SSNCSE_NLP@TamilNLP-ACL2022: Transformer based approach for detection of abusive comment for Tamil language

Josephine Varsha & B. Bharathi
Department of CSE
Sri Siva Subramaniya Nadar College of Engineering
Kalavakkam - 603110
josephine2010350@ssn.edu.in
bharathib@ssn.edu.in

Abstract

Social media platforms along with many other public forums on the Internet have shown a significant rise in the cases of abusive behavior such as Misogynism, Misandry, Homophobia, and Cyberbullying. To tackle these concerns, technologies are being developed and applied, as it is a tedious and time-consuming task to identify, report and block these offenders. Our task was to automate the process of identifying abusive comments and classify them into appropriate categories. The datasets provided by the DravidianLangTech@ACL2022 organizers were a code-mixed form of Tamil text. We trained the datasets using pre-trained transformer models such as BERT,m-BERT, and XLNET and achieved a weighted average of F1 scores of 0.96 for Tamil-English code mixed text and 0.59 for Tamil text.

1 Introduction

Abusive comment detection is the method of categorizing and detecting the user-generated offensive comments to any type of insult, vulgarity, or profanity that debases the target Schmidt and Wiegand (2017). Over the last decade, there has been an exponential growth of user-generated content on social media. Given this increase in usage of online platforms, technology must be leveraged in the detection of abusive comments, cyber-bullying, hate speech, and trolling. Social media companies have utilized multiple resources to censor comments demeaning others (Chakravarthi et al., 2021a,b, 2020a; Priyadharshini et al., 2020; Chakravarthi, 2020). It's nearly impossible to succeed at perfecting the detector, as a comment's tendency to be abusive depends on the thread of the previous comments (B and A, 2021a). Its subjectivity to the individual and its context-dependent characteristics has been one of the major reasons for its failure. This task aims to train these models to identify abusive language that directly targets an individual or a group without bias.

Our team SSN_CSE_NLP has participated in the shared task of Abusive comment detection. To this effect, we were provided with datasets for code mixed Tamil text comprising comments from YouTube. This poses several challenges due to the low availability of resources for the Tamil language. The task focused on the multilingual offensive language detection, categorization of offensive language, and target identification Kumaresan et al. (2021); Priyadharshini et al. (2022). We have used pre-trained machine learning transformers like BERT,m-BERT, and XLNET. In this paper, we investigate the efficacy of different learning models in detecting abusive languages. We then compare the F1-Score of the different transformer models for both datasets.

Tamil is one of the world's longest-surviving classical languages (Anita and Subalalitha, 2019b,a; Subalalitha and Poovammal, 2018; Subalalitha, 2019). According to A. K. Ramanujan, it is ”the only language of modern India that is recognizably continuous with a classical history.” Because of the range and quality of ancient Tamil literature, it has been referred to as ”one of the world's major classical traditions and literatures.” For about 2600 years, there has been a recorded Tamil literature. The earliest period of Tamil literature, known as Sangam literature, is said to have lasted from from 600 BC to AD 300 (Sakuntharaj and Mahesan, 2021, 2017,?, 2016). Among Dravidian languages, it possesses the oldest existing literature. The earliest epigraphic documents discovered on rock edicts and ”hero stones” date from the 6th century BC (Thavareesan and Mahesan, 2019, 2020a,b, 2021).

The remainder of the paper is organized into 5 sections. Section 2 discusses the related works in the field of Artificial Intelligence, on abusive com-

Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages, pages 158 - 164
May 26, 2022 ©2022 Association for Computational Linguistics

ment detection, for both Tamil and other languages. The methodology proposed for the model along with the models implemented are elaborately explained in the 3rd section of this paper. In section 4 the results and the observations are discussed. Section 5 concludes the paper.

2 Related works

A lot of research is being done on detecting offensive language from social media platforms in the field of Artificial Intelligence and Natural Language Processing(Priyadharshini et al., 2021; Chakravarthi and Muralidaran, 2021; Chakravarthi et al., 2020b; Sampath et al., 2022; Ravikiran et al., 2022; Chakravarthi et al., 2022; Bharathi et al., 2022; Priyadharshini et al., 2022). In this section, we will be reviewing the research work.

The authors of (Vasantharajan and Thayasivam, 2021) has analyzed various techniques and neural network models to detect offensive language in code-mixed romanized social media text in Tamil. They have implemented selective translation and transliteration for text conversion in romanized and code-mixed settings, and are positive that this can be extended to romanized and code-mixed contexts of other languages.

The authors of B and A (2021b) identified offensive language content of the code-mixed dataset of comments/posts in Dravidian Languages (Tamil-English, Malayalam-English, and Kannada-English) collected from social media. The basic TFIDF and count vectorizer features were found to perform best when compared to sentence embeddings. They detected that machine learning models are giving better performance than deep learning models.

A large transliterated Bengali corpus Sazzed (2021) introduced, consisting of 3000 comments collected from YouTube, which are manually annotated into abusive and non-abusive categories. The comparative performances of various supervised ML and deep learning-based classifiers are given, and BiLSTM, the deep learning-based architecture, obtains a relatively lower F1 score compared to LR and SVM, which could be attributed to the small size (i.e.,3000 comments) of the corpus.

The authors of Hande et al. (2021) emphasized on improving offensive language identification by prioritizing the construction of a bigger dataset and generating pseudo-labels on the transliterated dataset, combining the latter with the former to have extensive amounts of data for training.

Language	Training	Development	Test
Tamil	2240	560	700
Tamil-English	5948	1488	1859

Table 1: Dataset description

A three-level classification system with Naive Bayes classifier in the first level, Multinomial Updatable Naive Bayes in the second level, and a rule-based classifier named DTNB in the third level is used in Pang et al. (2002).

The authors Zampieri et al. (2020) reported the lexical features, static and deep contextualized embedding for the Support Vector Machine classifiers to detect Arabic offensive language and also determined the topics, dialects, and genders which are associated with the offensive tweets.

The addition of the sentiment and contextual features provide significantly improved performance to a basic TFIDF model in Yin et al. (2009).

The authors of Mishra et al. (2019)proposed an approach based on graph convolutional networks to show that author profiles that directly capture the linguistic behavior of authors along with the structural traits of their community significantly advance the current state of the art.

The authors of Pitsilis et al. (2018) shows an approach that outperforms all other models and has achieved better performance in classifying short messages. The approach taken did not rely on pre-trained vectors, which provides a serious advantage when dealing with short messages.

3 Methodology and Data pre-processing

In this section, we have illustrated our implementation of the pre-trained machine learning transformer models in detail. Further, we will investigate the performance of the various transformer models in the coming sections. The architecture of the proposed model and the steps are given below 1.

The dataset provided by the LT-EDI 2021 Priyadharshini et al. (2022) for the Tamil, and code-mixed Tamil text consisted of 3499, and 9293 Youtube comments respectively. The details are given in Table 1.

3.1 Data-set Analysis

The goal of this task is to identify whether a given comment contains an abusive comment. A com-

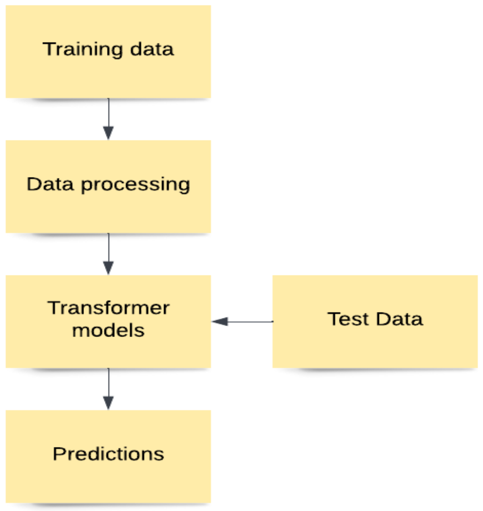

Figure 1: The architecture of the proposed system

ment or post within the corpus may contain more than one sentence but the average sentence length of the corpora is 1. The annotations in the corpus are made at a comment/post level in Priyadharshini et al. (2022)

The dataset provided by LT-EDI 2021 organizers, consisted of the training set, development set, and test set of 2240, 560, and 699 instances respectively for the Tamil text, and 5948, 1488, and 1857 instances for the code-mixed text. It contained text sequences that include user utterances along with the context, followed by the offensive class label. The task was to classify and label them under any of the following: Misogyny, Misandry, Homophobia, Transphobia, Xenophobia, Counter Speech, Hope Speech.

3.2 Data Pre-processing

Data pre-processing is essential for any machine learning problem since the real-world data generally contains noise, and missing values, and may be in an unusable format that cannot be directly used for machine learning models. Data preprocessing is required for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model. First, the dataset is cleaned and processed before classifying. During pre-processing :

- Hashtags, HTML tags, mentions and URLs are removed
- Annotate emojis, emoticons, and replace them with the text they represent
- Convert uppercase characters to lowercase
- To expand abbreviations
- Remove special characters
- Remove accented characters
- Reduce lengthened words
- Remove extra white spaces

We've implemented data processing with the use of the nltk package, abbreviated as the Natural Language Toolkit, built to work with the NLP (Natural Language Processing). It provides various text processing libraries for classification, tokenization, parsing, semantic reasoning, etc. For our model, we've only used the regular expression (re) module. The re. sub() function was used to clean and scrape the text, remove URLs, remove numbers, and remove tags.

Using tokenize. regexp() module we were able to extract the tokens from the string by using the regular expression with the RegexpTokenizer() method. Tokenizing is a crucial step when it comes to cleaning the text. It is used to split the text into words or sentences, splitting it into smaller pieces that still hold its meaning outside the context of the rest of the text. When it comes to analyzing the text, we need to tokenize by word and tokenize by sentence. This is how unstructured data is turned into structured data, which is easier to analyze.

3.3 Model Description

The abusive comment text was classified using 3 transformer models, namely BERT, XLNet, and m-BERT

- BERT:
  BERT stands for Bidirectional Encoder Representations from Transformers. BERT is a pre-trained model on the top 104 languages of the world on Wikipedia (2.5B words) with 110 thousand shared word piece vocabulary, using masked language modeling (MLM) objective, which was first introduced in Devlin et al. (2018). BERT uses bi-directional learning to gain context of words from left to right context simultaneously. This is optimized by the Masked Language Modelling. The MLM

is different from the traditional recurrent neural networks (RNNs), which generally see the word one after the other. This model randomly masks 15% of the words in the input and predicts the masked words when the entire masked sentence is run through the model.

- XLNet:
  The XLNet transformer model was proposed in 'XLNet: Generalized Autoregressive Pretraining for Language Understanding' Yang et al. (2019). It is pre-trained using an autoregressive model (a model that predicts future behavior based on past behavior) which enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and overcomes the limitations of BERT thanks to its autoregressive formulation Yang et al. (2019). It integrates the Transformer-XL mechanism with a slight improvement in the language modeling approach.

- m-BERT:
  m-BERT is a pre-trained model on a large corpus of multilingual data It is trained on the top 104 languages with the largest Wikipedia using a masked language modeling (MLM) objective. It was first introduced in Devlin et al. (2018)

4 Results and Analysis

The BERT (Bidirectional Encoder Representations from Transformers) models and XLNET were used for the Task A dataset. The BERT model operates on the principle of an attention mechanism to learn contextual relations between words. The transformer encoder used is bidirectional, unlike the other directional methods which read input sequentially. BERT reads the entire sequence of text at once. This bidirectional property of the encoder has made it very useful for classification tasks. The BERT models BERT and m-BERT were trained for 5 epochs. XLNet does not suffer from pre-train fine-tune discrepancy since it does not depend on data corruption. We have trained the XLNet model for 5 epochs. The bert-base-uncased model showed the best F1-Score of 0.96 and 0.59 for code mixed Tamil text and Tamil dataset respectively.

4.1 Tamil-English Dataset

The accuracy obtained by the BERT model was found to be 0.96, XLNet and m-BERT showed an accuracy of 0.88, and 0.95 respectively. The bert-base-uncased model (BERT) showed the best performance with a weighted F1 score of 0.96. The weighted precision, weighted recall, weighted F1 score, and accuracy are given in the table below 2.

Pre-trained model	Precision	Recall	F1-score	Accuracy
bert-base-uncased	0.96	0.96	0.96	0.96
xlnet-base-cased	0.87	0.88	0.87	0.88
bert-base-mulitingual-uncased	0.96	0.95	0.95	0.96

Table 2: Performance analysis of the proposed system using development data for Tamil-English Dataset

Pre-trained model	Precision	Recall	F1-score	Accuracy
bert-base-uncased	0.56	0.65	0.59	0.65
xlnet-base-cased	0.46	0.61	0.48	0.61
bert-base-mulitingual-uncased	0.56	0.64	0.59	0.64

Table 3: Performance analysis of the proposed system using development data for Tamil Dataset

4.2 Tamil Dataset

The accuracy obtained by the BERT model was found to be 0.65, XLNet and m-BERT showed an accuracy of 0.61, and 0.64 respectively. The bert-base-uncased model (BERT) showed the best performance with a weighted F1 score of 0.59. The weighted precision, weighted recall, weighted F1 score, and accuracy are given in the Table 3.

The development dataset was used for evaluating the performance of the models after training them. The final performance results for the task are recorded in Table 4.

4.3 Error analysis

The adopted model fails to attain a perfect F1 score of 1. To investigate and analyze this, we have plotted the confusion matrix for the code-mixed Tamil dataset, and the Tamil dataset. The Fig.2 shows the confusion matrix of the code-mixed dataset. This is an 8 X 8 matrix that evaluates the performance of the BERT model, where 8 is the number of target classes. The Fig.3 shows the

Results	Tamil-English	Tamil
Accuracy	0.530	0.060
macro average Precision	0.260	0.130
macro average Recall	0.240	0.140
macro average F1score	0.250	0.090
weighted average Precision	0.510	0.040
weighted average Recall	0.530	0.060
weighted average F1score	0.520	0.030

Table 4: Performance analysis of the proposed system using test data for Tamil and Tamil-English Dataset

confusion matrix of the Tamil dataset. This is a 9 X 9 matrix that evaluates the performance of the BERT model, where 8 is the number of target classes. With the confusion matrix, it is possible to compute the performance metrics of the classification model, namely, Precision, Recall, and F1-score

Precision refers to the number of True Positives (TP) to the total number of predictions

$$Precision = \frac{TP}{TP + FP}$$

Recall refers to the number of Positives returned by the model.

$$Recall = \frac{TP}{TP + FN}$$

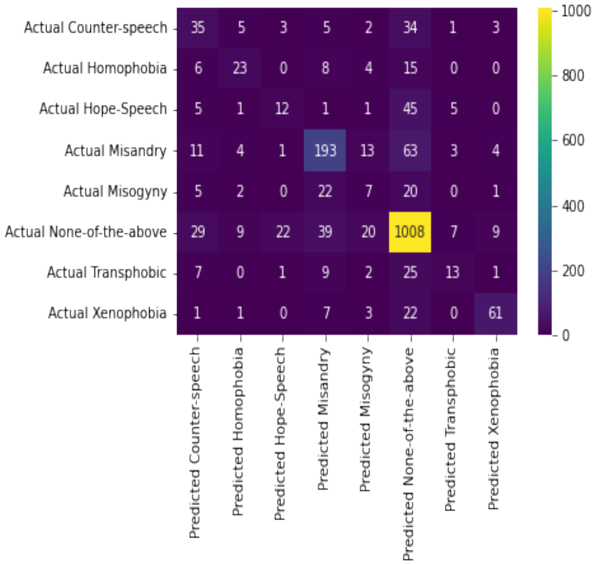

Figure 2: Confusion matrix of tamil-english dataset

5 Conclusions

In this paper, we have investigated the baseline accuracy of different models as well as their variants on the test datasets. There is an increase in demand for abusive language identification on social media, and the goal of this task was to detect whether the comment contains abusive language or not. Our team had secured the 8th rank in the shared Task for the code-mixed Tamil dataset, and the 12th rank for the Tamil dataset. Our models performed the baseline for both the tasks but performance can further be improved by adopting favorable features.

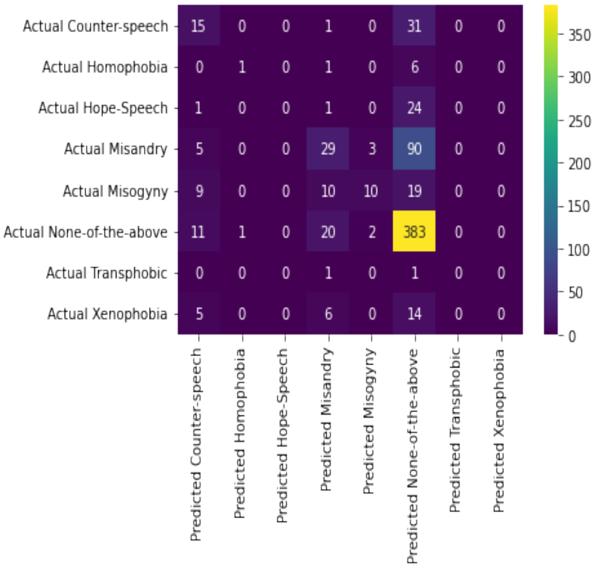

Figure 3: Confusion matrix of tamil dataset

References

R Anita and CN Subalalitha. 2019a. An approach to cluster tamil literatures using discourse connectives. In *2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP)*, pages 1–4. IEEE.

R Anita and CN Subalalitha. 2019b. Building discourse parser for thirukkural. In *Proceedings of the 16th International Conference on Natural Language Processing*, pages 18–25.

Bharathi B and Agnusimmaculate Silvia A. 2021a. SSNCSE_NLP@DravidianLangTech-EACL2021: Meme classification for Tamil using machine learning approach. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 336–339, Kyiv. Association for Computational Linguistics.

Bharathi B and Agnusimmaculate Silvia A. 2021b. SSNCSE_NLP@DravidianLangTech-EACL2021: Offensive language identification on multilingual code mixing text. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 313–318, Kyiv. Association for Computational Linguistics.

B Bharathi, Bharathi Raja Chakravarthi, Subalalitha Chinnaudayar Navaneethakrishnan, N Sripriya, Arunaggiri Pandian, and Swetha Valli. 2022. Findings of the shared task on Speech Recognition for Vulnerable Individuals in Tamil. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi. 2020. 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, pages 41–53, Barcelona, Spain (Online). Association for Computational Linguistics.

Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. 2021. Findings of the shared task on hope speech detection for equality, diversity, and inclusion. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 61–72, Kyiv. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020a. 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)*, pages 202–210, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Thenmozhi Durairaj, John Phillip McCrae, Paul Buitaleer, Prasanna Kumar Kumaresan, and Rahul Ponnusamy. 2022. Findings of the shared task on Homophobia Transphobia Detection in Social Media Comments. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Vigneshwaran Muralidaran, Shardul Suryawanshi, Navya Jose, Elizabeth Sherly, and John P McCrae. 2020b. Overview of the track on sentiment analysis for Dravidian languages in code-mixed text. In *Forum for Information Retrieval Evaluation*, pages 21–24.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Kayalvizhi Sampath, Durairaj Thenmozhi, Sathiyaraj Thangasamy, Rajendran Nallathambi, and John Phillip McCrae. 2021a. Dataset for identification of homophobia and transophobia in multilingual YouTube comments. *arXiv preprint arXiv:2109.00227*.

Bharathi Raja Chakravarthi, Priya Rani, Mihael Arcan, and John P McCrae. 2021b. A survey of orthographic information in machine translation. *SN Computer Science*, 2(4):1–19.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Adeep Hande, Karthik Puranik, Konthala Yasaswini, Ruba Priyadharshini, Sajeetha Thavareesan, Anbukkarasi Sampath, Kogilavani Shanmugavadivel, Durairaj Thenmozhi, and Bharathi Raja Chakravarthi. 2021. Offensive language identification in low-resourced code-mixed dravidian languages using pseudo-labeling. *arXiv preprint arXiv:2108.12177*.

Prasanna Kumar Kumaresan, Ratnasingam Sakuntharaj, Sajeetha Thavareesan, Subalalitha Navaneethakrishnan, Anand Kumar Madasamy, Bharathi Raja Chakravarthi, and John P McCrae. 2021. Findings of shared task on offensive language identification in Tamil and Malayalam. In *Forum for Information Retrieval Evaluation*, pages 16–18.

Pushkar Mishra, Marco Del Tredici, Helen Yannakoudakis, and Ekaterina Shutova. 2019. Abusive language detection with graph convolutional networks. *arXiv preprint arXiv:1904.04073*.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. *arXiv preprint cs/0205070*.

Georgios K Pitsilis, Heri Ramampiaro, and Helge Langseth. 2018. Detecting offensive language in tweets using deep learning. *arXiv preprint arXiv:1801.04433*.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Subalalitha Chinnaudayar Navaneethakrishnan, Thenmozhi Durairaj, Malliga Subramanian, Kogilavani Shanmugavadivel, Siddhanth U Hegde, and Prasanna Kumar Kumaresan. 2022. Findings of the shared task on Abusive Comment Detection in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Sajeetha Thavareesan, Dhivya Chinnappa, Durairaj Thenmozhi, and Rahul Ponnusamy. 2021. Overview of the DravidianCodeMix 2021 shared task on sentiment detection in Tamil, Malayalam, and Kannada. In *Forum for Information Retrieval Evaluation*, pages 4–6.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Mani Vegupatti, and John P McCrae. 2020. Named entity recognition for code-mixed Indian corpus using meta embedding. In *2020 6th international conference on advanced computing and communication systems (ICACCS)*, pages 68–72. IEEE.

Manikandan Ravikiran, Bharathi Raja Chakravarthi, Anand Kumar Madasamy, Sangeetha Sivanesan, Ratnavel Rajalakshmi, Sajeetha Thavareesan, Rahul Ponnusamy, and Shankar Mahadevan. 2022. Findings of the shared task on Offensive Span Identification in code-mixed Tamil-English comments. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2016. A novel hybrid approach to detect and correct spelling in tamil text. In *2016 IEEE International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 1–6.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2017. Use of a novel hash-table for speeding-up suggestions for misspelt tamil words. In *2017 IEEE International Conference on Industrial and Information Systems (ICIIS)*, pages 1–5.

Ratnasingam Sakuntharaj and Sinnathamby Mahesan. 2021. Missing word detection and correction based on context of tamil sentences using n-grams. In *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 42–47.

Anbukkarasi Sampath, Thenmozhi Durairaj, Bharathi Raja Chakravarthi, Ruba Priyadharshini, Subalalitha Chinnaudayar Navaneethakrishnan, Kogilavani Shanmugavadivel, Sajeetha Thavareesan, Sathiyaraj Thangasamy, Parameswari Krishnamurthy, Adeep Hande, Sean Benhur, and Santhiya Ponnusamy, Kishor Kumar Pandiyan. 2022. Findings of the shared task on Emotion Analysis in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Salim Sazzed. 2021. Abusive content detection in transliterated bengali-english social media corpus. In *CALCS*.

Anna Schmidt and Michael Wiegand. 2017. Proceedings of the fifth international workshop on natural language processing for social media. Association for Computational Linguistics.

C. N. Subalalitha. 2019. Information extraction framework for kurunthogai. *Sādhanā*, 44(7):156.

CN Subalalitha and E Poovammal. 2018. Automatic bilingual dictionary construction for tirukural. *Applied Artificial Intelligence*, 32(6):558–567.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2019. Sentiment analysis in tamil texts: A study on machine learning techniques and feature representation. In *2019 14th Conference on Industrial and Information Systems (ICIIS)*, pages 320–325.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020a. Sentiment lexicon expansion using word2vec and fasttext for sentiment prediction in tamil texts. In *2020 Moratuwa Engineering Research Conference (MERCon)*, pages 272–276.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2020b. Word embedding-based part of speech tagging in tamil texts. In *2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*, pages 478–482.

Sajeetha Thavareesan and Sinnathamby Mahesan. 2021. Sentiment analysis in tamil texts using k-means and k-nearest neighbour. In *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, pages 48–53.

Charangan Vasantharajan and Uthayasanker Thayasivam. 2021. Towards offensive language identification for tamil code-mixed YouTube comments and posts. *SN Computer Science*, 3(1).

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

Dawei Yin, Zhenzhen Xue, Liangjie Hong, Brian D Davison, April Kontostathis, and Lynne Edwards. 2009. Detection of harassment on web 2.0. *Proceedings of the Content Analysis in the WEB*, 2:1–7.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). *arXiv preprint arXiv:2006.07235*.