakrot, Murray,	AraVec trained on	YouTube,	CNN, BiLSTM,	87.84% CNN, 86.42% Bi-	84.05% CNN, 82.33% Bi LSTM,	82.24% CNN, 80.97% Bi LSTM, 81.51%	86.10% CNN, 83.74% Bi LSTM,

ne et al. 2019]	and Nikolov [Alakrot et al. 2018a] approaches	Twitter dataset and skip-gram model	15,050 comments	BiLSTM with attention mechanis, and CNN LSTM	LSTM, 85.75% Bi-LSTM with attention, 87.27% CNN-LSTM	81.70% BiLSTM with attention, 83.65% CNN-LSTM	BiLSTM with attention, 83.46% CNN-LSTM	85.75% BiLSTM with attention, 83.89% CNN-LSTM
[Djandji et al. 2020]	Tokenization, removing usersmention, retweet, URL, diacritics emojis and newline, replacing underscore in hashtag with white spaces	N/A	Twitter, 10,000 tweets	AraBERT	N/A	90.15%	N/A	N/A
[Husain 2020b]	Converting emoji and emoticon to text, letter normalization, replacing repeated letter with one, dialect normalization, converting hyponym to hypernym, replacing underscore in hashtag with white space, removing numbers, HTML tags, more than one space, symbols, stopwords, and diacritics	Count vectorizer	Twitter, 10,000 tweets	SVM	90.2%	89.8%	90.2%	89.9%
[Husain 2020a]	Adopt Husain [Husain 2020b] approach	Count vectorizer and TF-IDF	Twitter, 10,000 tweets	SVM, LR, J48, bagging ,random forest, and Adaboost	N/A	82% SVM, 81% LR, 69% J48, 88% bagging, 87% random forest, and 86% Adaboost	N/A	N/A

Table 1 : Qualitative Analysis of General Offensive or Abusive Language Detection Literature

CONCLUSION :

While several studies have been published on offensive language identification, there is considerable room for experimentation. New insights into low-resource Dravidian languages may result in more efficient and accurate models. Furthermore, to the best of our knowledge, this is the first study to propose an exhaustive examination of models, and newly proposed fusion of MPNet and CNN model used for offensive language identification in three Dravidian languages. As demonstrated in this paper, such models perform admirably on a particular language but do not generalise effectively.It can be deduced that the application of deep neural networks to identify offensive language in Dravidian languages is promising. In the future, we will examine better neural network architectures apart from CNN and pre-trained embedding.

REFERENCES:

1. Chakravarthi BR, Priyadharshini R, Jose N, Kumar MA, Mandl T, Kumaresan PK, Ponnusamy RRLH, McCrae JP, Sherly E. Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada. In: Proceedings of the first workshop on speech and language technologies for Dravidian languages. Kyiv: Association for Computational Linguistics; 2021,p. 133–45. Accessed 15 May 2021.

2. Vigna FD, Cimino A, Dell'Orletta F, Petrocchi M, Tesconi M. Hate me, hate me not: hate speech detection on Facebook. In: ITASEC; 2017.

3. Chakravarthi BR, Muralidaran V, Priyadharshini R, McCrae JP. Corpus creation for sentiment analysis in code-mixed Tamil-English Text. In: Proceedings of the 1st joint workshop on spoken language technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL). European Language Resources association, Marseille, France; 2020, p. 202–10. Accessed 28 Apr 2021.

4. Suryawanshi S, Chakravarthi BR, Arcan M, Buitelaar P. Multimodal Meme Dataset (MultiOFF) for Identifying Offensive Content in Image and Text. In: Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying. European Language Resources Association (ELRA), Marseille, France; 2020, p. 32–41.Accessed 15 May 2021.

5. Jose N, Chakravarthi BR, Suryawanshi S, Sherly E, McCrae JP. A Survey of Current Datasets for Code-Switching Research. In: 2020 6th international conference on advanced computing and communication systems (ICACCS); 2020, p. 136–41.

6. Dave B, Bhat S, Majumder P. IRNLP_DAIICT@DravidianLangTech-EACL2021:Offensive Language identification in Dravidian Languages using TF-IDF Char N-grams and MuRIL. In: Proceedings of the first workshop on speech and language technologies for Dravidian languages. Kyiv: Association for Computational Linguistics; 2021 , pp. 266–9. Accessed 10 May 2021.

7. Chakravarthi BR. HopeEDI: a multilingual hope speech detection dataset for equality, diversity, and inclusion. In: Proceedings of the third workshop on

computational modeling of people's opinions, personality, and emotion's in social media. Association for Computational Linguistics, Barcelona, Spain (Online); 2020, p. 41–53. Accessed 11 May 2021.

8. Kumar R, Ojha AK, Lahiri B, Zampieri M, Malmasi S, Murdock V, Kadar D (editors): Proceedings of the second workshop on trolling, aggression and cyberbullying. European Language Resources Association (ELRA), Marseille, France; 2020. Accessed 25 Apr 2021.

9. Chakravarthi BR, Arcan M, McCrae JP. Improving wordnets for under-resourced languages using machine translation. In: Proceedings of the 9th Global Wordnet Conference. Global Wordnet Association, Nanyang Technological University (NTU), Singapore; 2018, p. 77–86. Accessed 29 Apr 2021.

10. Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers). Association for Computational Linguistics, Minneapolis, Minnesota; 2019, p. 4171–86. Accessed 15 Apr 2021.

11. Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, Grave E, Ott M, Zettlemoyer L, Stoyanov V. Unsupervised Cross-lingual Representation Learning at Scale; 2019. arXiv preprint

12. Dadvar M, Trieschnigg D, Ordelman R, de Jong F. Improving Cyberbullying Detection with User Context; 2013, p. 693–6 .

13. Kalaivani A, Thenmozhi D. SSN_NLP_MLRG at SemEval-2020 Task 12: Offensive Language Identification in English, Danish, Greek Using BERT and Machine Learning Approach. In: Proceedings of the fourteenth workshop on semantic evaluation. International Committee for Computational Linguistics, Barcelona (online); 2020, p. 2161–70. Accessed 15 Apr 2021.

14. Hande A, Priyadharshini R, Chakravarthi BR. KanCMD: Kannada CodeMixed Dataset for Sentiment Analysis and Offensive Language Detection. In: Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media. Association for Computational Linguistics, Barcelona, Spain (Online); 2020, p. 54–63. Accessed 20 Apr 2021.

15. Aroyehun S.T., Gelbukh A. 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-2018). Association for Computational Linguistics, Santa Fe, New Mexico, USA; 2018, p. 90–7. Accessed 16 Apr 2021.

16. Malmasi S, Zampieri M. Detecting Hate Speech in Social Media. In: Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017. INCOMA Ltd., Varna, Bulgaria; 2017, p. 467–72.

17. Pitenis Z, Zampieri M, Ranasinghe T. Offensive language identification in Greek. In: Proceedings of the 12th language resources and evaluation conference. European Language Resources Association, Marseille, France; 2020, p. 5113–9. Accessed 18 Apr 2021.

18. Hettiarachchi H, Ranasinghe T. Emoji Powered Capsule Network to Detect Type and Target of Offensive Posts in Social Media. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019). INCOMA Ltd., Varna, Bulgaria; 2019, p. 474–80. Accessed 17 Apr 2021.

19. Chakravarthi BR, Priyadharshini R, Muralidaran V, Suryawanshi S, Jose N, Sherly E, McCrae JP. Overview of the Track on Sentiment Analysis for Dravidian Languages in Code-Mixed Text. In: Forum for Information Retrieval Evaluation, FIRE 2020. Association for Computing Machinery, New York, NY, USA; 2020, p. 21–4 .

20. Sanh V, Debut L, Chaumond J, Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter; 2019. arXiv e-prints

21. Vasantharajan C, Thayasivam U. Hypers@DravidianLangTech-EACL2021: Offensive language identification in Dravidian code-mixed YouTube Comments and Posts. In: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages. Kyiv: Association for Computational Linguistics; 2021, p. 195–202 . Accessed 12 May 2021.

22. Ranasinghe T, Gupte S, Zampieri M, Nwogu I. WLV-RIT at HASOC-Dravidian-CodeMix-FIRE2020: Offensive Language Identification in Code-switched YouTube Comments. In: FIRE; 2020.

23. Arora G. Gauravarora@HASOC-Dravidian-CodeMix-FIRE2020: Pre-training ULMFiT on Synthetically Generated Code-Mixed Data for Hate Speech Detection. In: FIRE; 2020.

24. Howard J, Ruder S. Universal Language Model Fine-tuning for Text Classification; 2018. arXiv e-prints

25. Zampieri M, Malmasi S, Nakov P, Rosenthal S, Farra N, Kumar R. 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, Vol. 1 (long and short papers). Association for Computational Linguistics, Minneapolis, Minnesota; 2019, pp. 1415–20. Accessed 08 May 2021.

26. Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. In Proceedings of the Eleventh International Conference on Weblogs and Social Media (ICWSM).

27. Waseem, Z., & Hovy, D. (2016). Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter. In Proceedings of the NAACL Student Research Workshop (NAACL-SRW).

28. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., & Chang, Y. (2016). Abusive Language Detection in Online User Content. In Proceedings of the International Conference on Social Informatics.

29. Ribeiro, F. N., Santos, J., Macdonald, C., & Jhaver, S. (2017). Like Sheep Among Wolves: Characterizing Deceptive Opinion Spam on Twitter. In Proceedings of the 26th International Conference on World Wide Web (WWW).

30. Founta, A. M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., ... & Kourtellis, N. (2018). Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior. In Proceedings of the 2018 World Wide Web Conference (WWW).

31. Salminen, J., Ojala, J., Jung, S. G., & An, J. (2018). From Offensive Language to Hate Speech: A Multilingual Overview of the State of the Art. In Proceedings of the 27th International Conference on Computational Linguistics (COLING).

32. Hosseinmardi, H., & Davison, B. D. (2015). Filtering and Evaluating Hate Speech in Microblogs. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).

33. Waseem, Z., & Davidson, T. (2018). Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter. In Proceedings of the First Workshop on Design and Analysis of Experiments on NLP.

34. Chatzakou, D., Kourtellis, N., Blackburn, J., & Vakali, A. (2017). Mean Birds: Detecting Aggression and Bullying on Twitter. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM).

35. Schmidt, A., & Wiegand, M. (2017). A Survey on Hate Speech Detection Using Natural Language Processing. In Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media.

36. Pavlopoulos, J., Malakasiotis, P., Androutsopoulos, I., & Paliouras, G. (2017). Deeper Attention to Abusive User Content Detection. In Proceedings of the 2nd Workshop on Abusive Language Online (ALW2).

37. Waseem, Z., & Hovy, D. (2016). Attention is All You Need for Hate Speech Detection. In Proceedings of the 27th International Conference on Computational Linguistics (COLING).

38. Djuric, N., Zhou, J., Morris, R. R., Grbovic, M., & Radosavljevic, V. (2015). Hate Speech Detection with Comment Embeddings. In Proceedings of the 24th International Conference on World Wide Web (WWW).

39. Davidson, T., & Warmsley, D. (2018). Black is to Criminal as Caucasian is to Police: Detecting and Removing Multiclass Biases in Word Embeddings. In Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society.

40. Wulczyn, E., Thain, N., & Dixon, L. (2017). Ex Machina: Personal Attacks Seen at Scale. In Proceedings of the 26th International Conference on World Wide Web (WWW).

41. Park, K., Lee, S., & Kim, J. (2018). A Hierarchical Transformer-based Model for Multi-class Classification of Abusive Language. In Proceedings of the 2nd Workshop on Abusive Language Online (ALW2).

42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. In Advances in Neural Information Processing Systems (NeurIPS).

43. Hosseini, H., & Sirjani, M. (2018). Detecting Cyberbullying on Social Media: A Machine Learning Approach. In Proceedings of the 9th ACM Multimedia Systems Conference.

44. Wulczyn, E., Dixon, L., & Thain, N. (2017). Ex Machina: Personal Attacks Seen at Scale. In Proceedings of the 26th International Conference on World Wide Web (WWW).

45. Mubarak, H., & Darwish, K. (2017). Abusive Language Detection on Arabic Social Media. In Proceedings of the 2nd Workshop on Abusive Language Online (ALW2).

46. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., & Chang, Y. (2016). Abusive Language Detection in Online User Content. In Proceedings of the International Conference on Social Informatics.

47. Zhang, Q., Irani, D., Wobbrock, J. O., & Kim, J. (2018). Reducing Online Hate in a Peer-driven Marketplace: A Multidisciplinary Approach. In Proceedings of the ACM on Human-Computer Interaction.

48. Badjatiya, P., Gupta, S., Gupta, M., & Varma, V. (2017). Deep Learning for Hate Speech Detection in Tweets. In Proceedings of the 26th International Conference on World Wide Web (WWW).

49. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., & Chang, Y. (2016). Abusive Language Detection in Online User Content. In Proceedings of the International Conference on Social Informatics.

50. Qian, J., Bethard, S., & Jurafsky, D. (2018). Assigning Emotional Labels to Debate Transcripts. In Proceedings of the 2nd Workshop on Abusive Language Online (ALW2).