TVKS@MMLow 2026: Factorized Stance × Target Classification of Tamil Political Memes with an Axis-Wise Multimodal Ensemble

Kevin Selva K and Thilagavathy R
Department of Computing Technologies, School of Computing,
Faculty of Engineering and Technology,
SRM Institute of Science and Technology, Kattankulathur – 603203, Tamil Nadu, India
ks8531@srmist.edu.in, thilagar2@srmist.edu.in

Abstract

We describe the TVKS submission to the MMLow 2026 shared task on multilevel classification of Tamil political memes, which asks systems to predict a stance label (Level 1: Support/Praise vs. Troll/Oppose) and a four-way target label (Level 2) that combines stance with a Person/Party distinction. We factorize the Level 2 label into two binary axes—stance and person/party target—reconstructing Level 2 as their product rather than predicting four classes directly. A flat four-way head drives the rarest class (Troll/Oppose Against Party) to an F1 of 0.000; factorization recovers a non-degenerate prediction over all four classes. We also fix a calibration failure caused by stacking a class-balanced sampler with a class-weighted loss, and combine two multimodal models (MuRIL + ViT and MuRIL + ResNet-50) in an axis-wise ensemble that routes each binary axis to its stronger model. Our system reaches a macro-F1 of 0.8594 on Level 1 and 0.5926 on Level 2 (mean 0.7260), ranking second of seven teams. An error analysis attributes 77% of Level 2 errors to the person/party axis, where architecturally distinct visual encoders converge to the same ceiling—evidence that the residual gap is dataset ambiguity rather than a modelling deficiency.[1]

1 Introduction

Political memes have become a dominant medium of online political communication. Their multimodal composition—an image, embedded text, symbols, and cultural context—lets them convey a stance even to audiences with limited command of the underlying language, which also makes them difficult to analyse automatically, especially for low-resource and code-mixed languages such as Tamil. The MMLow 2026 shared task (Rajiakodi et al., 2026) frames this as hierarchical classification of Tamil memes along two levels: a binary stance (Level 1) and a four-way target that pairs stance with whether the meme is directed at a person or a party (Level 2). Both levels are scored independently by macro-averaged precision, recall, and F1.

A naive solution predicts the four Level 2 classes with a single softmax head. On this dataset that choice is actively harmful: the rarest class, Troll/Oppose Against Party (146 of 969 training examples), collapses to an F1 of 0.000, and no amount of threshold tuning recovers a class the head never predicts. Our system is built around avoiding this failure mode.

Our contributions are: (i) a factorized stance × target formulation that decomposes Level 2 into two binary axes and reconstructs it as their product, preventing the rare-class collapse of a flat head; (ii) a diagnosis and fix of a calibration failure that arises from stacking two independent class-balancing mechanisms; (iii) an axis-wise ensemble, enabled by the factorization, that routes stance and target to different multimodal models; and (iv) an error analysis with perfect-axis ceilings and three converging negative results that localise the residual gap to the person/party axis and characterise it as a data ceiling rather than a modelling limitation.

2 Related Work

Multimodal meme classification was popularised by the Hateful Memes Challenge (Kiela et al., 2020), which established the joint modelling of image and embedded text as a distinct problem from unimodal classification. For Dravidian languages, prior shared tasks have targeted Tamil memes specifically: the TamilMemes troll corpus (Suryawanshi et al., 2020) and the associated shared task on troll meme classification (Suryawanshi and Chakravarthi, 2021), as well as multimodal meme classification for Tamil and Malayalam more broadly (Chakravarthi et al., 2024). Our

[1] Code: https://github.com/KevinselvaK/tvks-mmlow2026.

Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages, pages 70–75
November 9, 2026 ©2026 Association for Computational Linguistics

task differs in requiring a two-level hierarchical output combining stance and target. On the modelling side, severe class imbalance is commonly addressed with re-weighting or re-sampling, and with loss functions such as focal loss (Lin et al., 2017); these remedies are output-agnostic, easing learning of the rare class without changing what is learned, since a single softmax still allocates mass across all classes from one shared representation. Our factorization instead changes the output structure itself: decomposing the rare four-way label into two well-populated binary axes means the rarest joint class (146/969 examples) need not compete directly against classes it is outnumbered by 3.7:1 in a shared softmax—closer in spirit to structured/hierarchical output decomposition than to reweighting. Our ablation (Table 3) isolates this effect from calibration and ensembling, showing the F1 = 0.000 → 0.438 jump is attributable to the restructuring alone. We build on MuRIL (Khanuja et al., 2021) for text and on ViT (Dosovitskiy et al., 2021) and ResNet-50 (He et al., 2016) for images.

3 Task and Data

3.1 Task Definition

Each meme is annotated at two levels. Level 1 is a binary stance: Support/Praise vs. Troll/Oppose. Level 2 has four classes formed by pairing the stance with a Person/Party target: {Support for person, Support for party, Troll/Oppose Against Person, Troll/Oppose Against Party}. Systems are evaluated per level with scikit-learn's macro-averaged report (Pedregosa et al., 2011). We treat the two levels as separate scored tasks, in line with the evaluation.

3.2 Data and Label Distribution

The Tamil training set contains 969 labelled memes and the test set 244 unlabelled memes (Rajiakodi et al., 2026).[2] Table 1 gives the label distribution. Level 2 is severely imbalanced: Troll/Oppose Against Person dominates (547) while Support for party is rarest (99), and the party target is a global minority (245 vs. 724 person-target memes). Six rows have empty text; five rows have a Level 1/Level 2 stance mismatch. Because of those five hierarchy violations, we derive both axis labels directly from the Level 2 string—stance from whether the label begins with "Support", target from whether it contains "party"—so that each axis is trained and scored against its own consistent ground truth rather than an occasionally contradictory Level 1 column. Because official test labels are not released, we cannot directly measure whether this relabeling altered the OOF–test ranking; the small OOF→test drop reported in Table 2 (§5) is consistent with no adverse effect, but this is not a direct verification.

[2]A regex resolver maps each integer image id to its file, tolerating zero-padding and prefixes. Training rows whose image cannot be resolved are dropped; all counts reported here are annotation counts. Every test id is predicted, using a gray placeholder image for any unresolved test file.

Class	Level 1	Level 2
Support/Praise	277	
Support for person		177
Support for party		99
Troll/Oppose	692	
Troll/Oppose Against Person		547
Troll/Oppose Against Party		146
Person-target (all)	724	
Party-target (all)	245	
Total (train)	969	
Total (test)	244	

Table 1: Training-set label distribution; the test set (244 memes) is unlabelled. The rarest Level 2 class (Support for party) has 99 examples and the party target axis is a minority (245 vs. 724). Five rows have a Level 1/Level 2 stance mismatch and six have empty text.

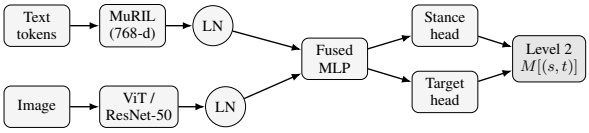


Figure 1: Factorized multimodal architecture. Text (MuRIL) and image (ViT or ResNet-50) are encoded separately, normalised per modality (LN), fused by an MLP, and read out by two binary heads. Level 2 is reconstructed from the (stance, target) pair via Eq. 1.

4 System

4.1 Axis Factorization

Instead of a four-way head, we predict two binary variables: stance $s \in \{0, 1\}$ (support/troll) and target $t \in \{0, 1\}$ (person/party). Level 2 is reconstructed deterministically from the pair,

$$\hat{y}_2 = M\big[(\hat{s}, \hat{t})\big], \tag{1}$$

where M maps each of the four (s, t) combinations to its Level 2 string. This is the same information a flat head must learn, but expressed

so that every combination—including the rare Troll/Oppose Against Party (s=1, t=1)—remains reachable whenever both axes fire, regardless of how few joint examples exist. A flat softmax, by contrast, must allocate probability mass to a class it sees 146 times against competitors it sees up to 547 times, and in practice never predicts it (F1 = 0.000). Figure 1 shows the resulting architecture.

4.2 Multimodal Encoders

Both models pair a MuRIL text encoder (768-dim) with an image encoder. MuRIL is pretrained on Indian languages including transliterated and code-mixed text, so it handles the Tamil–English (Tanglish) code-mixing present in these memes without special preprocessing. Model M1F uses a ViT-B/16 encoder and M2F-v2 uses ResNet-50 (2048-dim pooled features). We take the text `[CLS]` embedding and the image representation, apply a per-modality LayerNorm to each, concatenate, and pass the result through a two-layer fused MLP feeding the two binary heads. Per-modality normalisation is not cosmetic: without it, the 2048-dim unnormalised ResNet features dominate the 768-dim MuRIL embedding and the stance axis collapses toward a near-constant prediction, sharply degrading Level 1 macro-F1. Normalising each modality to a common scale, together with dual learning rates—2×10^{-5} for the pretrained encoders and 1×10^{-3} for the fusion and heads—keeps both modalities contributing. We use per-modality LayerNorm rather than learned linear projections because the mismatch is one of *scale*, not *subspace*: both encoders produce pooled, semantically comparable representations differing by roughly an order of magnitude in norm, which LayerNorm corrects with no added parameters and no projection-overfitting risk given our modest label budget. MuRIL, ViT-B/16, and ResNet-50 were chosen deliberately: MuRIL's pretraining on transliterated, code-mixed Indian-language text handles Tanglish without preprocessing, while ViT and ResNet-50 are architecturally distinct (global attention vs. local convolution) so the ensemble in §4.4 combines complementary, not redundant, visual errors.

4.3 Calibration: Removing Compound Balancing

Our first ResNet model was badly miscalibrated on stance, systematically over-predicting Support. The cause was compound balancing: the training sampler already rebalanced stance toward 50/50, and the cross-entropy loss was also class-weighted toward the minority stance, so the two corrections stacked and pushed predictions past the decision boundary. The fix is to apply exactly one balancing mechanism to stance. We keep a softened sampler (sampling weights raised to the power 0.5, which partially rebalances without fully flattening the prior) and make the stance cross-entropy unweighted; the target cross-entropy remains class-weighted, since the sampler does not balance the target axis. This single change raised Level 1 macro-F1 at the default threshold from 0.649 to 0.839—a calibration gain, obtained with no threshold tuning.

4.4 Axis-Wise Ensemble and Operating Points

Factorization lets us ensemble each axis independently. Given per-axis probabilities from the two models, we blend

$$s = w_s\, s_1 + (1{-}w_s)\, s_2, \quad t = w_t\, t_1 + (1{-}w_t)\, t_2, \tag{2}$$

and select the blend weights (w_s, w_t) together with per-level decision thresholds to maximise each level's macro-F1 on cross-validated out-of-fold (OOF) predictions. The selected operating points are $w_s = 0.70$ and $w_t = 0.70$, both favouring M1F (the ViT-based model), with Level 1 decision threshold $T_1 = 0.31$ and Level 2 thresholds $T_{2S} = 0.32$ (stance) and $T_{2T} = 0.48$ (target). Both axes draw more heavily on M1F, though the ensemble still meaningfully incorporates M2F-v2 on both—reflecting complementary rather than one-model-dominant error patterns, and precisely what the factorization makes possible by weighting each axis independently. We also tested a CLIP-based target specialist (Radford et al., 2021): added to the target axis it did not improve held-out Level 2 ($\Delta \approx -0.007$) and was excluded from the final system. All operating points are fixed on OOF and applied unchanged to the test set.

4.5 Training Protocol

We use 5-fold stratified cross-validation on the four-way Level 2 label (seed 42), selecting all models and operating points by OOF macro-F1 and never by test performance. Models are trained with the HuggingFace Transformers library (Wolf et al., 2020), AdamW with decoupled weight decay (Loshchilov and Hutter, 2019), a OneCycle schedule, and mixed precision. Training is check-

	Level 1	Level 2	Final
Dev (5-fold OOF)	0.886	0.645	0.766
Official test	0.859	0.593	0.726
Rank		2 / 7	

Table 2: System performance (macro-F1). Dev scores are 5-fold OOF estimates used for model selection (bootstrap 95% CIs: Level 1 [0.862, 0.909], Level 2 [0.610, 0.682]); test scores are the official leaderboard. The small OOF→test drop shows the OOF-selected operating points generalised.

Configuration	Level 1	Level 2
Flat 4-way head	—	0.381
+ axis factorization	0.649	0.438
+ calibration fix	0.839	0.612
+ axis-wise ensemble	**0.886**	**0.645**

Table 3: Development ablation (5-fold OOF macro-F1). Under the flat four-way head the rarest class (Troll/Oppose Against Party) collapses to F1 = 0.000; factorization restores a non-degenerate distribution over all four classes, reflected in the Level 2 gain. The flat head's Level 1 is obtained by marginalising its four-way output and is omitted as not directly comparable.

pointed per fold so runs resume after interruption, and only compact per-fold prediction arrays are retained. Full hyperparameters are in Appendix A.

5 Results

5.1 Main Results

Table 2 reports our development and official results. Our submission placed second of seven teams, with a test macro-F1 of 0.8594 on Level 1 and 0.5926 on Level 2 (mean 0.7260). The OOF→test drop is small (-0.027 on Level 1, -0.052 on Level 2), indicating that operating points selected on cross-validated OOF transferred to unseen data without overfitting. The top-ranked system (final score 0.7860) exceeded ours by 0.060; roughly two-thirds of the raw difference lies on Level 2 rather than stance, i.e. on exactly the person/party axis our error analysis (§6) identifies as the ceiling.

5.2 Ablation

Table 3 traces each design decision on OOF. The flat four-way head drives the rarest class (Against-Party) to F1 = 0.000; factorization lifts that class off zero and improves Level 2 overall. Removing the redundant stance weighting then recalibrates and improves both levels, and the axis-wise ensemble adds a further gain by routing each axis to its stronger model.

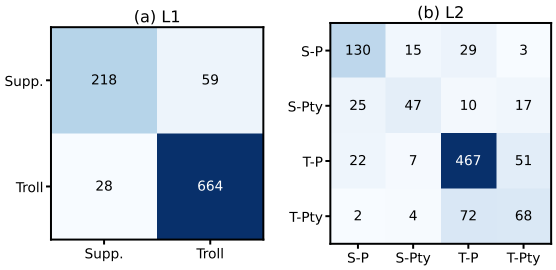


Figure 2: OOF confusion matrices: (a) Level 1, (b) Level 2 (S-P/S-Pty = Support person/party; T-P/T-Pty = Troll/Oppose Against person/party). Off-diagonal mass sits mainly between the two Troll/Oppose classes, confirming target-, not stance-axis, confusion drives residual error (§6).

6 Error Analysis

Because the factorization exposes each axis separately, we can decompose Level 2 errors exactly. On OOF, the stance axis reaches 0.922 accuracy while the target axis reaches 0.810, and 77% of all Level 2 errors are target-axis (person-vs-party) failures. Figure 2 shows the confusion matrices for both levels, visually confirming that target confusion, not stance confusion, is the dominant error mode.

The confusion is strongly one-directional: party-target memes are frequently predicted as person-target, while the reverse is rare—the model defaults to the majority (person) target under uncertainty. Quantitatively, 44.5% of true party-target memes (109/245) are misclassified as person-target, versus only 10.5% of true person-target memes (76/724) misclassified as party-target—a more than four-fold asymmetry.

The highest-confidence errors support the annotator-intent explanation directly: several Against-Party memes name a specific office-holder prominently (e.g., a welfare-promise meme naming the sitting Chief Minister), which the model reads as person-directed despite the party-coded label; conversely, some Support-for-person memes display a party name or acronym that pulls the model toward a party-target prediction. In both directions, the surface cue that seems most informative—a named person or party string—points to the wrong axis relative to the annotation.

The perfect-axis ceilings make the headroom

explicit. If the target axis were perfect, Level 2 would reach 0.891; if the stance axis were perfect, only 0.734. The target axis is unambiguously the bottleneck.

Three converging negative results bound that ceiling. First, a party-name lexicon is non-discriminative: party tokens appear at nearly equal rates in person-target and party-target memes, so surface party mentions do not signal the target. Second, named-entity features—politician-to-party mappings and multi-camp conflict detection—carry no discriminative power for the person/party distinction. Third, two architecturally distinct visual encoders, ResNet-50 and CLIP, plateau at essentially the same target-axis F1. Together these indicate that whether a meme targets a person or their party is largely an annotator intent judgment rather than a property recoverable from the image or text surface—a dataset-level ceiling rather than a deficiency of any one model. Region-specific vision backbones pretrained on Indian political iconography and OCR-based on-image text extraction are two concrete, low-cost directions for probing this ceiling further without new annotation.

7 Conclusion

We presented an axis-factorized multimodal system for hierarchical classification of Tamil political memes. Factorizing Level 2 into stance and target axes prevents the rare-class collapse of a flat head; removing a compound-balancing calibration bug recovers stance calibration without threshold tuning; and an axis-wise ensemble routes each axis to its stronger model. The system ranked second of seven teams. Our analysis localises the remaining gap to the person/party axis and gives converging evidence that it is data-bound. Promising directions are entity-aware pretraining and external political knowledge to inform the target judgment, larger annotated data, and sharper annotation guidelines for the person/party boundary.

Limitations

Our study covers a single language (Tamil) and a single dataset of modest size (969 training memes), so the findings may not transfer to other languages or domains; we did not evaluate on a second dataset, and generalization to other political-meme corpora (different parties, time periods, or annotation schemes) remains untested. Because the official test labels are not released, our per-class and axis-decomposed error analysis is computed on cross-validated OOF predictions rather than the test set; the perfect-axis ceilings are therefore estimates of headroom on development data. We did not conduct an inter-annotator agreement study, so the claim that the person/party distinction is annotator intent is inferred from converging negative results rather than measured directly.

Ethical Considerations

The dataset consists of real political memes and reflects a specific period and discourse; our models predict expressed stance and target, not the truth or legitimacy of any claim. Systems trained on such data could be misused for censorship or surveillance of political speech, and could encode the biases of the annotation process. We release our work for research on multimodal understanding of low-resource political discourse and do not advocate its use for automated moderation without human oversight.

Acknowledgments

Generative AI assistance was used for grammatical and language polishing of the manuscript text. All AI-assisted text was reviewed and verified by the authors, who take full responsibility for the accuracy of the final manuscript.

References

Bharathi Raja Chakravarthi, Saranya Rajiakodi, Rahul Ponnusamy, Kathiravan Pannerselvam, Anand Kumar Madasamy, Ramachandran Rajalakshmi, Hariharan LekshmiAmmal, Anshid Kizhakkeparambil, Susminu S Kumar, Bhuvaneswari Sivagnanam, and Charmathi Rajkumar. 2024. Overview of shared task on multitask meme classification – unraveling misogynistic and trolls in online memes. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 139–144, St. Julian's, Malta. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR), pages 770–778.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. MuRIL: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 2611–2624.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 8748–8763.

Saranya Rajiakodi, Shunmuga Priya Muthusamy Chinnan, Premjith B, Subalalitha CN, Rahul Ponnusamy, Anshid K A, Bhuvaneswari Sivagnanam, Jananayagan V, Ragavan N, Santhini P, and Bharathi Raja Chakravarthi. 2026. Overview of the shared task on multilevel political meme classification in Tamil and Malayalam. In *Proceedings of the Sixth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 101–113, Underline (Virtual). Association for Computational Linguistics.

Shardul Suryawanshi and Bharathi Raja Chakravarthi. 2021. Findings of the shared task on troll meme classification in Tamil. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 126–132.

Shardul Suryawanshi, Bharathi Raja Chakravarthi, Pranav Verma, Mihael Arcan, John Philip McCrae, and Paul Buitelaar. 2020. A dataset for troll classification of TamilMemes. In *Proceedings of the WILDRE5–5th Workshop on Indian Language Data: Resources and Evaluation*, pages 7–13, Marseille, France. European Language Resources Association (ELRA).

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

A Hyperparameters

Selected operating points (OOF). Stance blend $w_s = 0.70$; target blend $w_t = 0.70$ (both favouring M1F); Level 1 decision threshold $T_1 = 0.31$; Level 2 thresholds $T_{2S} = 0.32$ (stance) and $T_{2T} = 0.48$ (target). These values are chosen on cross-validated OOF and applied unchanged to the test set.

Setting	Value
Text encoder	MuRIL (`muril-base-cased`)
Image encoder	ViT-B/16 (M1F) / ResNet-50 (M2F-v2)
Folds	5 (StratifiedKFold on Level 2)
Epochs	8
Batch size	16
Max text length	128
Image size	224
LR (text / image)	2×10^{-5} / 2×10^{-5}
LR (fusion / heads)	1×10^{-3}
Optimizer / sched.	AdamW / OneCycle (warmup 0.1)
Sampler power	0.5 (stance)
Stance CE	unweighted
Target CE	class-weighted
Grad clip	1.0
Seed	42
Blend weights (w_s, w_t)	0.70, 0.70
Thresholds (T_1; T_{2S}, T_{2T})	0.31; 0.32, 0.48

Table 4: Full training configuration.