LATE-GIL-NLP at SemEval-2025 Task 11: Multi-Language Emotion Detection and Intensity Classification Using Transformer Models with Optimized Loss Functions for Imbalanced Data

Jesus Vázquez-Osorio[1,2], Helena Gómez-Adorno[1,3], Gerardo Sierra[1,4], Vladimir Sierra-Casiano[1,5], Diana Canchola-Hernández[1,5], José Tovar-Cortés [1,5], Roberto Solís-Vilchis[1,6], Gabriel Salazar[1,5],

[1]Universidad Nacional Autónoma de México, [2]Posgrado en Ciencia e Ingeniería de la Computación,

[3]Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, [4]Instituto de Ingeniería,

[5]Facultad de Ciencias, [6]Facultad de Estudios Superiores Acatlán

Correspondence: jesusvo5599@comunidad.unam.mx

Abstract

This paper addresses our approach to *Task 11: Bridging the Gap in Text-Based Emotion Detection* at the *SemEval-2025*, which focuses on the challenge of multilingual emotion detection in text, specifically identifying perceived emotions. The task is divided into tracks, we participated in two tracks: Track A, involving multilabel emotion detection, and Track B, which extends this to predicting emotion intensity on an ordinal scale. Addressing the challenges of imbalanced data and linguistic diversity, we propose a robust approach using pre-trained language models, fine-tuned with techniques such as extensive and deep hyperparameter optimization along with loss function combinations to improve performance on imbalanced datasets and underrepresented languages. Our results demonstrate strong performance on Track A, particularly in low-resource languages such as Tigrinya (ranked 2^{nd}), Igbo (ranked 3^{rd}), and Oromo (ranked 4^{th}). This work offers a scalable framework for emotion detection with applications in cross-cultural communication and human-computer interaction.

1 Introduction

Emotions are central to human communication, yet they are inherently complex and often difficult to express or interpret accurately through text, (Mohammad et al., 2018). While we regularly communicate our emotions, the way people perceive emotions in a text can be highly subjective, influenced by individual experiences, cultural backgrounds, and context, (Mohammad and Kiritchenko, 2018).

The *Task 11: Bridging the Gap in Text-Based Emotion Detection*, (Muhammad et al., 2025b), focuses on this challenge, dividing it into two tracks: Track A, which involves multilabel emotion detection, and Track B, which extends this to predicting emotion intensity on an ordinal scale (level 0 to 3). The task is particularly challenging due to imbalanced data and the diversity of languages (28 for Track A and 11 for Track B), each with unique linguistic and cultural nuances, (Muhammad et al., 2025a; Belay et al., 2025a).

This work addresses these challenges by leveraging pre-trained language models from *HuggingFace*, fine-tuned through an extensive search for optimal hyperparameters and custom loss functions tailored to emotion detection. Our approach combines models with advanced techniques such as *Cross Entropy Loss*, *Focal Loss*, and *Label Smoothing*, improving *F1 score* and Pearson correlation metrics, especially for underrepresented languages and imbalanced datasets. Key contributions include:

1. Systematic model selection, hyperparameter tuning, and loss function combination across 28 and 11 languages for Tracks A and B, respectively.

2. Custom loss functions to address the class imbalance and improve performance.

3. Strong results in low-resource languages, such as Tigrinya (ranked 2/35), Igbo (ranked 3/35), and Oromo (ranked 4/37) for Track A.

Our method provides a robust framework for multilingual emotion detection. It has potential applications in cross-cultural communication and human-computer interaction. The main codes used to address this task are available in the *GitHub*[1] repository of our research group.

2 Related Work

Emotion detection and classification have been widely studied in Natural Language Processing (*NLP*), with significant progress driven by deep learning architectures, transfer learning, and

[1]https://github.com/PLN-disca-iimas/Semeval2025-task11

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 666–674

multimodal approaches (Mohammad and Bravo-Marquez, 2017). Early methods relied on lexicon-based techniques and traditional machine learning models, but recent advancements leverage large-scale pre-trained models and hybrid neural architectures to capture complex linguistic patterns more effectively.

The field of emotion detection has evolved from lexicon-based approaches to modern deep-learning models capable of capturing contextual nuances. The adoption of deep learning significantly improved performance by allowing models to learn representations directly from data. Early neural network architectures, particularly recurrent models like Long Short-Term Memory (*LSTM*) networks, enhanced the ability to capture sequential dependencies in text. However, these models required substantial labeled data and computational resources. The introduction of transformer-based models, such as *BERT*, *RoBERTa*, and *XLM-R*, further advanced emotion detection by leveraging self-attention mechanisms to model complex linguistic structures, (Devlin et al., 2018). Pre-trained on large-scale corpora, these models set new benchmarks in emotion detection, outperforming earlier architectures. More recently, *instruction-tuned models*, such as *GPT-4* and *T5*, have shown promise in classifying emotions in ambiguous or contextually complex text, (Longpre et al., 2023). While multimodal approaches integrating textual, auditory, and visual data have gained traction, text-based models remain widely used due to their efficiency and accessibility.

While emotion detection focuses on identifying whether an emotion is present in a given text, an equally important challenge is determining its intensity. Emotion intensity classification has gained increasing attention in *NLP*, with notable advancements in deep-learning architectures and transfer learning. The advent of large-scale pre-trained transformers has further advanced emotion intensity classification. Models such as *BERT* and its derivatives have been fine-tuned on emotion-labeled datasets, achieving state-of-the-art results, (Qin et al., 2023). More recently, frameworks such as *DeepEmotex* have demonstrated the effectiveness of fine-tuned transformer-based models for multi-class emotion classification, significantly outperforming conventional deep learning models, (Hasan et al., 2022).

A major challenge in emotion classification is class imbalance, where certain emotions are significantly underrepresented in datasets. To address the imbalance, recent work has explored hierarchical classification and weighted loss functions to improve model performance in multilingual settings. In the *WASSA 2024* shared task, Vázquez-Osorio et al. (Vázquez-Osorio et al., 2024) proposed a two-stage hierarchical classification approach. The first stage classified tweets into four broad categories, while the second stage further distinguished between underrepresented emotions. Additionally, they employed *FocalLoss* and weighted *CrossEntropyLoss* to emphasize minority classes during training. Their approach, implemented with a fine-tuned *DeBERTa-v3-large* model, resulted in improved performance, ranking among the top 15 submissions. These findings highlight the effectiveness of hierarchical classification and adaptive loss functions in handling imbalanced emotion datasets.

Recent work on multilingual emotion detection (Belay et al., 2025a,b) highlights challenges such as class imbalance and low-resourced languages. Therefore, our approach introduces custom loss functions for imbalanced data, which is critical for some languages with underrepresented data.

3 System Overview

Our system is based on fine-tuning pre-trained transformer models for multilingual emotion recognition and intensity prediction. We designed a two-stage pipeline tailored to the task's requirements of Track A and Track B. In Track A, we focused on detecting the presence of emotions in text through multilabel classification. In Track B, we extended this setup by incorporating an additional step to predict the intensity of each detected emotion. All models were language-specific and selected based on empirical performance.

For Track A, we fine-tuned a multilabel binary classification model for each language to determine whether each emotion was present in the text.

Track B expanded on Track A's process, incorporating a two-step approach to predict emotion intensity. The initial step, the same as in Track A, employed the multilabel classification model to identify emotions. In the next phase, if an emotion was detected (label 1), a separate model estimated its intensity on a scale from 1 to 3.

All our models were based on pre-trained transformer models, which were selected as the most suitable for each language.

3.1 Fine-tuning process

Fine-tuning is the process of adapting the weights of the neural network to optimize the performance for one specific task.

Our approach involved two sequential fine-tuning steps:

1. Binary emotion detection: We first fine-tuned a multilabel classification model to determine whether each emotion was present in a given text. The best-performing model variant for each language was selected based on this step.

2. Emotion intensity classification: Once the best binary classification model was identified, we fine-tuned separate models, one for each emotion intensity prediction. These models classified intensity levels into three categories (1, 2, and 3).

We performed hyperparameter optimization on all fine-tuned models. For Track A, only the first step was necessary. For Track B, both steps were required. Figure 1 shows the workflow of model prediction generation for both tracks.

3.2 Loss Function Optimization

We experimented with multiple loss functions, including standard *Cross Entropy Loss*, *BCE With Logits Loss*, *Focal Loss*, *Label Smoothing Loss*, *MSE Loss*, and a custom loss function that averaged *Focal Loss* with sum reduction, *Weighted Cross Entropy Loss* and *Weighted Smooth Cross Entropy Loss*. This approach has been effective in handling imbalanced datasets, as proposed in (Shi et al., 2024)

To optimize performance in Track A, we experimented with various loss functions and tested a code implementation that combined different previously mentioned losses. The results in Table 4 showed that *BCE With Logits Loss* and *Cross Entropy Loss* were the most frequently selected in the tested combinations, leading to better performance on imbalanced datasets.

For Track B, for each language-emotion combination, we trained models with both *Cross Entropy Loss* and the custom loss function and selected the one that yielded the best performance. As a result, some models used cross-entropy, while others benefited from the custom loss function. In some cases where the training dataset was highly imbalanced, the custom loss function generally provided better results.

4 Experimental Setup

4.1 Data

The dataset provided for all task languages was delivered in separate corpora and divided into three standard splits: *train*, *dev*, and *test*. The *train* and *dev* sets contained golden labels, whereas the *test* set contained no labels. The data were divided as follows:

- Development phase: 80% of the *train* set was used for training the models, and the remaining 20% was reserved for validation during the development phase. This was done for all stages of our solution, i.e., model selection, hyperparameter tuning, and custom loss functions combination. The data split was performed in a stratified way for the emotion classes. The *dev* set was used exclusively for testing the trained models and evaluating their performance, with a particular focus on the macro *F1-score*, which was the primary evaluation metric for Track A.

- Test phase: After finalizing the best-performing model and hyperparameters, the model was retrained using the entire set of *train* and *dev* (combined) to make predictions in the unlabeled test set without an explicit validation stage in the training.

For Track B, the same data split strategy was applied, but the task was extended to predict the intensity of the emotions for each emotion class in the provided languages. In this track, each dataset contains texts annotated with one of six emotions (five for English), with intensity levels ranging from 0 (lower/no emotion detected) to 3 (higher level of intensity). However, a strong class imbalance was observed in all data sets. Most instances were labeled with an intensity of 0, indicating the absence of emotion. A smaller proportion of instances had an intensity of 1, while intensity level 2 was even less frequent. Instances labeled with the highest intensity, 3, were extremely rare and almost nonexistent in some datasets. This imbalance was a consistent pattern across all languages, posing a challenge for model training and evaluation.

4.2 Methods

4.2.1 Preprocessing and Parameter Tuning

The preprocessing and parameter-tuning process involved the following steps:

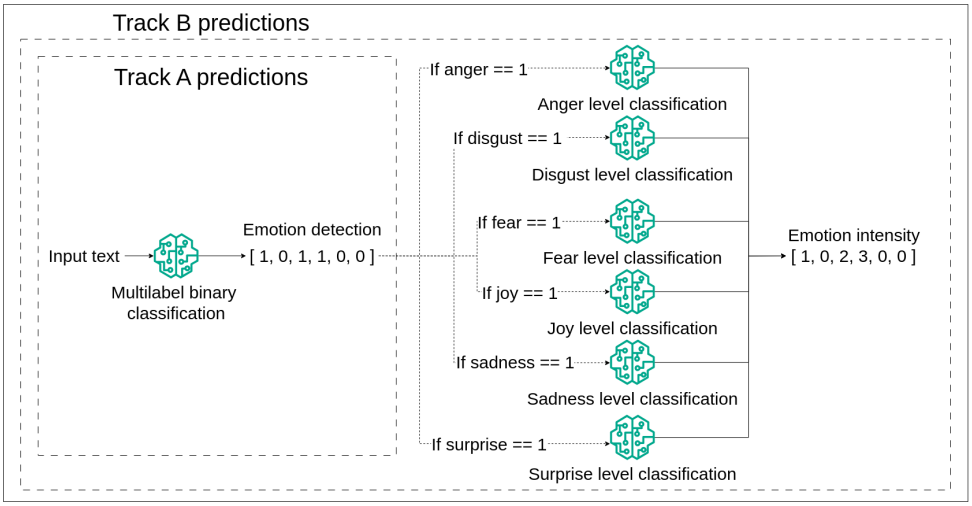

Figure 1: System workflow for predictions

1. **Model selection:** *Python* script was developed to evaluate multiple pre-trained models available on *HuggingFace* that supported the target languages. Models were filtered based on language compatibility, and the top 5 most downloaded models for each language were selected for initial testing. Each model was fine-tuned for 1 epoch on the training set to identify the best-performing model for each language.

2. **Hyperparameter tuning:** A grid search was conducted to optimize hyperparameters using the best-performing model identified in the previous step, all models were trained with the *AdamW*[2] optimizer. The hyperparameters and their respective search ranges were as indicated in Table 1:

Hiperparameter	Proposed values			
Learning rate	$1e^{-5}$	$2e^{-5}$	$3e^{-5}$	$5e^{-5}$
Weight decay	0.001	0.01	0.1	
Dropout prob	0.3		0.5	

Table 1: Proposed hyperparameters for grid search

3. **Custom Loss Functions Combination:** To further improve performance, a customized loss function selection process was applied. The following loss functions were evaluated in all possible combinations: *BCEWithLogitsLoss*, *MSELoss*, *CrossEntropyLoss*, *FocalLoss* with *alpha*=0.25, *gamma*=2.0, *LabelSmoothingLoss* with *smoothing*=0.1.

4. **Final Model Training:** After identifying the best model, hyperparameters, and loss function combination, the final model was trained on the combined *train* and *dev* sets for submission on the *test* set.

4.3 External tools and libraries

The following tools and libraries were used for preprocessing, training, and evaluation:

- *HuggingFace transformers*[3]: For accessing and fine-tuning pre-trained language models.
- *PyTorch*[4]: For implementing custom loss functions and training pipelines.
- *Scikit-learn*[5]: For evaluating model performance using metrics such as macro *F1-score*.
- *Pandas*[6] and *NumPy*[7]: For data manipulation and preprocessing.

4.4 Test Phase Submissions

For the test phase in Track A, three submission attempts were made:

1. **First submission:** Predictions were generated using models trained with the best hyperparameters and custom loss functions.

2. **Second submission:** Predictions were generated using models trained only with the best hyperparameters (without custom loss functions).

[2] https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html

[3] https://huggingface.co/

[4] https://pypi.org/project/torch/

[5] https://pypi.org/project/scikit-learn/

[6] https://pypi.org/project/pandas/

[7] https://pypi.org/project/numpy/

3. **Third submission:** The best-performing predictions from the first two submissions were selected based on their macro *F1 scores* and submitted as the final results.

4.5 Track B Extension

For Track B, the approach was extended to predict the intensity for each emotion class. Separate models were trained for each emotion and language, following the same pipeline as Track A but adapted for the ordinal intensity classification task. However, it is important to mention that, due to a lack of resources and time, we only made one submission in this final/test phase for this track.

5 Results

Macro *F1 score* was the official metric for Track A, and Table 2 shows only the third submission's results with its unofficial ranking. Our best score was 0.8731 in Hindi, ranking 14^{th}, while for Tigrinya, we placed 2^{nd} with a performance of 0.5874.

Language	Ranking	Macro *F1 score*
Afrikaans	31	0.3496
Amharic	22	0.5815
Arabic (Algerian)	30	0.4862
Arabic (Moroccan)	22	0.5132
Chinese (Mandarin)	26	0.5742
Emakhuwa	17	0.1626
English	21	0.7632
German	20	0.6334
Hausa	21	0.6149
Hindi	14	0.8731
Igbo	3	0.5628
Kinyarwanda	9	0.5112
Marathi (English)	30	0.7876
Nigerian Pidgin	23	0.5173
Oromo	4	0.5920
Portuguese (Brazil)	17	0.5470
Portuguese (Mozambique)	12	0.4754
Romanian	25	0.7082
Russian	37	0.7975
Somali	25	0.3692
Spanish	31	0.7517
Sundanese	25	0.4083
Swahili	20	0.2850
Swedish	31	0.4599
Tatar	18	0.6554
Tigrinya	2	0.5874
Ukrainian	31	0.4748
Yoruba	6	0.3754

Table 2: Results for each language in Track A

Table 3 shows the results for every language in Track B. Pearson correlation was used as the official evaluation metric for Track B. It evaluates the degree of linear association between the predicted labels and the gold ones. Our highest score was in Russian with a value of 0.7793.

As we can see, for both tracks, the macro *F1 score* and the Pearson correlation vary significantly across languages. Given that for each language, we fine-tuned a language-specific model, we can observe that the results are highly dependent on the base model. For Hindi, English, Russian, and Spanish, the scores are considerably higher compared to languages like Amharic, Algerian Arabic, or Ukrainian.

Language	Ranking	Pearson correlation
Amharic	15	0.4787
German	23	0.4676
English	23	**0.7228**
Spanish	22	0.6719
Portuguese (Brazil)	16	**0.5079**
Russian	24	0.7793
Arabic (Algerian)	17	**0.3982**
Chinese (Mandarin)	20	**0.4842**
Hausa	11	**0.6360**
Ukrainian	19	**0.4196**
Romanian	15	**0.6053**

Table 3: Result for each language in Track B.
Note: The Pearson correlation results shown in **bold** exceeded the organizers' baseline.

Other factors to take into consideration that can influence the performance of the classification models include the size of the dataset and class imbalance (see Appendix B for details on class distribution for Track A).

Unlike top teams reported in the Task paper (Muhammad et al., 2025b), such as Pai and Chinchunmei, who rely on large LLM ensembles, contrastive learning, and prompt engineering, our approach focuses on robust fine-tuning of pre-trained models using hyperparameter optimization and tailored loss functions for imbalanced and low-resource data. Without external augmentation or instruction tuning, our method achieved competitive results, ranking in the top 10 in multiple languages and significantly outperforming baselines in 17 languages, highlighting the strength of our optimization-based strategy (see Table 4). In addition, our system prioritizes reproducibility, with a simplified architecture and fully documented training settings, making it practical and easy to reproduce with promising results.

6 Ethical Considerations

Predicting perceived emotions and their intensity is inherently subjective and influenced by cultural and individual differences. Biases may arise from the dataset, the annotation process, or the model

itself. As noted in (Muhammad et al., 2025a), the dataset focuses on perceived emotions (what most annotators believe the speaker may have felt and their intensity) rather than determining the true emotional state of the speaker. Therefore, our prediction models should not be used for high-stakes decisions, nor should their outputs be interpreted as definitive assessments of the speaker's actual emotions or intensity. For more ethical considerations and details on the data annotation process, see (Muhammad et al., 2025a).

7 Conclusion

In this paper, we presented our approach to the emotion detection and intensity classification task at *SemEval-2025*. Our approach leveraged fine-tuned models based on pre-trained transformer models. We achieved our best results by incorporating some custom loss functions for certain languages and emotions, demonstrating their effectiveness in handling imbalanced data. However, the performance varied significantly across languages, underscoring the importance of further analyzing and exploring additional techniques and architectures.

Acknowledgments

This work was partially supported by UNAM PAPIIT projects IG400725, IN104424, IG400325 and by the Mexican Government through SECIHTI Proyect FC-2023-G-64.

References

Tadesse Destaw Belay, Israel Abebe Azime, Abinew Ali Ayele, Grigori Sidorov, Dietrich Klakow, Philip Slusallek, Olga Kolesnikova, and Seid Muhie Yimam. 2025a. Evaluating the capabilities of large language models for multi-label emotion understanding. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3523–3540, Abu Dhabi, UAE. Association for Computational Linguistics.

Tadesse Destaw Belay, Dawit Ketema Gete, Abinew Ali Ayele, Olga Kolesnikova, Grigori Sidorov, and Seid Muhie Yimam. 2025b. Enhancing multi-label emotion analysis and corresponding intensities for ethiopian languages.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Maryam Hasan, Elke Rundensteiner, and Emmanuel Agu. 2022. Deepemotex: Classifying emotion in text messages using deep transfer learning. *Preprint*, arXiv:2206.06775.

Shayne Longpre, Le Hou, Albert Webson, et al. 2023. Scaling instruction-finetuned language models. *Preprint*, arXiv:2303.08774.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 task 1: Affect in tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics.

Saif Mohammad and Svetlana Kiritchenko. 2018. Understanding emotions: A dataset of tweets to study interactions between affect categories. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Saif M. Mohammad and Felipe Bravo-Marquez. 2017. Wassa-2017 shared task on emotion intensity. *Preprint*, arXiv:1708.03700.

Shamsuddeen Hassan Muhammad, Nedjma Ousidhoum, Idris Abdulmumin, Jan Philip Wahle, Terry Ruas, Meriem Beloucif, Christine de Kock, Nirmal Surange, Daniela Teodorescu, Ibrahim Said Ahmad, David Ifeoluwa Adelani, Alham Fikri Aji, Felermino D. M. A. Ali, Ilseyar Alimova, Vladimir Araujo, Nikolay Babakov, Naomi Baes, Ana-Maria Bucur, Andiswa Bukula, Guanqun Cao, Rodrigo Tufino Cardenas, Rendi Chevi, Chiamaka Ijeoma Chukwuneke, Alexandra Ciobotaru, Daryna Dementieva, Murja Sani Gadanya, Robert Geislinger, Bela Gipp, Oumaima Hourrane, Oana Ignat, Falalu Ibrahim Lawan, Rooweither Mabuya, Rahmad Mahendra, Vukosi Marivate, Andrew Piper, Alexander Panchenko, Charles Henrique Porto Ferreira, Vitaly Protasov, Samuel Rutunda, Manish Shrivastava, Aura Cristina Udrea, Lilian Diana Awuor Wanzare, Sophie Wu, Florian Valentin Wunderlich, Hanif Muhammad Zhafran, Tianhui Zhang, Yi Zhou, and Saif M. Mohammad. 2025a. Brighter: Bridging the gap in human-annotated textual emotion recognition datasets for 28 languages. *arXiv preprint arXiv:2502.11926*.

Shamsuddeen Hassan Muhammad, Nedjma Ousidhoum, Idris Abdulmumin, Seid Muhie Yimam, Jan Philip Wahle, Terry Ruas, Meriem Beloucif, Christine De Kock, Tadesse Destaw Belay, Ibrahim Said Ahmad, Nirmal Surange, Daniela Teodorescu, David Ifeoluwa Adelani, Alham Fikri Aji, Felermino Ali, Vladimir Araujo, Abinew Ali Ayele, Oana Ignat, Alexander Panchenko, Yi Zhou, and Saif M. Mohammad. 2025b. SemEval-2025 task 11: Bridging the gap in text-based emotion detection. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, Vienna, Austria. Association for Computational Linguistics.

Xiangyu Qin, Zhiyu Wu, Jinshi Cui, Tingting Zhang, Yanran Li, Jian Luan, Bin Wang, and Li Wang.

2023. Bert-erc: Fine-tuning bert is enough for emotion recognition in conversation. *Preprint*, arXiv:2301.06745.

Ning Shi, Senyu Li, Guoqing Luo, Amirreza Mirzaei, Ali Rafiei, Jai Riley, Hadi Sheikhi, Mahvash Siavashpour, Mohammad Tavakoli, Bradley Hauer, and Grzegorz Kondrak. 2024. UAlberta at SemEval-2024 task 1: A potpourri of methods for quantifying multilingual semantic textual relatedness and similarity. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1798–1805, Mexico City, Mexico. Association for Computational Linguistics.

Jesús Vázquez-Osorio, Gerardo Sierra, Helena Gómez-Adorno, and Gemma Bel-Enguix. 2024. PCICU-NAM at WASSA 2024: Cross-lingual emotion detection task with hierarchical classification and weighted loss functions. In *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 490–494, Bangkok, Thailand. Association for Computational Linguistics.

A Appendix 1: Experimental Setup Final Training

Table 4 summarizes the configurations and results for the final training of models in Track A, covering all 28 languages. For each language, we report the best configuration of model architecture, learning rate, weight decay, dropout probability, and the combination of custom loss functions, which were used during the training of the final model used in the competition phase. If no loss function is found in the language row, this indicates that the best configuration obtained omits the use of any of the custom loss functions. The custom loss functions are encoded as follows: *BCEWithLogitsLoss* (1), *CrossEntropyLoss* (2), *FocalLoss* (3), and *LabelSmoothingLoss* (4). The table also includes the unofficial ranking achieved by our final submission, with results exceeding the organizers' baseline highlighted in bold. Submissions marked with an asterisk (*) represent the best-performing configuration combining hyperparameters and custom loss functions, while double asterisks (**) indicate submissions using only the best hyperparameters (without custom loss functions). For some languages (marked with ***), the model was trained differently due to the unique structure of the pretrained model used. This table highlights the effectiveness of our systematic approach, particularly in low-resource languages, where our method achieved competitive rankings, such as Tigrinya, Igbo, Nigerian Pidgin, and Yoruba.

B Appendix 2: Heatmap of emotions distribution by language

Figure 2 shows a heatmap of emotion distribution across languages. It can be observed that English (ENG) has a noticeable imbalance in its emotional distribution, with *Fear* making up a large portion at 58.2% and *Sadness* following at 31.7%. Similarly, Chinese (CHN) and Brazilian Portuguese (PTBR) stand out for their unusually high levels of *Anger*, at 44.6% and 32.3%, respectively. The most striking case is Sundanese (SUN), where *Joy* dominates, making up a significant 72.7% of the texts. On the other hand, some languages show a clear lack of certain emotions. For example, Afrikaans (AFR) and Ukrainian (UKR) have surprisingly low levels of *Anger*, at just 3.6% and 4.0%, respectively. Meanwhile, Oromo (ORM) and Yoruba (YOR) fall short in representing *Sadness* (8.7%) and *Joy* (9.1%), respectively.

Language	Pre-trained model	Learning rate	Weight decay	Dropout prob	Custom loss functions used	Unofficial ranking
Afrikaans (AFR)*	distilbert/distilbert-base-multilingual-cased	$5e^{-5}$	0.1	0.3	2, 3, 4	31 of 38
Algerian Arabic (ARQ)*	microsoft/mdeberta-v3-base	$5e^{-5}$	0.001	0.5	2, 3, 4	**30 of 44**
Amharic (AMH)*	FacebookAI/xlm-roberta-base	$2e^{-5}$	0.01	0.5	1, 2, 3, 4	22 of 44
Chinese (CHN)*	microsoft/mdeberta-v3-base	$5e^{-5}$	0.1	0.5	2, 3, 4	**26 of 42**
Emakhuwa (VMW)***	cis-lmu/glotlid	$1e^{-5}$				**17 of 32**
English (ENG)*	facebook/bart-large-mnli	$5e^{-5}$	0.01	0.5	1	**21 of 97**
German (DEU)*	benjamin/roberta-base-wechsel-german	$5e^{-5}$	0.1	0.5	1, 2, 3	20 of 51
Hausa (HAU)**	FacebookAI/xlm-roberta-base	$5e^{-5}$	0.01	0.3	1, 2, 4	**21 of 41**
Hindi (HIN)**	ai4bharat/IndicBERTv2-MLM-only	$1e^{-5}$	0.01	0.5	1, 2, 4	**14 of 44**
Igbo (IBO)*	castorini/afriberta_large	$2e^{-5}$	0.01	0.5	1, 2, 3	**3 of 35**
Kinyarwanda (KIN)*	castorini/afriberta_large	$5e^{-5}$	0.1	0.5	2, 3	**9 of 32**
Marathi (MAR)*	Twitter/twhin-bert-base	$1e^{-5}$	0.1	0.5	1, 3	30 of 43
Moroccan Arabic (ARY)**	SI2M-Lab/DarijaBERT	$5e^{-5}$	0.01	0.5	2, 3, 4	**22 of 42**
Nigerian-Pigdin (PCM)*	castorini/afriberta_large	$5e^{-5}$	0.1	0.5	2, 3, 4	23 of 35
Oromo (ORM)**	castorini/afriberta_large	$5e^{-5}$	0.1	0.5	2, 3	**4 of 37**
Portuguese (Brazilian) (PTBR)*	neuralmind/bert-large-portuguese-cased	$2e^{-5}$	0.001	0.5	1, 2, 3, 4	**17 of 43**
Portuguese (Mozambican) (PTMZ)**	neuralmind/bert-large-portuguese-cased	$5e^{-5}$	0.001	0.5	1, 2, 3, 4	**12 of 37**
Romanian (RON)*	FacebookAI/xlm-roberta-base	$2e^{-5}$	0.001	0.5	1, 4	25 of 44
Russian (RUS)*	DmitryPogrebnoy/distilbert-base-russian-cased	$3e^{-5}$	0.1	0.5	1, 4	37 of 51
Somali (SOM)**	FacebookAI/xlm-roberta-base	$5e^{-5}$	0.1	0.5	2	25 of 35
Spanish (Latin American) (ESP)*	dccuchile/bert-base-spanish-wwm-cased	$3e^{-5}$	0.01	0.5	1	31 of 48
Sundanese (SUN)**	w11wo/sundanese-roberta-base	$5e^{-5}$	0.01	0.5	2, 3, 4	**25 of 38**
Swahili (SWA)*	microsoft/mdeberta-v3-base	$2e^{-5}$	0.001	0.5	2, 3, 4	**20 of 33**
Swedish (SWE)*	FacebookAI/xlm-roberta-base	$5e^{-5}$	0.01	0.5	1, 2, 3, 4	31 of 41
Tatar (TAT)*	google-bert/bert-base-multilingual-cased	$5e^{-5}$	0.1	0.5	1, 2, 3	**18 of 37**
Tigrinya (TIR)**	Davlan/afro-xlmr-large-76L	$2e^{-5}$	0.1	0.5	1, 2, 3	**2 of 35**
Ukranian (UKR)*	google-bert/bert-base-multilingual-cased	$5e^{-5}$	0.1	0.5	1, 2, 3	31 of 41
Yoruba (YOR)*	castorini/afriberta_large	$2e^{-5}$	0.1	0.5	2, 3, 4	**6 of 33**

Table 4: Configuration to final training for Track A. The best combination of custom loss functions used for the competition phase in each language are coded as: *BCEWithLogitsLoss*=1, *CrossEntropyLoss*=2, *FocalLoss*=3, *LabelSmoothingLoss*=4.
*Final submission with the best hyperparameters and custom loss functions combination.
**Final submission only with the best hyperparameters (without custom loss functions combination).
***The model was trained differently due to the structure of the pre-trained model used.
Note: The unofficial ranking results shown in **bold** exceeded the organizers' baseline.

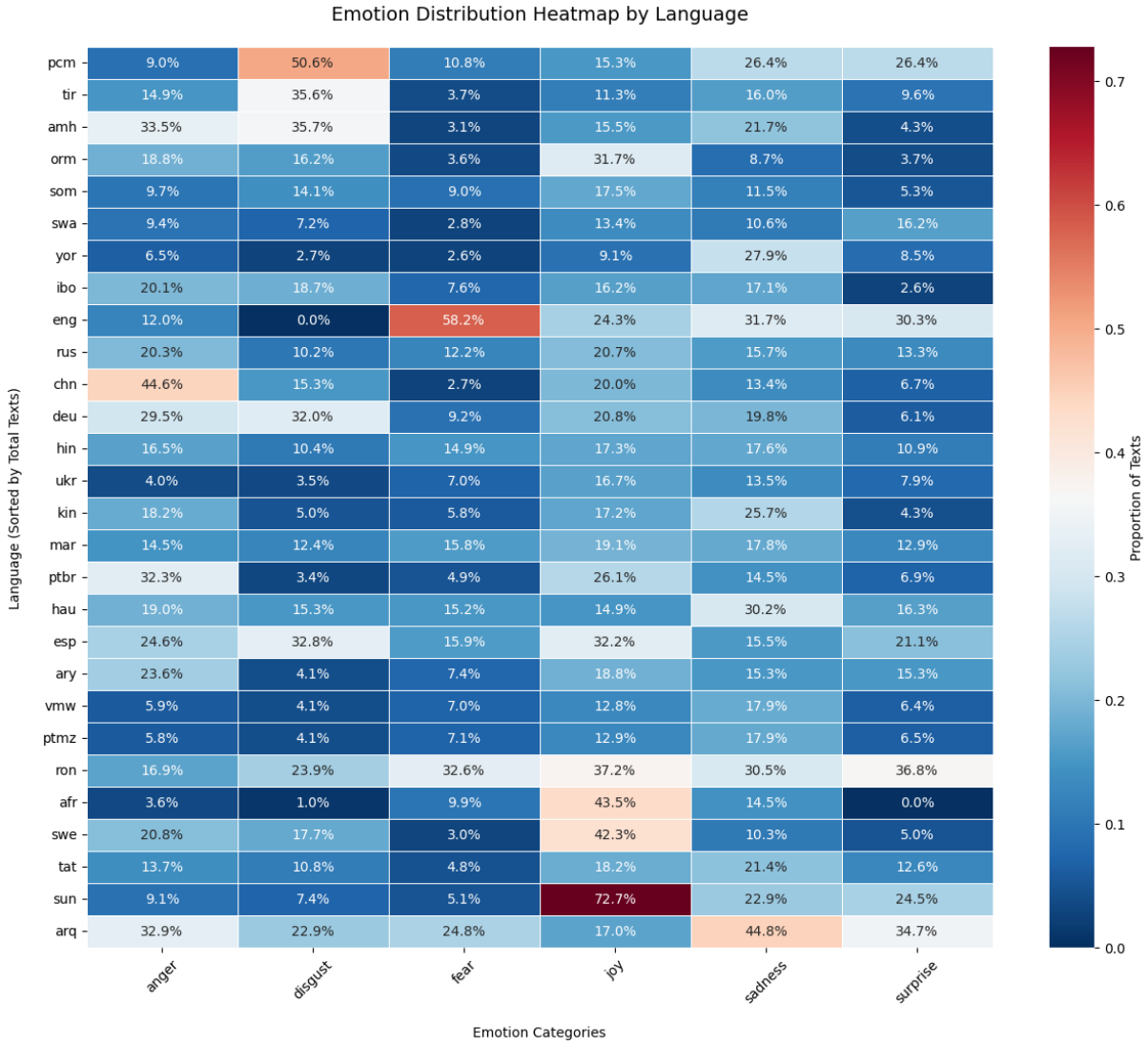

Figure 2: Heatmap of the distribution of emotions by language in the Track A training dataset. Warmer colors indicate higher prevalence.
Note: English lacks the emotion of disgust, and Afrikaans lacks the surprise emotion, so the corresponding cells of the heat map show 0.0%