UMUTeam@LT-EDI-ACL2022: Detecting homophobic and transphobic comments in Tamil

José Antonio García-Díaz and Camilo Caparrós-Laiz and Rafael Valencia-García
Facultad de Informática, Universidad de Murcia, Campus de Espinardo, 30100, Spain
{joseantonio.garcia8,camilo.caparrosl,valencia}@um.es

Abstract

This working-notes are about the participation of the UMUTeam in a LT-EDI shared task concerning the identification of homophobic and transphobic comments in YouTube. These comments are written in English, which has high availability to machine-learning resources; Tamil, which has fewer resources; and a transliteration from Tamil to Roman script combined with English sentences. To carry out this shared task, we train a neural network that combines several feature sets applying a knowledge integration strategy. These features are linguistic features extracted from a tool developed by our research group and contextual and non-contextual sentence embeddings. We ranked 7th for English subtask (macro f1-score of 45%), 3rd for Tamil subtask (macro f1-score of 82%), and 2nd for Tamil-English subtask (macro f1-score of 58%).

1 Introduction

This document outlines the participation of the UMUTeam in the workshop on Language Technology for Equality, Diversity, Inclusion (LT-EDI, ACL 2022) (Chakravarthi et al., 2022). Specifically, we describe our participation in a shared task regarding the identification of homophobic and transphobic comments in YouTube, written in English, Tamil, and a transliteration from Tamil to Roman script combined with English sentences. We ranked 7th for English subtask (macro f1-score of 45%), 3rd for Tamil subtask (macro f1-score of 82%), and 2nd for Tamil-English subtask (macro f1-score of 58%).

2 Related work

There are several surveys in the bibliography related with the identification of homophobic and transphobic comments. For instance, the works described at (Fortuna and Nunes, 2018) and (Jahan and Oussalah, 2021). Homophobic and transphobic comments are usually categorised as a form of hate-speech based on sexism or gender discrimination. These surveys indicate that there is a generic pipeline for building hate-speech detectors, that are based on the development of automatic document classification systems. The features for extracting data from textual sources are, usually, statistical methods. To name just a few, we mention the Bag of Word model, TF–IDF weights, topic modelling, and word and sentence embeddings. The models are traditional machine learning’s classifiers (Support Vector Machines, Logistic Regression, or Random Forest, among others) and different neural network architectures based on convolutional, recurrent neural networks and the usage of state-of-the-art models based on transformers, such as BERT.

Our research group has experience dealing with hate-speech. In (García-Díaz et al., 2021a), the Spanish MisoCorpus 2020, concerning misogyny identification, was released and evaluated. This dataset is released into three minor splits, concerning (1) the identification of misogyny towards relevant women, (2) to find the differences between safe, 9% labelled as Homophobic, and 5% labelled as transphobic labels.

| Label            | English | Tamil | Tamil-English |
|------------------|---------|-------|---------------|
| Homophobic       | 276     | 723   | 465           |
| Transphobic      | 13      | 233   | 184           |
| Safe             | 4567    | 3205  | 5385          |

Table 1: Number of labels for English, Tamil, and Tamil-English.

The details of the provided datasets can be found at (Chakravarthi et al., 2021). These datasets are divided into three splits, namely, training, validation, and test. Table 1 depicts the number of labels per dataset. as it can be observed, the datasets are heavily imbalanced. In general, each dataset has, approximately, 86% of the documents labelled as
Spanish of Spain and Latin-America, and (3) to identify general traits concerning misogyny, such as stereotypes of derailing. In (García-Díaz et al., 2022), we conduct an in-depth analysis concerning linguistic features and word and sentence embeddings. Specifically, we evaluated which are the best strategies to combine these features to build better hate-speech detectors.

As part of the doctoral thesis of one of the members of the team, in this shared-task we evaluate a subset of linguistic features that are language-independent. Therefore, a secondary objective of our participation is to observe if the combination of linguistic features and embeddings improves the performance of the automatic document classifiers.

3 Methodology

Our methodology can be summarised as follows. First, the pre-process the documents by removing extra spaces, blank lines, certain punctuation symbols, and emojis. Just for the English subtask, we also normalised the text by expanding acronyms and transformed the whole text into their lowercase form. Second, we extracted four feature sets that include linguistic features (LF), pretrained word embeddings from FastText (WE), sentences embeddings from FastText (SE), and sentence embeddings from BERT (BF). Third, we conduct an hyperparameter tuning strategy to build a neural network per feature set and one additional neural network that combines all feature sets (knowledge integration). Forth, we build two additional systems based on ensemble learning. Finally, we evaluate these methods to select the best approach for the final submission.

Next, we describe the feature sets employed in this work. The first feature set, LF, is a subset of language-independent features computed from the UMUTextStats tool (García-Díaz et al., 2021b; García-Díaz and Valencia-García, 2022). These features are related to stylometry, Part-of-Speech, emojis, and social media jargon. The second feature set, WE, is based on non-contextual embeddings from FastText. For this we use the pretrained embeddings of English (Mikolov et al., 2018) and the pretrained embeddings of Tamil (Grave et al., 2018). FastText calculates sentences embeddings by averaging word embeddings. The third feature set, SE, are sentence embeddings from FastText. The forth feature set, BF, is based on contextual sentence embeddings. We use BERT for English and the distilled version of multilingual BERT (Sanh et al., 2019). We use the distilled version because our machine could not train large batches with default BERT. The sentence embeddings are extracted with the [CLS] token (Reimers and Gurevych, 2019). To obtain the sentence embeddings from BERT, we evaluate 10 models with Tree of Parzen Estimators (TPE) (Bergstra et al., 2013). The evaluated parameters were the weight decay, the batch size, the warm-up speed, the number of epochs, and the learning rate.

Next, we train a neural network for each feature set and a neural network that combines all the feature sets using a knowledge integration strategy. For each training, we conduct a hyperparameter optimisation stage. The training is performed with RayTune (Liaw et al., 2018). In this stage, we evaluated shallow neural networks and deep neural networks. The main difference is the number of layers, using only one or two in shallow neural networks whereas deep neural networks use up to 8 hidden layers. Another difference is the composition of the neurons in each layer. In shallow neural networks, all the layers have the same number of neurons. In deep neural networks, on the other hand, we arranged the neurons in different shapes (brick-shape, triangle-shape, diamond-shape, rhombus-shape, short and long funnel-shape).

It is worth noting that the knowledge strategy allows to combine the features into the neural network consists in outputting each one into a different layer and then combine all the results into a new hidden layer. This strategy allows us to include two specific architectures with the non-contextual word embeddings from fastText: convolutional and recurrent neural networks. These networks exploit different characteristics of a text represented as a sequence. Convolutional networks exploit the spatial dimension, as it can make up new features from words that are together. Recurrent neural networks, on the other hand, exploits the temporal dimension. Specifically, we evaluate two bidirectional recurrent layers (BiLSTM and BiGRU). Besides, we evaluate several activation functions to connect the hidden layers, different learning rates and a dropout for regularisation.

Table 2 depicts the results achieved for every dataset with the validation split. We can observe that the performance for the homophobic and transphobic labels in English and Tamil-English is limited, but the results are promising for Tamil, reach-
ing a macro f1-score of 85.06%. For English and Tamil-English, both the precision and the recall are limited for the homophobic and transphobic labels. The lower results are caused by the strong class imbalance, which is not that big in the Tamil dataset.

In order to observe the performance of the best neural network with the validation split we obtain every confusion matrix (see Figure 1).

4 Results

The official results in the leaderboard are depicted in Tables 3, 4, 5 for English, Tamil, and Tamil-English respectively. We can observe that we achieved good results for Tamil and Tamil-English, achieving the third and second position in the leaderboard. However, our results were more limited with the English dataset, in which we ranked 7th.

We achieved a macro F1-score of 45% in the English dataset (see Table 3). This result is 12% below the best result (Abliment team, 57% of F1-score). We achieved similar f1-score with niksss, achieving slightly superior precision but lower recall.

Regarding Tamil (see Table 4) we achieved the 3rd position, with a macro f1-score of 82%. This result is 5% below the best result (ARGUABLY, f1-score of 87%) and 2% below the second-best result (NAYEL, 84% of f1-score). Besides, our system achieved worse precision and recall than both participants.

Finally, the results from the Tamil-English dataset output a macro F1-score of 58% (see Table 5). Similar to the ones achieved by bitsa_nlp. However, we achieved a significant drop in precision (61% vs 54%) but better recall (67% vs 56%).

5 Conclusions

In this article, we have summarised the participation of the UMUTeam in a task concerning the identification of homophobic and transphobic in social media posts. We are very pleased with our participation as we have participated with all the datasets as we have achieved competitive results.

As future work, we will continue adapting these techniques to Tamil and English, specially those focused on figurative language (del Pilar Salas-Zárate et al., 2020). One limitation that we found on our approach is that we do not handle code-mixed language properly. As future work, we will explore the reliability of using multilingual resources.
Table 2: Validation classification report for English, Tamil, and Tamil-English datasets with the neural network that combines all feature sets. P stands for precision, R for recall, and F1 for the macro f1-score

| Feature Set   | English | Tamil | Tamil-English |
|---------------|---------|-------|---------------|
| Homophobic    | 43.08   | 32.56 | 37.09         |
| Safe          | 96.11   | 97.59 | 96.84         |
| Transphobic   | 50.00   | 25.00 | 33.33         |
| Macro Avg     | 63.06   | 51.72 | 55.75         |
| Weighted Avg  | 93.11   | 93.88 | 93.44         |

Table 3: Official results for English. The columns indicate the accuracy (acc) and macro (m-) values of Precision (P), Recall (R) and F1-Score (F1)

| Team          | acc  | m-P  | m-R  | m-F1 |
|---------------|------|------|------|------|
| ARGUABLY      | 94   | 88   | 85   | 87   |
| UMUTeam       | 92   | 86   | 81   | 84   |
| NAYEL         | 92   | 85   | 80   | 82   |
| hate-alert    | 90   | 83   | 75   | 78   |
| bitsa_nlp     | 89   | 81   | 71   | 75   |
| nikss         | 85   | 69   | 61   | 64   |
| NAYEL         | 88   | 52   | 58   | 55   |
| SSNCSE_NLP    | 77   | 55   | 47   | 50   |
| SOA_NLP       | 69   | 36   | 36   | 36   |

Table 4: Official results for Tamil. The columns indicate the accuracy (acc) and macro (m-) values of Precision (P), Recall (R) and f1-score (F1)

| Team          | acc  | m-P  | m-R  | m-F1 |
|---------------|------|------|------|------|
| ARGUABLY      | 94   | 88   | 85   | 87   |
| UMUTeam       | 92   | 86   | 81   | 84   |
| NAYEL         | 92   | 85   | 80   | 82   |
| hate-alert    | 90   | 83   | 75   | 78   |
| bitsa_nlp     | 89   | 81   | 71   | 75   |
| nikss         | 85   | 69   | 61   | 64   |
| NAYEL         | 88   | 52   | 58   | 55   |
| SSNCSE_NLP    | 77   | 55   | 47   | 50   |
| SOA_NLP       | 69   | 36   | 36   | 36   |

Table 5: Official results for Tamil-English. The columns indicate the accuracy (acc) and macro (m-) values of Precision (P), Recall (R) and f1-score (F1)

| Team          | acc  | m-P  | m-R  | m-F1 |
|---------------|------|------|------|------|
| ARGUABLY      | 89   | 63   | 60   | 61   |
| UMUTeam       | 85   | 54   | 67   | 58   |
| bitsa_nlp     | 88   | 61   | 56   | 58   |
| hate-alert    | 83   | 54   | 63   | 56   |
| SOA_NLP       | 90   | 65   | 50   | 54   |
| Ablimet       | 80   | 49   | 64   | 53   |
| nikss         | 88   | 56   | 50   | 52   |
| NAYEL         | 90   | 62   | 47   | 51   |
| SSNCSE_NLP    | 89   | 66   | 43   | 47   |
| Sammaan       | 83   | 34   | 35   | 35   |
| Ajetavya      | 87   | 34   | 34   | 34   |

Acknowledgements

This work is part of the research project LaTe4PSP (PID2019-107652RB-I00) funded by MCIN/AEI/10.13039/501100011033. This work is also part of the research project PDC2021-121112-I00 funded by MCIN/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR. In addition, José Antonio García-Díaz is supported by Banco Santander and the University of Murcia through the Doctorado Industrial programme.

References

James Bergstra, Daniel Yamins, and David Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In International conference on machine learning, pages 115–123. PMLR.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Thenmozhi Durairaj, John Phillip McCrae, Paul Buitalaer, Prassana Kumar Kumaresan, and Rahul Ponnusamy. 2022. Findings of the shared task on Homophobia Transphobia Detection in Social Media Comments.
Bharathi Raja Chakravarthi, Ruba Priyadharshini, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Kayalvizhi Sampath, Durairaj Themmozhi, Sathiyaraj Thangasamy, Rajendran Nallathambi, and John Phillip McCrae. 2021. Dataset for identification of homophobia and transphobia in multilingual youtube comments. arXiv preprint arXiv:2109.00227.

María del Pilar Salas-Zárate, Giner Alor-Hernández, José Luis Sánchez-Cervantes, Mario Andrés Paredes-Valverde, Jorge Luis García-Alcaraz, and Rafael Valencia-García. 2020. Review of english literature on figurative language applied to social networks. Knowledge and Information Systems, 62(6):2105–2137.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. ACM Computing Surveys (CSUR), 51(4):1–30.

José Antonio García-Díaz, Mar Cánovas-García, Ricardo Colomo-Palacios, and Rafael Valencia-García. 2021a. Detecting misogyny in spanish tweets. an approach based on linguistics features and word embeddings. Future Generation Computer Systems, 114:506–518.

José Antonio García-Díaz, Ricardo Colomo-Palacios, and Rafael Valencia-García. 2021b. Psychographic traits identification based on political ideology: An author analysis study on spanish politicians’ tweets posted in 2020. Future Generation Computer Systems.

José Antonio García-Díaz, Salud María Jiménez-Zafra, Miguel Angel García-Cumberras, and Rafael Valencia-García. 2022. Evaluating feature combination strategies for hate-speech detection in spanish using linguistic features and transformers. Complex & Intelligent Systems, pages 1–22.

José Antonio García-Díaz and Rafael Valencia-García. 2022. Compilation and evaluation of the spanish satiricorpus 2021 for satire identification using linguistic features and transformers. Complex & Intelligent Systems, pages 1–14.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018).

Md Saroar Jahan and Mourad Oussalah. 2021. A systematic review of hate speech automatic detection using natural language processing. arXiv preprint arXiv:2106.00742.

Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. 2018. Tune: A research platform for distributed model selection and training. arXiv preprint arXiv:1807.05118.