IIITK@LT-EDI-EACL2021: Hope Speech Detection for Equality, Diversity, and Inclusion in Tamil , Malayalam and English

Nikhil Kumar Ghanghor [1], **Rahul Ponnusamy** [2], **Prasanna Kumar Kumaresan** [2]
Ruba Priyadharshini [3] **Sajeetha Thavareesan** [4] **Bharathi Raja Chakravarthi** [5]
[1] Indian Institute of Information Technology Kottayam
[2]Indian Institute of Information Technology and Management Kerala
[3] ULTRA Arts and Science College, Madurai, India, [4]Eastern University, Sri Lanka
[5] National University of Ireland Galway
nkghanghor18bcs@iiitkottayam.ac.in

Abstract

This paper describes the IIITK's team submissions to the hope speech detection for equality, diversity and inclusion in Dravidian languages shared task organized by LT-EDI 2021 workshop@EACL 2021. We have used the transformer-based pretrained models along with the customized versions of those models with custom loss functions. Our best configurations for the shared tasks achieve weighted F1 scores of 0.60 for Tamil, 0.83 for Malayalam, and 0.93 for English. We have secured ranks of 4, 3, 2 in Tamil, Malayalam and English respectively. We have open-sourced our code implementations for all the models across both the tasks on GitHub [1].

1 Introduction

According to Wikipedia hope is being defined as *an optimistic state of mind that is based on the expectation of outcomes with respect to events and circumstances in one's life or the world at large*. The hope speech detection shared task [2] organized by LT-EDI aimed to detect hope speeches in the given corpus for English, Tamil, and Malayalam (Chakravarthi and Muralidaran, 2021). The data set has been gathered from some social media remarks. We participated in this task given a social media remarks in hope speech, frameworks need to characterize if a post is hope speech or not.

Tamil (ISO 639-3: tam) and Malayalam (ISO 639-3: tam) belong to same family. Tamil was the first to be listed as a classical language of India, one of 22 scheduled languages in the Constitution of India, also official language of Tamil Nadu, Puducherry, Singapore and Sri Lanka and is one of the world's longest-surviving classical languages (Norman, 1977; Stein, 1977; Hart III, 2015). The oldest epigraphic documents discovered date from about the 6th century BC on pottery, rock edicts and hero blocks. Over 55 percent of the epigraphic inscriptions discovered by the Archaeological Survey of India (about 55,000) are in the Tamil language (Maloney, 1970; Abraham, 2003). A Tamil prayer book in ancient Tamil script called Thambiran Vanakkam was written by Portuguese Christian missionaries in 1578, thereby rendering Tamil the first Indian language to be printed and published (Balachandran, 2005). Malayalam split from Tamil during 16th century by Thunchaththu Ramanujan Ezhuthachan until then it was west coast dialect of Tamil (Menon, 1938; Steever, 1998).

Over time various methodologies are being proposed by the researchers throughout the Natural Language Processing (NLP) community for building better textual analysis systems. Solving the text classification problem has been improvised throughout by building better architectures and better representation techniques for texts. The community has also benefitted by borrowing the ideas from other domains like computer vision and incorporating those in these systems which have given promising results. Initially, the models used to deal with the Bag Of Words (BOW) representations, then came the ideas of lemmatization and stemming, which helped in improving the representation techniques further. Then around the early 2010s word embeddings were proposed by (Mikolov et al., 2013).

The NLP domain has also observed many architectural innovations which have further pushed the performances to give state of the art (SOTA) results. Some of them are the LSTMs (Hochreiter and Schmidhuber, 1997), BiLSTMs (Ghaeini et al., 2018), GRUs (Chung et al., 2014) and then the mighty transformers (Vaswani et al., 2017). The introduction of transformers changed the entire land-

[1]https://github.com/nikhil6041/HopeSpeechDetection
[2]https://competitions.codalab.org/competitions/27653

Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, pages 197–203
April 19, 2021.

scape, and the models built upon the transformer architecture are consistently pushing the results on the GLUE (Wang et al., 2019) benchmarks. There have been instances where the researchers have tried to incorporate the architectural innovations from different domains to NLP. In this paper, we have tried several architectures built upon the transformer architecture and have fine-tuned them on our task, details of which are being discussed in the later sections of the paper.

2 Related Work

Hope speech detection is a relatively new field and an active area of research in the NLP domain. With the rise of the Internet and the social media platforms, people from various places around the globe are now connected through these platforms which have given them a common place to express their views. These views can often be specifically targeted to a particular person or community that can convey either of a positive, neutral or negative emotion to the concerned person or community. This makes it an important aspect to have systems that can automatically classify these content and filter out the ones having a negative impact on the society. In other words, this also means that we have systems that explicitly detect positive content and help it stay in the social good system. As defined earlier, hope speech can also be considered a piece of text conveying a positive sentiment to the reader of it. One of the first works on hope speech detection is done by Chakravarthi (2020a), Puranik et al. (2021), and Palakodety et al. (2020) . Palakodety et al. (2020) used the polyglot word embeddings to have clusters of texts that conveys similar sentiments and obtained promising results. Hope speech detection can also be considered as the opposite task of hate speech detection.

There has been a significant amount of work done for the hate speech detection task (Mandl et al., 2020; Chakravarthi et al., 2020c; Yasaswini et al., 2021; Ghanghor et al., 2021; Hegde et al., 2021). It has even been a part of several conferences like SemEval [3] as challenges. However, these conferences mainly focused on datasets which were constructed for resource abundant languages. However, in the mid-2020s several competitions have been organized which centred around these under-resourced languages. To build a system that performs well on under-resourced languages like Dravidian languages, several researchers have developed systems that have given noticeable results on these tasks (Hande et al., 2020; Chakravarthi, 2020b; Chakravarthi et al., 2020d,b,a). HASOC-Dravidian-CodeMix-FIRE2020 participants used traditional ml methods like Naive Bayes Classifier, Support Vector Machines (SVMs) and Random Forest along with the pretrained transformers models like XLM-Roberta (XLMR) (Conneau et al., 2020) and BERT (Devlin et al., 2019) for the offensive content identification in code-mixed datasets (Tamil-English and Malayalam-English). (Arora, 2020) at HASOC-Dravidian-CodeMix-FIRE2020 used ULMFit (Howard and Ruder, 2018) to pretrain on a synthetically generated code-mixed dataset and then fine-tuned it to the downstream tasks of text classification.

[3] https://semeval.github.io/

3 Dataset Description

Distribution	Tamil	Malayalam	English
Train	16,160	8,564	22,762
Dev	2,018	1,070	2,843
Test	2,020	1,071	2,846

Table 1: Hope Speech EDI Dataset

The competition organizers have provided us with datasets (Chakravarthi, 2020a) for three different languages Tamil, Malayalam and English. Across each dataset we had three different classes Hope_Speech, Non_Hope_Speech and not_**lang** where **lang** can be either of Tamil, Malayalam or English depending upon the dataset we are dealing with. The train, dev and test set distributions of the dataset are as shown in the table 1.

4 Methodology

The analysis of the nature of texts has been one of the central tasks in NLP. Textual analysis can be defined as a separation of the texts into different classes based upon the underlying meaning they convey (Priyadharshini et al., 2020). The NLP domain has observed many advancements for solving the textual analysis problem. However, it remains an unsolved problem because of the linguistic diversity worldwide and the difficulties in expressing texts to a suitable format for feeding it into the textual analysis systems (Jose et al., 2020). Over time, various methods have been proposed for representation of texts, ranging from Bag of Words, TF-IDF to word embeddings. The word embeddings were introduced with the Word2Vec model, which gives a vectorized representation for a word. After the introduction of the Word2Vec word embedding model, different word embedding techniques have been proposed throughout the NLP domain such as Glove (Pennington et al., 2014), Doc2Vec (Le and Mikolov, 2014), Fasttext (Bojanowski et al., 2017). Using these different kinds of word representations, various different models have been proposed for solving the textual analysis problem. These models consisted of the primitive machine learning models like Naive Bayes (NB), Logistic Regression (LR), Multinomial Naive Bayes (MNB), Support Vector Machines (SVMs). Apart from these models, various models based upon neural networks were also being used such as LSTMs, Bidirectional LSTMs, GRUs. However, the current State Of The Art (SOTA) models are based upon the transformer architecture. There are numerous models built upon the transformer architecture which were being trained on large corpora of texts and are available for fine-tuning to different downstream tasks like textual classification, question answering. These models based upon the transformer architecture uses their tokenizers for the conversion of texts into embeddings which are based upon their own vocabularies. One major problem faced with these models built upon the transformer architecture is that they are only available for high resourced languages like English, German, and Chinese. To use these models for under-resourced languages, the researchers came up with the idea of cross-lingual transfer learning, which means training a model on a high resourced language and then fine-tuning it on a downstream task. A separate benchmark known as XNLI (Conneau et al., 2018) was being made to evaluate the model's performances across multiple languages.

The competition organizers have used the SVMs, MNBs , Decision Trees and other machine learning models as the baseline models for the given datasets. So, we went with using the models built upon the transformer architecture while approaching the problem. We have used the hugging face [4] transformers library for our implementations and used the original versions of the models as well as their customized versions with different loss functions. We have used multilingual-cased BERT (mBERT-cased), XLM-Roberta (XLMR), IndicBERT (Kakwani et al., 2020) , BERT-base-cased (BERT-cased) and BERT-base-uncased (BERT-uncased) models for our implementations. Pertaining to the large size of mBERT-cased and XLMR models we have used their customized versions as well by freezing the original model and stacking a fully connected layer of 512 neurons with a final layer having the same number of neurons as the number of our output classes. With this customized versions, we have used two different loss functions the Negative Log Likelihood (NLL) loss function with class weights, and the Sadice (Li et al., 2020) Loss function both of which were used to handle the data imbalance in the datasets. A pictorial representation of our customized architecture can be seen in the figure 1. Out of all the models mentioned above, mBERT-cased, XLMR, and IndicBERT are multilingual models, and BERT-cased and BERT-uncased models are monolingual models.

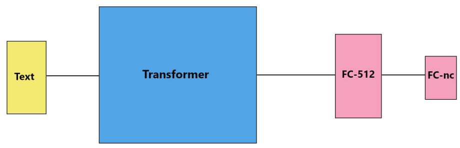

Figure 1: **Custom architecture :** We defined our custom architecture apart from the original transformer models as being built upon the transformer models as the base unit. The output attention heads from the transformer layers are further connected to a 512 neuron fully connected (FC) layer which is finally connected to another fully connected layer having number of neurons same as the number of classes denoted by nc

5 Results and Discussion

We have tried several different combinations of models discussed in the section 4 across the datasets of each language and have reported our results on the development set and the test set in the Table 2 and Table 3 respectively. The results are being reported in terms of weighted F1 scores as it was the evaluation measure being used by the competition organizers. We have used mBERT-cased, XLMR and IndicBERT as our models common across all the three datasets.

The original versions of these models as well as their customized versions, were also being used for finetuning purposes over the datasets. Attributing to the huge model size of mBERT-cased and

[4] https://huggingface.co/transformers/

Model	Tamil	Malayalam	English
mBERT-cased	0.62	0.84	0.93
Custom mBERT-cased with NLL loss and Class weights	0.52	0.63	0.87
Custom mBERT-cased with Sadice Loss	0.50	0.65	0.84
XLMR	0.62	0.83	0.93
Custom XLMR with NLL loss and Class weights	0.33	0.62	0.86
Custom XLMR with Sadice Loss	0.33	0.62	0.86
IndicBERT	0.59	0.82	0.92
BERT-cased	-	-	0.92
BERT-uncased	-	-	0.92

Table 2: Experiments with development dataset (in terms of weighted F1 scores)

Model	Tamil	Malayalam	English
mBERT-cased	**0.60**	**0.83**	**0.93**
Custom mBERT-cased with NLL loss and Class weights	0.50	0.65	0.87
Custom mBERT-cased with Sadice Loss	0.50	0.64	0.85
XLMR	0.59	0.59	**0.93**
Custom XLMR with NLL loss and Class weights	0.30	0.61	0.87
Custom XLMR with Sadice Loss	0.30	0.62	0.87
IndicBERT	0.55	0.72	0.92
BERT-cased	-	-	**0.93**
BERT-uncased	-	-	0.92

Table 3: Experiments with test dataset (in terms of weighted F1 scores)

XLMR we have also tried out their customized versions with the NLL loss and Sadice loss functions. Since the IndicBERT model is comparatively smaller than the other models, its original version was only being considered. Apart from these multilingual models two different monolingual models for the English dataset were also considered. Out of all the models, the original versions outperformed the customized versions of the models.

For the Tamil dataset XLMR, mBERT-cased and IndicBERT gave similar results on the development dataset. However, mBERT-cased gave comparatively better performances than XLMR and IndicBERT on the test dataset. For the Malayalam dataset, XLMR, mBERT-cased and IndicBERT had almost equivalent performances on the development dataset, but mBERT-cased gave much better results on the test set. Surprisingly, XLMR performed even worse than the IndicBERT model on the test set for Malayalam. For the English dataset, apart from the XLMR, IndicBERT, mBERT-cased the BERT-cased and BERT-uncased versions were also being tried and almost each model performed equivalently on the development dataset as well as the test datasets. The superior performance of mBERT over the other two models can be attributed to the training strategy of mBERT. It employs zero-shot cross-lingual model transfer, in which task-specific annotations in one language are used to fine-tune the model for evaluation in another language. A brief explaination of the multilingual nature of mBERT is being discussed in (Pires et al., 2019). On the other hand XLMR although being trained over much more data and having the same training strategy as (Liu et al., 2019) was expected to perform better across the multilingual tasks but it hasn't. We hypothesize the reason behind the degradation in it's performance can be attributed to the code-mixed nature of our dataset in hand. Since the XLMR model was being trained over the CommonCrawl data it could be possible that the data being utilised for the pretraining had very fewer instances of code-mixed data which thus leads to an overall inferior performance as compared to other models. The customized versions of these models were expected to address the skewness of the dataset but failed to do so. When inspecting these model's performances the reason for this performance degradation turned out to be the freezening of the base layers of the transformer models. The performance of these custom models can be further improved by having unfreezed layers which can further increase the performance of these models and can be considered for the future works on this task.

6 Conclusion

We have presented the IIITK team's approach for the hope speech detection shared task organized by DravidianLangTech. Our approach consisted of using the existing pretrained models and finetuning their original as well as the custom versions on the datasets. Out of all the models, the mBERT-cased model gave the best results for the Tamil and Malayalam datasets as 0.60 and 0.83 weighted F1 scores. For the English dataset, mBERT-cased and BERT-cased gave exactly similar results of 0.93 weighted F1 scores.

References

Shinu A Abraham. 2003. Chera, Chola, Pandya: Using archaeological evidence to identify the Tamil kingdoms of early historic South India. *Asian Perspectives*, pages 207–223.

Gaurav Arora. 2020. Gauravarora@hasoc-dravidian-codemix-fire2020: Pre-training ulmfit on synthetically generated code-mixed data for hate speech detection.

R Balachandran. 2005. Pioneers of tamil literature: Transition to modernity. *Indian Literature*, 49(2 (226):179–184.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information.

Bharathi Raja Chakravarthi. 2020a. 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. 2020b. *Leveraging orthographic information to improve machine translation of under-resourced languages*. Ph.D. thesis, NUI Galway.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John Philip McCrae. 2020a. A sentiment analysis dataset for code-mixed Malayalam-English. 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 177–184, Marseille, France. European Language Resources association.

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*. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020b. 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, Vigneshwaran Muralidaran, Shardul Suryawanshi, Navya Jose, Elizabeth Sherly, and John P. McCrae. 2020c. Overview of the Track on Sentiment Analysis for Dravidian Languages in Code-Mixed Text. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 21–24, New York, NY, USA. Association for Computing Machinery.

Bharathi Raja Chakravarthi, Navaneethan Rajasekaran, Mihael Arcan, Kevin McGuinness, Noel E. O'Connor, and John P. McCrae. 2020d. Bilingual lexicon induction across orthographically-distinct under-resourced Dravidian languages. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 57–69, Barcelona, Spain (Online). International Committee on Computational Linguistics (ICCL).

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Reza Ghaeini, Sadid A. Hasan, Vivek Datla, Joey Liu, Kathy Lee, Ashequl Qadir, Yuan Ling, Aaditya Prakash, Xiaoli Z. Fern, and Oladimeji Farri. 2018. Dr-bilstm: Dependent reading bidirectional lstm for natural language inference. Cite arxiv:1802.05577Comment: 18 pages, Accepted as a long paper at NAACL HLT 2018.

Nikhil Kumar Ghanghor, Parameswari Krishnamurthy, Sajeetha Thavareesan, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2021. IIITK@DravidianLangTech-EACL2021: Offensive Language Identification and Meme Classification in Tamil, Malayalam and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, Online. Association for Computational Linguistics.

Adeep Hande, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2020. 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*, pages 54–63, Barcelona, Spain (Online). Association for Computational Linguistics.

George L Hart III. 2015. *Poets of the Tamil anthologies: Ancient poems of love and war*. Princeton University Press.

Siddhanth U Hegde, Adeep Hande, Ruba Priyadharshini, Sajeetha Thavareesan, and Bharathi Raja Chakravarthi. 2021. UVCE-IIITT@DravidianLangTech-EACL2021: Tamil Troll Meme Classification: You need to Pay more Attention. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification.

Navya Jose, Bharathi Raja Chakravarthi, Shardul Suryawanshi, Elizabeth Sherly, and John P. McCrae. 2020. A Survey of Current Datasets for Code-Switching Research. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 136–141.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced nlp tasks.

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

Clarence Maloney. 1970. The beginnings of civilization in South India. *The Journal of Asian Studies*, pages 603–616.

Thomas Mandl, Sandip Modha, Anand Kumar M, and Bharathi Raja Chakravarthi. 2020. Overview of the HASOC Track at FIRE 2020: Hate Speech and Offensive Language Identification in Tamil, Malayalam, Hindi, English and German. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 29–32, New York, NY, USA. Association for Computing Machinery.

Chelnat Achyuta Menon. 1938. *Ezuttaccan and his age*. Ph.D. thesis, SOAS University of London.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

KR Norman. 1977. The Poems of Ancient Tamil: Their Milieu and Their Sanskrit Counterparts.

Shriphani Palakodety, Ashiqur R. KhudaBukhsh, and Jaime G. Carbonell. 2020. Hope speech detection: A computational analysis of the voice of peace.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert?

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.

Karthik Puranik, Adeep Hande, Ruba Priyadharshini, Sajeetha Thavareesan, and Bharathi Raja Chakravarthi. 2021. IIITT@LT-EDI-EACL2021-Hope Speech Detection: There is always hope in Transformers. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Sanford B Steever. 1998. Introduction to the Dravidian languages. *The Dravidian languages*, 1:39.

Burton Stein. 1977. Circulation and the historical geography of Tamil country. *The journal of Asian studies*, 37(1):7–26.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Glue: A multi-task benchmark and analysis platform for natural language understanding.

Konthala Yasaswini, Karthik Puranik, Adeep Hande, Ruba Priyadharshini, Sajeetha Thavareesan, and Bharathi Raja Chakravarthi. 2021. IIITT@DravidianLangTech-EACL2021: Transfer Learning for Offensive Language Detection in Dravidian Languages. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.