ISSN 2007-9737

Harnessing Uncleaned Data for Stress Detection in Tamil and Telugu Code-Mixed Texts

Luis Ramos, Moein Shahiki-Tash*, Zahra Ahani, Alex Eponon,
Olga Kolesnikova, Hiram Calvo

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

{lramos2020, mshahikit2022, z.ahani2023, aeponon2023,
kolesnikova, hcalvo}@cic.ipn.mx

Abstract. Stress is a common experience in daily life, but it can significantly impact mental well-being in certain situations, making the development of robust detection models imperative. This proposal introduces a methodical approach to the stress detection in code-mixed texts for Dravidian languages. The challenge encompassed two datasets, targeting Tamil and Telugu languages respectively. This proposal underscores the importance of testing uncleaned text, such as deleting emojis, special characters, etc., in classification methodologies. In this proposal were evaluated Logistic Regression, Random Forest and Support Vector Machine algorithms featuring three textual representations: TF-IDF, word and character N-grams. This proposal demonstrated strong performance across both languages, achieving a Macro F1-score of 0.75 for Tamil and 0.74 for Telugu, surpassing the results obtained using other complex techniques involving LLMs. The results underscore the value of uncleaned text for mental state detection and the challenges of classifying code-mixed texts in Dravidian languages, indicating that there is potential to be explored, especially in Tamil and Telugu texts.

Keywords. Stress, NLP, machine learning, LLM, SMOTE, code-mixed, tamil, telugu.

1 Introduction

According to the World Health Organization (WHO)[1], stress is defined as a condition of anxiety or mental strain caused by challenging circumstances. Although, stress is a natural reaction to threats and stimuli that every individual experiences to some degree. Thus, not all stress states are harmful; chronicity, quality, magnitude, subjective appraisal, and context of stressors are important moderators of the stress response, but acute and chronic stress experiences can affect optimal neuroendocrine reactivity, leading to increased vulnerability of the organism to stressors [2, 18].

Social media has rapidly transformed into one of the major means of expressing oneself and communicating with the public and become a great platform to look over one's mood and feelings [29], in addition to serving as a tool for initial assessment and intervention of mental health concerns [8]. As these platforms allow exponential connectivity to transpire, developing solid strategies for identifying stress problems in real time is paramount in monitoring mental health on an ongoing basis [21].

When there is an insertion of words, phrases and morphemes from one language into a statement or expression from another language, it is called code mixing [19].The reasons could include lack of suitable equivalences to represent specialised terminologies, topic shift, speech clarification, in quotations, lack of proficiency in a language, when using familiar expressions in the other language [23], today mixing multiple languages together is a popular trend [30]. The prevalence of multilingualism on the internet, and code-mixed text data, has

[1] https://www.who.int/news-room/questions-and-answers/item/stress

doi: 10.13053/CyS-29-3-5499

Table 1. Data set distribution

Data set	Label	Train	Validation	Test	Total
Tamil	Non stressed	3720	939	650	5309
	stressed	1784	439	370	2593
Telugu	Non stressed	3314	799	650	4763
	stressed	1783	440	400	2623

Table 2. Sample instances of data set

Data set	Label	Text
Tamil	Non stressed	Bro video clip swap agi iruku atha gavanichingala
	stressed	bhaLLi Suttu vīzhthappattadhu.
Telugu	Non stressed	super comment pettav bro , chala navvostundi
	stressed	Nēnu 10 rōjulaṅgā snānam chēyalēdu!

become a popular research topic in natural language processing (NLP). It is a difficult task to handle bilingual and multilingual communication data. [39], one of these tasks is hope speech detection [5, 9, 32] or hate speech detection [4, 31, 40].

They attempt to compose a text that combines two or three different languages, resulting in the generation of Code-Mix data [33]. Some applications have been developed using NLP and machine learning (ML) in Dravidian languages, such as sentiment classification [27], abuse detection [10], hate and offensive language detection [26, 28]. There have not been any recent attempts to identify stress in Dravidian languages like Tamil and Telugu.

This approach seeks to bridge the gap by addressing the task with foundational methods that utilize text representation models, bypassing preprocessing steps such as the deletion of stop words, special characters, emojis, spelling symbols, etc. The results represent a benchmark for assessing future applications that employ diverse techniques in text preprocessing, feature extraction and ML models.

The rest of the paper follows as such: In Section 2, the related works on stress recognition using different machine learning approaches are discussed. Section 3 explains the dataset that was used in this proposal. Section 4 provides the methodology followed. Section 5 presents the results and discussion, and Section 6 offers the conclusion of the proposal. At the end, in Section 7 the limitations of the work are highlighted.

2 Related Work

Stress detection has been explored using various ML methods. Nijhawan et al. [24] employed sentiment analysis with five labels (Joy, Sadness, Neutral, Anger, and Fear), along with Latent Dirichlet Allocation (LDA), and ML to detect mental stress in social media texts. They achieved their best results using Random Forest (RF) with a precision of 97.78%. Yang et al. [38] collected texts from Twitter (today X) and applied predefined patterns to filter stress- related data, followed by manual tagging and the use of several classifiers. Preprocessing step involved lowercasing text and anonymizing URLs and usernames.

The 20,000 most frequent N-Grams were used as text representation, each word or character sequence was replaced with a dense numerical vector. Their best result was obtained with BERT reporting an F1-score of 0.86 for the negative class and 0.79 for the positive class and an accuracy of 83.6Shared task is basically a community-wide challenge where researchers are invited to solve a specific problem using a common dataset and evaluation metrics. Similar to previous proposals in NLP detection tasks, researchers utilized a range of machine learning approaches, including traditional classifiers like LR, SVM and advanced deep learning techniques [3] such as Transformers and LLM-based models.

The StressIdent_LT-EDI@EACL2024[2] shared task had the aim to detect stress in social media posts in which people share their feelings and emotions [34]. Given social media postings in Tamil and Telugu code-mixed languages, the submitted system should classify into two labels, “stressed” or “not stressed”. Various teams employed diverse machine learning approaches, ranging from traditional classifiers like Logistic Regression and Support Vector Machine to deep learning techniques such as CNNs, Transformers, and LLM-based models. In this workshop, Eponon et al. [14] proposed an approach for stress detection in Tamil and Telugu languages achieved a macro F1 score of 0.77 for Tamil and 0.72 for Telugu with FastText and Naïve Bayes. Raihan et al. [25] tested several methods to classify stress in Tamil

[2] https://codalab.lisn.upsaclay.fr/competitions/16092

Table 3. Train set distribution after data augmentation

Data set	Label	Train	Balanced Train
Tamil	Non stressed	3720	3720
	stressed	1784	3720
Telugu	Non stressed	3314	3314
	stressed	1783	3314

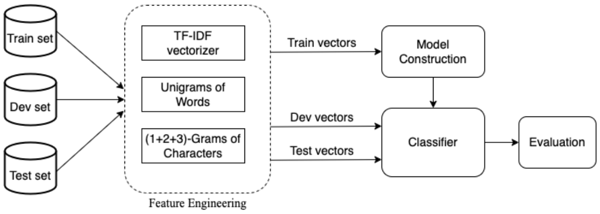

Fig. 1. Overview of the proposed methodology for traditional ML models

and Telugu languages, from traditional ML models to Transformers, but the best results were obtained using BERT-based models, these models achieved a macro F1 score of f 0.71 for Tamil and 0.72 for Telugu.

Andrew [8] uses GPT2 to detect stress in Tamil and Telugu languages. Although they used a transformer model with billions of parameters, it was not possible to achieve good performance. They only achieve a macro F1 score of 0.273 for Tamil and 0.251 for Telugu.

Overall, the literature analysed shows a wide variety of methods for stress detection in social media texts, including the use of conventional ML to advanced deep learning techniques.

3 Data Description

The data used for this proposal consisted of two datasets for educational purposes provided by the organizers of StressIdent_LT-EDI workshop. One dataset is in Tamil while the other is in Telugu.

Each dataset has two labels ("Non stressed" and "stressed") and is further split into training, validation, and test subsets. Nonetheless, only the training and testing subsets were utilized in this proposal. Both tables 1 and 2 display the statistics of the datasets for both languages, and show a few sample records from the Tamil and Telugu datasets respectively.

4 Methodology

This section outlines the methodology for each task. The primary objective of this proposal is to detect stress using a different ML models that is to detect stress using various machine learning (ML) models that rely on text representation models applied directly to raw data, without any cleaning steps and compare it with other complex models, like LLM's, and previously reported outcomes.

Logistic Regression (LR) is one of the most popular and widely used statistical method in medical research [12]. Random Forest (RF) was selected because the distribution of the random vectors does not depend on the training set, does not concentrate weight on any subset of the instances and the noise effect is smaller [11]. Finally, Support Vector Machine (SVM) is a very powerful ML model for identifying subtle patterns in complex data sets [16]. The overview of the proposed methodology for traditional ML models is exposed in Figure 1.

4.1 Feature Engineering

In this proposal, three feature extraction (FE) methods were used for each task: TF-IDF and N-Grams of words and characters. TF-IDF was chosen because it uses the term frequency and the document frequency for weighting each word [15]. N-grams of words or characters are groups of consecutive elements in a tokenized sentence, paragraph or document. In particular, Unigrams have been noted to work well with code mixed text classification [7]. Furthermore, character N-grams are hand-crafted features frequently used as discriminative factors in text categorization, language variety identification, and others. [20] Text representations were encoded using method *TfidfVectorizer* from *scikit-learn* with default parameters and following values of *ngram_range*: [(1,1), (2,2), (3,3), (1,2), (2,3), (1,2,3)]. For character N-grams, the parameter *analyzer*='char' was set.

Table 4. Comparison between the best performance model with unbalanced and balanced data (*Tamil & **Telugu)

Model (FE)	Weighted Scores			Macro Scores			
	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
SVM-RBF (1-CH)*	0.80	0.75	0.75	0.76	0.78	0.75	0.75
Unbalanced SVM-RBF (1-CH)*	0.80	0.73	0.74	0.75	0.77	0.73	0.73
Balanced- SMOTE SVM-RBF (2-CH)**	0.77	0.74	0.74	0.74	0.76	0.74	0.74
Unbalanced SVM-RBF (2-CH)**	0.77	0.73	0.73	0.74	0.75	0.73	0.73
Balanced- SMOTE							

Table 5. Best traditional ML performance in Tamil

Model (FE)	Weighted Scores			Macro Scores			
	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
LR (TF-IDF)	0.81	0.74	0.74	0.76	0.78	0.74	0.74
RF (1+2+3-W)	0.81	0.74	0.74	0.77	0.78	0.74	0.74
SVM-Linear (2-W)	0.80	0.74	0.74	0.75	0.77	0.74	0.74
SVM-RBF (1-CH)	0.80	0.75	0.75	0.76	0.78	**0.75**	0.75

Table 6. Best traditional ML performance in Telugu

Model (FE)	Weighted Scores			Macro Scores			
	Precision	Recall	F1-score	Precision	Recall	F1-score	Accuracy
LR (1-W)	0.77	0.72	0.72	0.74	0.75	0.72	0.72
RF (TF-IDF)	0.79	0.73	0.73	0.76	0.76	0.73	0.73
SVM-Linear (1-W)	0.80	0.73	0.73	0.76	0.76	0.73	0.73
SVM-RBF (2-CH)	0.77	0.74	0.74	0.74	0.76	**0.74**	0.74

4.2 Data augmentation

An over sampling approach was implemented using the Synthetic Minority Over-sampling Technique (SMOTE) to alleviate the imbalance in the dataset using the *imblearn* library. SMOTE involves oversampling the minority class by generating synthetic instances [6]. Different combinations of the SMOTE function parameters were tried, and the optimal outcome was attained by using the default values with only the 'sampling_strategy' parameter set to 'minority'. Significantly, these outcomes were worse than those obtained in the no over-sampling condition, which is presented in Table 4.

4.3 Evaluation

The performance of every model was analyzed using metrics such as the Macro F1-Score, Macro Recall, Macro Precision, Weighted F1-score, Weighted Recall, Weighted Precision, and Accuracy. Nonetheless, comparison with other proposals was mainly centered on the Macro F1-score, due to the Stress shared task utilizing it for evaluation.

Table 7. Performance of the best model in comparison with other proposals

Data set	Author	Macro F1-score	Macro Recall	Macro Precision	Weighted F1-score	Weighted Recall	Weighted Precision	Accuracy
Tamil	Eponon et al. [14]	0.723	0.775	-	-	-	0.822	0.724
Telugu		0.727	0.756	-	-	-	0.779	0.729
Tamil	Raihan et al. [25]	0.71	-	-	-	-	-	0.71
Telugu		0.72	-	-	-	-	-	0.72
Tamil	Andrew [8]	0.273	0.498	0.459	0.202	0.364	0.485	-
Telugu		0.251	0.247	0.255	0.287	0.281	0.293	-
Tamil	**This proposal**	**0.75**	0.78	0.76	0.75	0.75	0.80	**0.75**
Telugu		**0.74**	0.76	0.74	0.74	0.74	0.77	**0.74**

Table 8. Tamil Results with LLMs

Model	Weighted Scores			Macro Scores			
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Accuracy**
Gemma 2 7b	0.6874	0.4235	0.3273	0.6219	0.5407	0.3751	0.4235
Llama 3 8b	0.7742	0.4000	0.2692	0.3444	0.2646	0.1647	0.4000
Gemini 1.5 Flash	0.8041	0.0431	0.0809	0.3693	0.0172	0.0323	0.0431

Table 9. Telugu Results with LLMs

Model	Weighted Scores			Macro Scores			
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Accuracy**
Gemma 2 7b	0.6726	0.4514	0.3671	0.6214	0.5483	0.4064	0.4514
Llama3 8b	0.7671	0.4010	0.2520	0.6944	0.5162	0.3112	0.4010
Gemini 1.5 Flash	0.1779	0.3638	0.2390	0.1167	0.2387	0.1568	0.3638

4.4 LLM’s

Large Language Models (LLM’s) such as GPT-4 [1], Gemini [35] and Claude [13] have shown impressive performance; however, they remain closed-source. On the other hand, open-source models like LLaMA [37], Gemma [36] or Mistral [17] outperform older closed-source models [22]. In this proposal, we attempeted to conduct predictions with our dataset using Gemma, LLaMA, and Gemini models and comparing results with what we obtained using traditional ML models.

5 Results and Discussion

Following the classification task, as seen from Table 5 and Table 6, the SVM with an RBF kernel performed the best on both datasets.

The text was represented in various ways with TF-IDF and character and word n-grams. The best achievement was obtained using character unigrams and bigrams. The differences in F1-scores against the other previously reported results are illustrated in Table 4.3.

Evaluating the outcome using LLM’s; In Tamil, the analysis of the F1 score indicated that Gemma 2 7b model performed the best, with a Macro F1 Score of 0.3751. Following in rank was Llama3 8b, who although had a Macro F1 Score of 0.1647. Gemini 1.5 Flash is at the bottom when it comes to performance metrics. With a Macro F1 Score of 0.0323, Gemini 1.5 Flash performed poorly on both classes.

Similarly, for Telugu, the model that performs best is Gemma 2 7b with a Macro F1-Score of 0.4064. Llama3 8b had a Macro F1 score of 0.3112. Gemini 1.5 Flash yet again achieved the poorest results with a Macro F1 Score of

0.1568 where there was a lack of predictive performance. This leads us to conclude that Gemma 2 7b is the best model among these LLM's, but the performance is far below compared to traditional ML models.

From the results, each task had a different optimal text representation: For Tamil, a text representation that treated each character separately appeared to work best, while for Telugu, character bigrams proved to be more useful. This variation most likely reflects the characteristics of the particular text of each language. Results from LLM's suggest that these models struggle with code-mixed text, which is an area that requires further exploration especially when different alphabets are involved.

6 Conclusion

In conclusion, this approach showcases the ability to detect stress in Tamil and Telugu code-mixed texts using traditional ML models, along with not having any cleansing processes performed. It provides a valuable starting point for future efforts in stress detection for Tamil and Telugu.

Additionally, it highlights the importance of testing with uncleaned data in code-mixed texts for mental state detection. The findings highlight the challenges of classifying code-mixed texts in Dravidian languages (primarily due to the interplay of different languages and alphabets), and demonstrate that our proposal, as compared to more complex models, yields strong results. This paper contributes valuable insights for research in stress detection in code-mixed texts.

7 Limitations

This stress detection proposal for code-mixed texts in Dravidian languages (Tamil and Telugu) has shown effectiveness using traditional ML models, but limitations must be raised. The most significant limitation stems from dataset size, which is in general small and requires further investigation to assess whether it affects performance, especially regarding LLM's. Additionally, we have confined our analysis to the use of uncleaned data; however, a more in-depth study is necessary to ascertain the sources of the observed performance improvements in classification relative to previous approaches. Lastly, we have used a small number of LLMs, yet there is still room to analyze other available pre-trained models and apply fine-tuning or transfer learning to tailor these models to Tamil and Telugu languages.

Acknowledgments

The work was done with partial support from the Mexican Government through the grant A1-S-47854 of CONACYT, Mexico, grants 20241816, 20241819, and 20240951 of the Secretaría de Investigación y Posgrado of the Instituto Politécnico Nacional, Mexico. The authors thank the CONACYT for the computing resources brought to them through the Plataforma de Aprendizaje Profundo para Tecnologías del Lenguaje of the Laboratorio de Supercómputo of the INAOE, Mexico and acknowledge the support of Microsoft through the Microsoft Latin America PhD Award.

References

1. **Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023).** Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

2. **Agorastos, A., Chrousos, G. P. (2022).** The neuroendocrinology of stress: the stress-related continuum of chronic disease development. Molecular Psychiatry, Vol. 27, No. 1, pp. 502–513.

3. **Ahani, Z., Shahiki Tash, M., Ledo Mezquita, Y., Angel, J. (2024).** Utilizing deep learning models for the identification of enhancers and super-enhancers based on genomic and epigenomic features. Journal of Intelligent & Fuzzy Systems, , No. Preprint, pp. 1–11.

4. **Ahani, Z., Tash, M., Zamir, M., Gelbukh, I. (2024).** Zavira@ DravidianLangTech 2024: Telugu hate speech detection using LSTM. Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pp. 107–112.

5. **Ahani, Z., Tash, M. S., Tash, M., Gelbukh, A., Gelbukh, I. (2024).** Multiclass hope speech detection through transformer methods. Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2024), co-located with the 40th Conference of the Spanish Society for Natural Language Processing (SEPLN 2024), CEUR-WS. org.

6. **Alcauter, I., Martinez-Villaseñor, L., Ponce, H. (2023).** Explaining factors of student attrition at higher education. Computación y Sistemas, Vol. 27, No. 4, pp. 929–940.

7. **Ameer, I., Sidorov, G., Gomez-Adorno, H., Nawab, R. M. A. (2022).** Multi-label emotion classification on code-mixed text: Data and methods. IEEE Access, Vol. 10, pp. 8779–8789.

8. **Andrew, J. J. (2024).** JudithJeyafreeda_StressIdent_LT-EDI@ EACL2024: GPT for stress identification. Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pp. 173–176.

9. **Arif, M., Tash, M. S., Jamshidi, A., Ameer, I., Ullah, F., Kalita, J., Gelbukh, A., Balouchzahi, F. (2024).** Exploring multidimensional aspects of hope speech computationally: A psycholinguistic and emotional perspective. Preprint.

10. **Bansal, V., Tyagi, M., Sharma, R., Gupta, V., Xin, Q. (2022).** A transformer based approach for abuse detection in code mixed Indic languages. ACM transactions on Asian and low-resource language information processing.

11. **Breiman, L. (2001).** Random forests. Machine learning, Vol. 45, pp. 5–32.

12. **DeMaris, A., Selman, S. H., DeMaris, A., Selman, S. H. (2013).** Logistic regression. Converting Data into Evidence: A Statistics Primer for the Medical Practitioner, pp. 115–136.

13. **Enis, M., Hopkins, M. (2024).** From llm to nmt: Advancing low-resource machine translation with claude. arXiv preprint arXiv:2404.13813.

14. **Eponon, A. A., Batyrshin, I., Sidorov, G. (2024).** Pinealai_StressIdent_LT-EDI@ EACL2024: Minimal configurations for stress identification in Tamil and Telugu. Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pp. 152–156.

15. **Hakim, A. A., Erwin, A., Eng, K. I., Galinium, M., Muliady, W. (2014).** Automated document classification for news article in Bahasa Indonesia based on term frequency inverse document frequency (TF-IDF) approach. 2014 6th international conference on information technology and electrical engineering (ICITEE), IEEE, pp. 1–4.

16. **Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., Xu, W. (2018).** Applications of support vector machine (svm) learning in cancer genomics. Cancer genomics & proteomics, Vol. 15, No. 1, pp. 41–51.

17. **Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. (2023).** Mistral 7B. arXiv preprint arXiv:2310.06825.

18. **Kayalvizhi, S., Durairaj, T., Chakravarthi, B. R., et al. (2022).** Findings of the shared task on detecting signs of depression from social media. Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, pp. 331–338.

19. **Khandelwal, A., Swami, S., Akhtar, S. S., Shrivastava, M. (2018).** Gender prediction in English-Hindi code-mixed social media content: Corpus and baseline system. Computación y Sistemas, Vol. 22, No. 4, pp. 1241–1247.

20. **Kruczek, J., Kruczek, P., Kuta, M. (2020).** Are n-gram categories helpful in text classification? Computational Science–ICCS 2020: 20th International Conference, Amsterdam,

The Netherlands, June 3–5, 2020, Proceedings, Part II 20, Springer, pp. 524–537.

21. **Li, R., Liu, Z. (2020).** Stress detection using deep neural networks. BMC Medical Informatics and Decision Making, Vol. 20, pp. 1–10.

22. **Liu, T., Xiao, Y., Luo, X., Xu, H., Zheng, W., Zhao, H. (2024).** Geneverse: A collection of open-source multimodal large language models for genomic and proteomic research. Findings of the Association for Computational Linguistics: EMNLP 2024, pp. 4819–4836.

23. **Mpofu, P. (2022).** Code mixing in Kwayedza: Language subversion and the existence of African language newspapers. African Journalism Studies, Vol. 43, No. 4, pp. 15–30.

24. **Nijhawan, T., Attigeri, G., Ananthakrishna, T. (2022).** Stress detection using natural language processing and machine learning over social interactions. Journal of Big Data, Vol. 9, No. 1, pp. 33.

25. **Raihan, A., Rahman, T., Rahman, M., Hossain, J., Ahsan, S., Das, A., Hoque, M. M. (2024).** CUET_DUO@ StressIdent_LT-EDI@ EACL2024: Stress identification using Tamil-Telugu BERT. Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pp. 265–270.

26. **Rajalakshmi, R., Selvaraj, S., Vasudevan, P., et al. (2023).** Hottest: Hate and offensive content identification in Tamil using transformers and enhanced stemming. Computer Speech & Language, Vol. 78, pp. 101464.

27. **Rashmi, K., Guruprasad, H., Shambhavi, B. (2021).** Sentiment classification on bilingual code-mixed texts for Dravidian languages using machine learning methods. FIRE (Working Notes), pp. 899–907.

28. **Roy, P. K., Bhawal, S., Subalalitha, C. N. (2022).** Hate speech and offensive language detection in Dravidian languages using deep ensemble framework. Computer Speech & Language, Vol. 75, pp. 101386.

29. **S, K., Durairaj, T., Chakravarthi, B. R., C, J. M. (2022).** Findings of the shared task on detecting signs of depression from social media. Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, Association for Computational Linguistics, Dublin, Ireland, pp. 331–338. DOI: 10.18653/v1/2022.ltedi-1.51.

30. **Shekhar, S., Sharma, D. K., Beg, M. (2020).** An effective bi-lstm word embedding system for analysis and identification of language in code-mixed social media text in English and Roman Hindi. Computación y Sistemas, Vol. 24, No. 4, pp. 1415–1427.

31. **Tash, M., Ahani, Z., Zamir, M., Kolesnikova, O., Sidorov, G. (2024).** Lidoma@ LT-EDI 2024: Tamil hate speech detection in migration discourse. Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pp. 184–189.

32. **Tash, M. S., Ahani, Z., Tash, M., Kolesnikova, O., Sidorov, G. (2024).** Exploring sentiment dynamics and predictive behaviors in cryptocurrency discussions by few-shot learning with large language models. arXiv preprint arXiv:2409.02836.

33. **Tash, M. S., Ahani, Z., Tonja, A., Gemeda, M., Hussain, N., Kolesnikova, O. (2022).** Word level language identification in code-mixed Kannada-English texts using traditional machine learning algorithms. Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, pp. 25–28.

34. **Tash, M. S., Kolesnikova, O., Ahani, Z., Sidorov, G. (2024).** Psycholinguistic and emotion analysis of cryptocurrency discourse on X platform. Scientific Reports, Vol. 14, No. 1, pp. 8585.

35. **Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. (2023).** Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805.

36. **Team Gemma, Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S.,**

Sifre, L., Rivière, M., Kale, M. S., Love, J., et al. (2024). Gemma: Open models based on gemini research and technology. arXiv preprint arXiv:2403.08295.

37. **Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023).** Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

38. **Yang, Y.-C., Xie, A., Kim, S., Hair, J., Al-Garadi, M., Sarker, A. (2023).** Automatic detection of twitter users who express chronic stress experiences via supervised machine learning and natural language processing. CIN: Computers, Informatics, Nursing, Vol. 41, No. 9, pp. 717–724.

39. **Yigezu, M. G., Tonja, A. L., Kolesnikova, O., Tash, M. S., Sidorov, G., Gelbukh, A. (2022).** Word level language identification in code-mixed Kannada-English texts using deep learning approach. Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, pp. 29–33.

40. **Zamir, M., Tash, M., Ahani, Z., Gelbukh, A., Sidorov, G. (2024).** Lidoma@ Dravidian-LangTech 2024: Identifying hate speech in Telugu code-mixed: A BERT multilingual. Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pp. 101–106.

Article received on 02/03/2025; accepted on 20/05/2025.
**Corresponding author is Moein Shahiki-Tash.*