Leveraging Sentiment Data for the Detection of Homophobic/Transphobic Content in a Multi-Task, Multi-Lingual Setting Using Transformers

Filip Nilsson[1], Sana Sabah Al-Azzawi[1] and György Kovács[1]

[1]EISLAB Machine Learning, Luleå University of Technology, 977 54 Luleå, Sweden

Abstract

Hateful content is published and spread on social media at an increasing rate, harming the user experience. In addition, hateful content targeting particular, marginalized/vulnerable groups (e.g. homophobic/transphobic content) can cause even more harm to members of said groups. Hence, detecting hateful content is crucial, regardless of its origin, or the language used. The large variety of (often underresourced) languages used, however, makes this task daunting, especially as many users use code-mixing in their messages. To help overcome these difficulties, the approach we present here uses a multi-language framework. And to further mitigate the scarcity of labelled data, it also leverages data from the related task of sentiment-analysis to improve the detection of homophobic/transphobic content. We evaluated our system by participating in a sentiment analysis and hate speech detection challenge. Results show that our multi-task model outperforms its single-task counterpart (on average, by 24%) on the detection of homophobic/transphobic content. Moreover, the results achieved in detecting homophobic/transphobic content put our system in 1st or 2nd place for three out of four languages examined.

Keywords

Multi-Task, Multi-Language Learning, Hateful Language, Sentiment Analysis, Detecting Homophobic/-Transphobic Language.

1. Introduction

The increasing use of social media such as Twitter and Youtube has escalated the exploitation of these platforms to propagate violence [1]. This violence can take the form of hateful, offensive, and abusive language causing harm [2, 3]. To help preventing this harm, social media has been analyzed using various methods designed to detect offensive language, or more particularly, detect hateful language [4], and homophobic/transphobic content [5]. Homophobic/Transphobic content is a type of hateful language intending to harm LGBT+ people. Unfortunately, the shortage of labeled data has limited research in this area, especially in low resource languages [6]. One approach to overcome this problem is leveraging additional data for improving the detection of hateful language. Among others, Kovács et al. [7, 8] examined this option by (among other methods) using additional datasets created for the same task. In this paper, we extend this idea by leveraging datasets from related tasks, as well as datasets in different languages to improve the detection of homophobic/transphobic content in a multi-task, multi-language setting.

FIRE 2022: Forum for Information Retrieval Evaluation, Decemeber 09–13, 2022, Kolkata, India

filnil-8@student.ltu.se (F. Nilsson); sana.al-azzawi@ltu.se (S. S. Al-Azzawi); gyorgy.kovacs@ltu.se (G. Kovács)

https://github.com/flippe3/fire_2022 (F. Nilsson)

CEUR Workshop Proceedings (CEUR-WS.org)

For these experiments, we use the "Sentiment Analysis and Homophobia detection of YouTube comments in Code-Mixed Dravidian Languages" (hereinafter DravidianCodeMix) challenge at FIRE [9] which presents two shared tasks. One task for sentiment analysis (Task A), and another one for the detection of homophobic/transphobic content (Task B). The two tasks combined provided seven datasets with altogether four different languages. We participated in both tasks and used multi-task learning to fine-tune an XLM-RoBERTA (XLM-R) pre-trained language model to deal with multilingualism [10].

In this paper, we describe our approach and the results it attained. First, we discuss the related literature in Section 2. Then, in Section 3 we describe the tasks and datases of the challenge. This is followed by the description of our methods in Section 4, after which we present our experiments and results in Section 5. In Section 6 we analyze our experiments, then share our conclusions and plans for future work in Section 7.

2. Related Work

Here, we discuss part of the related literature providing relevant background to the DravidianCodeMix challenge, including work on detection of hateful language in general, and homophobic/transphobic language in particular. We also discuss different approaches to multi-task learning, its use in Natural Language Processing (NLP) and for the detection of hateful language.

2.1. Hateful Language

Hateful language is any insult directed against a person or group based on their protected category that aims to cause damage or stir hatred [11]. Hateful posts/comments incur the risk of prompting a shift towards violence. Moreover, this content can potentially cause a harmful emotional effect to its readers. For these reasons, automatic hate speech identification has attracted a lot of attention, and many approaches have been investigated for the detection of hate speech, as well as other relevant areas [12].

Existing methods mainly deal with hateful language detection as a classification task. These approaches can be categorized into two groups, namely traditional machine learning methods such as SVM [7, 11], logistic regression [13], random forest [11], gradient boosting decision tree models [14] and naive Bayes [11]. The other main category is that of deep learning-based methods, which can be further partitioned into deep learning architecture, word embedding based methods, and transformer-based methods. Transformer-based methods utilize pre-trained transformer models (e.g. BERT, ELECTRA, T5), and fine-tune them on datasets annotated for hateful language detection to detect harmful remarks in social media posts. These methods show remarkable performance on harm identification across different languages [15, 16, 17, 18, 19].

Homophobic/Transphobic comments are usually categorized as a type of hateful language directed toward LGBT+ individuals. This phenomenon has been a growing concern. Chakravarthi et al. [9] collected and created a dataset in 2021 for Homophobic classification. The dataset is a collection of 15,141 YouTube multilingual comments on social media. In addition, they made detection experiments using several classical ML and DL models as baselines, and in 2022 they organized a shared task based on the data collected, at an ACL workshop [5], to promote research on identifying homophobic/transphobic content.

2.2. Multi-Task Learning

One method to combat data scarcity in supervised learning is multi-task learning (MTL). In MTL models are simultaneously trained on multiple related tasks to improve their performance and generalization ability. MTL has shown promising results in various areas, including computer vision, reinforcement learning, speech processing [20], and NLP, where MTL has been studied in various levels of relatedness, goals, and features [21].

One prominent example of the application of MTL in NLP is the T5 [22] text-to-text transformer model. This model was primarily trained using a combination of supervised and unsupervised learning, and an MTL approach, which was shown to have a positive influence. The authors also experimented with various types of mixing, namely examples-proportional, temperature-scaled and equal mixing. Results indicated that examples-proportional mixing leads to the best performance, while equal mixing can degrade the performance due to the model overfitting on low-resource tasks. Other researchers working on MTL in NLP [23] demonstrated that joint fine-tuning a model on multiple languages could bring substantial improvements to the performance of a universal language encoder. For this, they used a tree-like structure with 16 heads, one for each language. The authors found that it was only in rare cases when this joint MTL training hurt the performance of their model.

MTL has also been used in the context of detecting hate speech [24]. Here, the authors used a multi-task model to detect hate speech in Spanish using related tasks of polarity and emotion classification to improve their model. They showed that their MTL system with task-specific output heads outperformed its single-task counterpart and achieved state-of-the-art results.

3. Data

The shared challenge includes two main tasks (Task A and Task B), both coupled with annotated datasets comprised of Youtube comments in various languages. **Task A** being a sentiment analysis problem [25], where the goal is to classify each comment into one of five categories: Positive, Negative, Mixed feelings, Unknown state, and comment not in the target language. The detailed statistics and the split of the datasets are shown in Table 1. As can be seen in Table 1, the data is relatively imbalanced, both in terms of the labels, and languages (the Tamil dataset having more examples than the other two languages put together). Moreover, we can also see that the different partitions have largely different class label distributions.

Table 1
Statistics of the data sets distributed for Task A

Labels	Languages								
	Tamil Dataset			Malayalam Dataset			Kannada Dataset		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Positive	20070	2257	73	6421	706	780	2823	321	374
Negative	4271	480	338	2105	237	258	1188	139	157
Mixed Feelings	4020	438	101	926	102	134	574	52	65
Unknown state	20070	611	137	5279	580	643	711	69	62
Text not in the target language	1667	176	0	1157	141	147	916	110	110

Table 2
Statistics of the data sets distributed for Task B

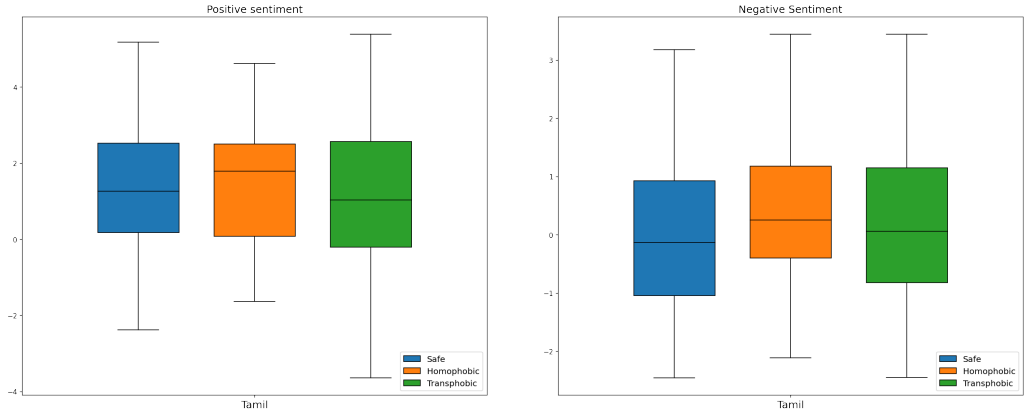
Labels	Languages											
	Tamil Dataset			Malayalam Dataset			English Dataset			Code-mixed Tamil-English Dataset		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Non-Anti-LGBT	2022	526	352	2434	692	971	3001	732	924	3438	862	1085
Homophobic	485	103	271	491	133	182	157	58	61	311	66	88
Transphobic	155	37	26	189	41	60	6	2	5	112	38	34

The main objective of **Task B** was to identify whether a comment was Transphobic, Homophobic, or Non-Anti-LGBT (Safe) [9]. The datasets provided for this task consist of Tamil, English, Malayalam, and the remaining code-mixed Tamil-English. The detailed statistics and the split of the datasets are shown in Table 2. As Table 2 shows, the Non-Anti-LGBT (safe) class outweighs the other two labels. This is most prominent in the English dataset, where transphobic messages represent less than 0.3% of the full data.

Some examples of the comments are also shown in Table 3. As can be seen from the table, two kinds of code-mixing is present in the data set. One, where different languages are mixed (e.g. the comment labelled as "mixed feelings" in row 6). We can see examples of the other type of code-mixing as well, where native and Roman scripts are mixed (e.g. the Non-LGBT comment in row 8).

Table 3
Sample comments from some of the datasets provided for the challenge

Task	Language	Text	Label
A	Malayalam	സ്വാസിക.... സ്വാസു.... love u dear.... all the best മുത്തേ... ALL KERALA SWASIKA FANS ASSOCIATION.....	Positive
		Padam onnu eragikotteda pahayanmare appalekum chelakkan kuthirikunu ororthanmaru.. Onnu poyinada..	Negative
		After jimikki kammal ഇതിനാവും കൂടുതൽ viewers 50M+ ആവും എന്നുള്ളവർക്ക് ലൈക്ക് ചെയ്യാം	unknown_state
		HISTORY OF POWER HISTORY OF VENGEANCE HISTORY OF PAIN HISTORY OF HOPE HISTORY OF AN ERA HISTORY OF THE BRAVE	not-malayalam
		nyc but best actor poley aayaaal seriyaavilla	Mixed_feelings
B	Tamil	இவளயெல்லாம் நாட்டில விட்டு வைப்பதே மிக தவறு நாய்	Homophobic
		ஆமா அண்ணா சரியா சொன்னீங்க govarnmend வேலை குடுத்தா உலகம் தாங்காது	Non-LGBT
		எல்லா மொள்ளமாறி தனமும் பண்ணிட்டு சமூகம் எங்கள புறக்கணிக்குது மயிரா புடுங்குதுன்னு ஒப்பாறிவேற தூக்கிப்போட்டு மிதிச்சா சரியாகும்	Transphobic
B	English-Tamil	Ipdi ellarm kum panna...AIDS varatha...tha...broo	Homophobic
		Kasta paduravangalukku sapadu thara neenga tha sir kadavul	Non-LGBT
		Saniyanungala savadikanum ithungala intha Uzhagathula irukanum nu thevaiya illa	Transphobic



(a) Positive sentiment (b) Negative sentiment

Figure 1: The result of applying a sentiment analysis model trained on Tamil sentiment data on Tamil hate data.

We hypothesized that the two tasks of the challenge are closely related. That is, negative sentiment would be more strongly associated with homophobic/transphobic comments than with safe ones, and the opposite would be true for the positive sentiment. To verify this, we trained a classifier for the Tamil dataset in Task-A (sentiment analysis), and applied it on the Tamil dataset in Task-B. Results of these experiments are shown in Figure 1b. As can be seen here, the negative sentiment on average was higher in homophobic and transphobic content than in safe comments. While for positive sentiment, on average higher scores were attained for safe comments than for transphobic ones. Although this was not the case for homophobic comments, the results partially supporting our hypothesis encouraged us to further examine the relation between the two tasks.

4. Methodology

One of our targets during our experiments was to examine how the task of identifying hateful comments can benefit from the availability of a sentiment analysis dataset. For this, we decided to work in a multi-task paradigm to benefit from the connection between the two tasks (supported by the sentiment scores attained using the data from the task of identifying hateful language in Figure 1b). For this, as our model, we chose to use a multi-lingual language model pre-trained on 100 different languages that included all languages in our datasets.

4.1. Preprocessing

To handle the irregularities often found in Youtube comments, we applied a preprocessing step in our pipeline. First, we used the Hugging Face normalizer to remove blank spaces and emojis. Then, we used the SentencePiece [26] tokenizer. SentecePiece is a language-independent subword tokenizer we chose to use since we are dealing with different types of code-mixing, a domain where SentencePiece has previously shown promising results [5].

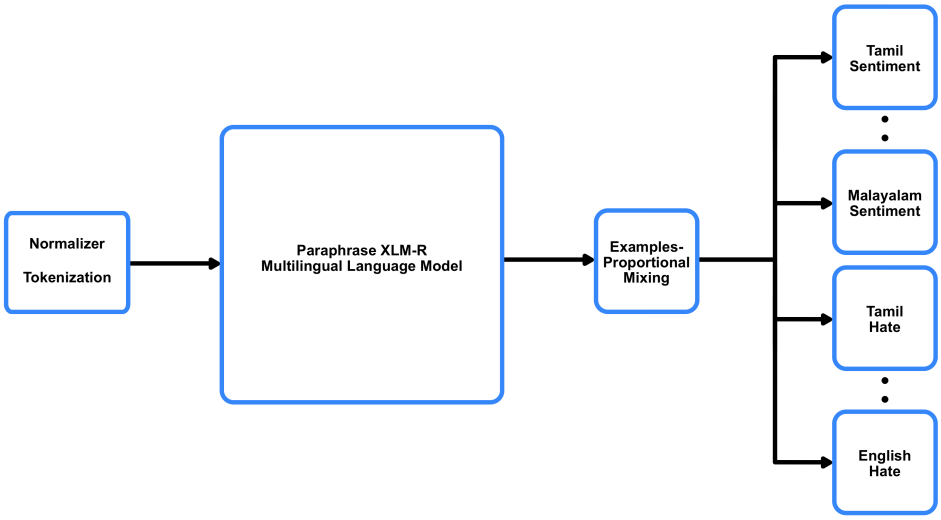


Figure 2: Pipeline of our seven-headed multi-task model

4.2. Model

We used the XLM-RoBERTA (XLM-R) [10] multilingual sentence-transformer which is trained using an XLM-R model as a student model and an SBERT [27] model as the teacher model.

4.3. Multi-Task Training

From our study of the data we hypothesized that hateful content would in general have a higher degree of negativity (and a lower degree of positivity). Therefore a multi-task model should be adequate to help improve the result for Task B. To train the model, we split the output heads into seven different RoBERTA [28] output heads shown in Figure 2, similar to the architecture used by Unicoder [23] that produced excellent results. This enabled us to fine-tune the language model using both tasks and all seven datasets simultaneously.

An essential part of MTL is the technique used to mix the tasks. In our case, due to the imbalance among datasets and labels, equal mixing would have incurred the risk of missing vital data points, negatively affecting the model. Thus, following the findings of [22] we used examples-proportional mixing, which means sampling from tasks proportionally to their dataset size.

In our MTL training we experimented using different sized output layers. We did not find any significant improvements when adding more layers, which can also be an indication of the closely related nature of the two tasks. The final model we used for training uses a RoBERTa classification head with two dense layers.

The training of our models was done using 4 epochs with a learning rate of 3e-5. Furthermore, we used a linear scheduler and an AdamW optimizer. These experiments were trained on a shared DGX-1 cluster using 2 x 32GB Nvidia V100 GPUs.

Table 4
Macro F1 scores for Task A [sentiment analysis]

Architecture	Languages	Output Heads	Language					
			Tamil		Malayalam		Kannada	
			Dev	Test	Dev	Test	Dev	Test
Multi-task	All	7	0.50	**0.20**	0.66	0.66	**0.55**	**0.52**
Multi-task	All	2	0.48	0.19	0.63	0.60	0.51	0.50
Single-task	One	1	0.51	0.20	**0.67**	0.66	0.54	0.52
LS Multi-task	Tamil & English	4	**0.51**	0.20	-	-	-	-
LS Multi-task	Malayalam	2	-	-	0.63	**0.71**	-	-
Best competing team	-	-	-	0.27	-	0.66	-	0.55

5. Experiments and results

We performed four experiments to evaluate our proposed multi-task system, each with a different architecture. In this section will present these architectures, as well as the results of these experiments. The four different architectures we were using in our experiments were as follows. 1) **Single-task architecture**: here, a separate model with only one output head was trained for each individual language and task. 2) **Multi-task learning with seven output heads**: here, a joint model was trained for all tasks and languages, equipped with one output head for each language and task. That is, the model had three output heads for the sentiment analysis (corresponding to the three languages), and had four more output heads (one for each language in Task B). 3) **Multi-task learning with two output heads**: in this approach, similar to the previous, all tasks and languages were used to train the same model. Here, however, unlike in the previous case, our goal was to train only one output head for each task. Thus one output head was trained to predict the labels of Task A and Task B respectively. This is similar to the arcitecture used by the Spanish hate-speech detection [24]. 4) **Language-specific multi-task learning**: similar to the two previous multi-task systems, but here we selected a specific language and only used datasets containing that language.

Results of our experiments on **Task A** are summarized in Table 4. As Table 4 shows, although the single-language multi-task model attained markedly higher results than that reported by the winning team, in most cases the use of multi-task architecture did not lead to marked improvements compared to its single-task counterpart. Our goal with the multi-task architecture, however, was not to improve the performance of the sentiment analysis model, but rather to improve the detection of hateful speech. Results of our experiments on **Task B** are summarized in Table 5. As Table 5 shows, we attained the best results on the test (and dev) set for all cases using a multi-language multi-task architecture, or a single-language multi-task architecture.

Based on the results attained on the dev set (where available at the deadline), we chose the seven-headed multi-task multi-language model for our final submission. Results attained in the official competition [29] by this model are summarized in table Table 6. The table shows that this model performed relatively well in both tasks. The rankings achieved, however, were markedly better for Task B, our main target. Thus, for the detection of hateful language, the multi-task multi-language model beyond attaining an improved performance compared to its single-task counterpart, also attained a competitive performance in the challenge.

Table 5
Macro F1 scores for Task B [detection of homophobic/transphobic language]

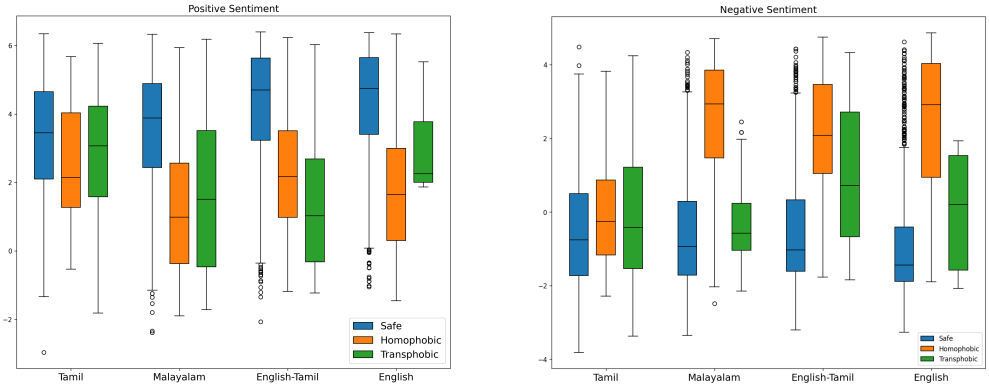
Architecture	Languages	Output Heads	Language							
			Tamil		Malayalam		English		Tamil-English	
			Dev	Test	Dev	Test	Dev	Test	Dev	Test
Multi-task	All	7	**0.69**	**0.32**	0.66	0.75	0.49	**0.49**	**0.64**	0.55
Multi-task	All	2	0.68	0.31	**0.71**	**0.76**	**0.51**	0.49	0.60	0.57
Single-task	One	1	0.50	0.27	0.52	0.52	0.48	0.45	0.47	0.48
Multi-task	Tamil & English	4	0.67	0.31	-	-	0.49	0.47	0.61	**0.59**
Multi-task	Malayalam	2	-	-	0.72	0.65	-	-	-	-
Best competing team	-	-	-	0.37	-	0.97	-	0.55	-	0.49

Table 6
Our official macro F1 scores and rankings across the two tasks tasks.

Task	Language	Precision	Recall	Macro F1	Rank
Task A	Tamil	0.15	0.13	0.13	7
	Malayalam	0.66	0.62	0.64	2
	Kannada	0.50	0.49	0.48	5
Task B	Tamil	-	-	0.33	2
	Tamil-English	-	-	0.55	1
	Malayalam	-	-	0.70	6
	English	-	-	0.49	2

6. Analysis

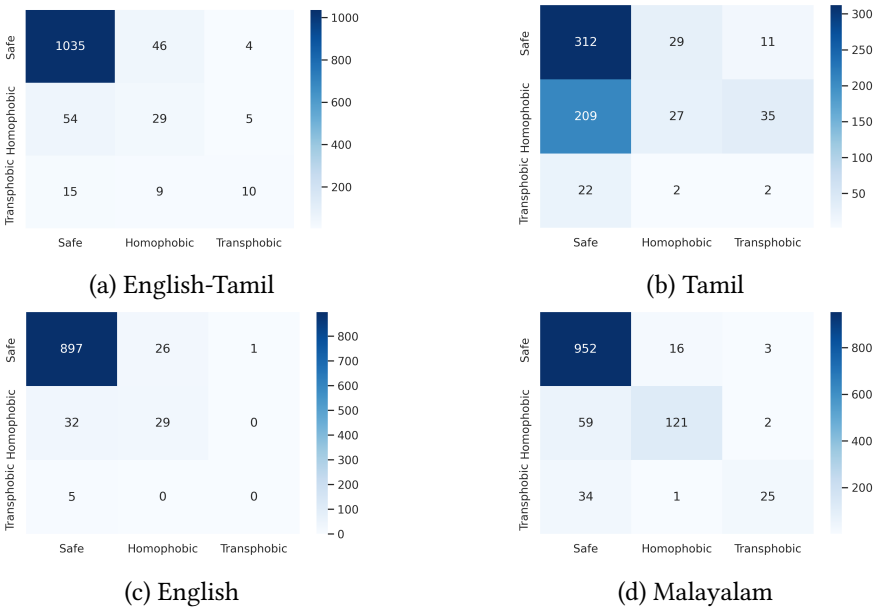
First, we examined our sentiment output heads when probed with hateful data to analyze further whether our system is viable. In Figure 3a and Figure 3b we show the logits when probed with hateful data through the matching sentiment head. In Figure 3a we show that the safe quartile is greater than the homophobic and transphobic quartile. This suggests that the model learned an expected feature (i.e. safe posts having more positivity, and less negativity than hateful ones).



(a) Positive sentiment

(b) Negative sentiment

Figure 3: Sentiments of probing our sentiment heads with hateful data



(a) English-Tamil

(b) Tamil

(c) English

(d) Malayalam

Figure 4: Language specific confusion matrices for hateful language

We also examined the confusion matrix, shown in Figure 4. This was an important tool of analysis, as classifying homophobic comments as transphobic might not be a large problem, but classifying either as safe is problematic. In Figure 4c we can see that five transphobic comments got misclassified as safe. However, this is most likely due to the training data since there were only six labels of transphobic data (see Table 2).

7. Conclusion and future work

We introduced a pipeline that fine-tunes a multi-lingual transformer using multi-task learning. The pipeline uses sentiment data to improve homophobic/transphobic detection. Our experiments show that the multi-task model outperforms the single-task model and that language-specific training can improve the accuracy further. We have also demonstrated that the sentiment output heads of our model identify hateful content as more negative than safe content.

Future work could focus on studying solely sentiment labels relevant for hateful speech. One could also look at other types of task-mixing, such as temporal-scaled mixing. Different early stopping methods should also be considered. Furthermore, data augmentation methods should also be considered, to counteract the problem of data imbalance. Finally, one can experiment with various tasks with different levels of relatedness and languages with varied similarities.

8. Acknowledgments

The work presented here was partially supported by Vinnova, in the project Language models for Swedish authorities (Språkmodeller för svenska myndigheter) ref. no.: 2019-02996

References

[1] A. A. Siegel, Online Hate Speech, SSRC Anxieties of Democracy, Cambridge University Press, 2020, p. 56–88.

[2] I. Bigoulaeva, V. Hangya, A. Fraser, Cross-lingual transfer learning for hate speech detection, in: Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, 2021, pp. 15–25.

[3] K. Gelber, L. McNamara, Evidencing the harms of hate speech, Social Identities 22 (2016) 324–341.

[4] A. Schmidt, M. Wiegand, A survey on hate speech detection using natural language processing, in: Proceedings of the fifth international workshop on natural language processing for social media, 2017, pp. 1–10.

[5] B. R. Chakravarthi, R. Priyadharshini, T. Durairaj, J. McCrae, P. Buitelaar, P. Kumaresan, R. Ponnusamy, Overview of the shared task on homophobia and transphobia detection in social media comments, in: Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 369–377. URL: https://aclanthology.org/2022.ltedi-1.57. doi:10.18653/v1/2022.ltedi-1.57.

[6] M. Singh, P. Motlicek, Idiap submission@ lt-edi-acl2022: Homophobia/transphobia detection in social media comments, in: Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, 2022, pp. 356–361.

[7] G. Kovács, P. Alonso, R. Saini, Challenges of hate speech detection in social media, SN Computer Science 2 (2021) 1–15. doi:10.1007/s42979-021-00457-3.

[8] G. Kovács, P. Alonso, R. Saini, M. Liwicki, Leveraging external resources for offensive content detection in social media, AI Commun. 35 (2022) 87–109. URL: https://doi.org/10.3233/AIC-210138. doi:10.3233/AIC-210138.

[9] B. R. Chakravarthi, R. Priyadharshini, R. Ponnusamy, P. K. Kumaresan, K. Sampath, D. Thenmozhi, S. Thangasamy, R. Nallathambi, J. P. McCrae, Dataset for identification of homophobia and transophobia in multilingual youtube comments, arXiv preprint arXiv:2109.00227 (2021).

[10] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, 2019. URL: https://arxiv.org/abs/1911.02116. doi:10.48550/ARXIV.1911.02116.

[11] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, in: Proceedings of the international AAAI conference on web and social media, volume 11, 2017, pp. 512–515.

[12] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, O. Frieder, Hate speech detection: Challenges and solutions, PLOS ONE 14 (2019) 1–16. URL: https://doi.org/10.1371/journal.pone.0221152. doi:10.1371/journal.pone.0221152.

[13] Z. Waseem, D. Hovy, Hateful symbols or hateful people? predictive features for hate speech detection on twitter, in: Proceedings of the NAACL student research workshop, 2016, pp. 88–93.

[14] E. Katona, J. Buda, F. Bolonyai, Using n-grams and statistical features to identify hate speech spreaders on twitter., in: CLEF (Working Notes), 2021, pp. 2025–2034.

[15] S. S. Sabry, T. Adewumi, N. Abid, G. Kovács, F. Liwicki, M. Liwicki, Hat5: Hate language identification using text-to-text transfer transformer, arXiv preprint arXiv:2202.05690 (2022).
[16] T. Adewumi, S. S. Sabry, N. Abid, F. Liwicki, M. Liwicki, T5 for hate speech, augmented data and ensemble, arXiv preprint arXiv:2210.05480 (2022).
[17] J. S. Malik, G. Pang, A. v. d. Hengel, Deep learning for hate speech detection: A comparative study, arXiv preprint arXiv:2202.09517 (2022).
[18] P. Alonso, R. Saini, G. Kovács, Hate speech detection using transformer ensembles on the hasoc dataset, in: International conference on speech and computer, Springer, 2020, pp. 13–21.
[19] E. Lavergne, R. Saini, G. Kovács, K. Murphy, Thenorth@ haspeede 2: Bert-based language model fine-tuning for italian hate speech detection, 2020.
[20] Y. Zhang, Q. Yang, An overview of multi-task learning, National Science Review 5 (2017) 30–43. URL: https://doi.org/10.1093/nsr/nwx105. doi:10.1093/nsr/nwx105. arXiv:https://academic.oup.com/nsr/article-pdf/5/1/30/31567358/nwx105.pdf.
[21] S. Chen, Y. Zhang, Q. Yang, Multi-task learning in natural language processing: An overview, 2021. URL: https://arxiv.org/abs/2109.09138. doi:10.48550/ARXIV.2109.09138.
[22] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, 2019. URL: https://arxiv.org/abs/1910.10683. doi:10.48550/ARXIV.1910.10683.
[23] H. Huang, Y. Liang, N. Duan, M. Gong, L. Shou, D. Jiang, M. Zhou, Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 2485–2494. URL: https://aclanthology.org/D19-1252. doi:10.18653/v1/D19-1252.
[24] F. M. Plaza-Del-Arco, M. D. Molina-González, L. A. Ureña-López, M. T. Martín-Valdivia, A multi-task learning approach to hate speech detection leveraging sentiment analysis, IEEE Access 9 (2021) 112478–112489. doi:10.1109/ACCESS.2021.3103697.
[25] B. R. Chakravarthi, R. Priyadharshini, V. Muralidaran, N. Jose, S. Suryawanshi, E. Sherly, J. P. McCrae, Dravidiancodemix: sentiment analysis and offensive language identification dataset for dravidian languages in code-mixed text, Language Resources and Evaluation 56 (2022) 765–806. URL: https://doi.org/10.1007/s10579-022-09583-7. doi:10.1007/s10579-022-09583-7.
[26] T. Kudo, J. Richardson, SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 66–71. URL: https://aclanthology.org/D18-2012. doi:10.18653/v1/D18-2012.
[27] N. Reimers, I. Gurevych, Making monolingual sentence embeddings multilingual using knowledge distillation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2020. URL: https://arxiv.org/abs/2004.09813.
[28] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer,

V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. URL: https://arxiv.org/abs/1907.11692. doi:10.48550/ARXIV.1907.11692.

[29] K. Shumugavadivel, M. Subramanian, P. K. Kumaresan, B. R. Chakravarthi, B. B, S. Chinnaudayar Navaneethakrishnan, L. S.K, T. Mandl, R. Ponnusamy, V. Palanikumar, M. Balaji J, Overview of the Shared Task on Sentiment Analysis and Homophobia Detection of YouTube Comments in Code-Mixed Dravidian Languages, in: Working Notes of FIRE 2022 - Forum for Information Retrieval Evaluation, CEUR, 2022.