MMLow 2026

# Fifth Workshop on Multimodal Machine Learning in Low-resource Languages, AACL-IJCNLP 2026

## Proceedings of the Conference

November 9, 2026

The MMLow organizers gratefully acknowledge the support from the following sponsors.









©2026 Association for Computational Linguistics

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-445-3

# Introduction

Multimodal machine learning has advanced significantly in recent years, driven by rapid progress in computer vision, speech processing, and natural language processing. However, much of this progress has focused on high-resource languages, particularly English. The availability of labelled datasets, benchmarks, and computational tools remains limited for low-resource languages, especially those spoken across Asia, Africa, Europe, and South America. This scarcity presents significant challenges for developing robust multimodal systems capable of understanding and processing content in regional and underrepresented languages. The Fifth Workshop on Multimodal Machine Learning in Low-resource Languages (MMLow 2026), co-located with AACL-IJCNLP 2026 in Hengqin, China, continues a workshop series initiated in 2022. The workshop provides a forum for researchers and practitioners to exchange ideas and present advances in multimodal data fusion, representation learning, and applications involving regional and low-resource languages. It aims to address the persistent scarcity of datasets, tools, and evaluation resources while fostering research directions that support future developments and real-world applications. The workshop welcomed submissions on a broad range of topics, including multimodal sentiment and emotion analysis in regional languages; hate speech, misogyny, offensive content, and troll or political meme detection; multimodal data fusion and representation learning; benchmark datasets and evaluation methodologies for low-resource multimodal tasks; fake news and misinformation detection in regional languages; speech-text and vision-language modelling for low-resource languages; and data collection and annotation methods for safer social media and content moderation. MMLow 2026 also featured two shared tasks: Multi-Level Political Meme Classification, which focused on classifying political memes at multiple levels, and Identifying Misogynistic Videos in Dravidian Languages, which addressed the detection of misogynistic content in videos featuring Dravidian languages. These shared tasks encouraged systematic evaluation and provided opportunities for researchers to explore practical solutions to challenging multimodal language understanding problems. The workshop received 18 submissions, of which 11 papers were accepted following a rigorous review process, corresponding to an acceptance rate of 61.1%. We sincerely thank all authors and shared-task participants for their valuable contributions. We are also grateful to the programme committee members for their timely, and constructive reviews, and to the AACL-IJCNLP 2026 workshop chairs for their guidance and support.

# Organizing Committee

### Workshop Organizers

Rahul Ponnusamy, University of Galway, Ireland
Prasanna Kumar Kumaresan, University of Galway, Ireland
Saranya Rajiakodi, Central University of Tamil Nadu, Thiruvarur, India
Premjith B, Amrita Vishwa Vidyapeetham, Coimbatore, India
Shunmuga Priya MC, University of Galway, Ireland
Abirami Murugappan, Anna University, Tamil Nadu, India
Jyothish Lal G, Amrita Vishwa Vidyapeetham, Coimbatore, India
Sowmya V, Amrita Vishwa Vidyapeetham, Coimbatore, India
Bharathi Raja Chakravarthi, University of Galway, Ireland

# Program Committee

**Program Committee**

Bhuvaneshwari M, Kumaraguru College of Technology, India
Jairam R, Amrita Vishwa Vidyapeetham, India
Thomas Mandl, Universität Hildesheim
Rajesh Balaji, Costco Wholesale Corporation
Divya Sasidharan, Amrita Vishwa Vidyapeetham, India
Hariharan R L, Vellore Institute of Technology
Shreyash Rawat, Facebook
Anisha Ramakrishna Yarlapati, Adobe Systems
Priyanka C, Amrita Vishwa Vidyapeetham, India
Suryaa Veerabathiran Seran, Adobe Systems
Manjula R, SRM Institute of Science and Technology
Abhishek Anilkumar, Amrita Vishwa Vidyapeetham, India
Archudha Arjunasamy, Amrita Vishwa Vidyapeetham, India
Yokesh Dhanabal, Kumaraguru College of Technology
Anita R, SRM Institute of Science and Technology
Ipsita Mohanty, Amazon
Kathiresan Jayabalan, Sathyabama Institute of Science and Technology
Krishnakumari K, University College of Engineering, Panruti
Ravindrakumar M. Purohit, Dhirubhai Ambani Institute of Information and Communication Technology
Shalmiya Paulraj, Kumaraguru College of Technology, India
Naman Ahuja, Facebook
Rizwana Kallooravi Thandil, Sullamussalam Science College Areekode
Kamal Kishore Chigullarevu Thirumala, Stanford University
Rathipriya Easwaramoorthy, Amrita Vishwa Vidyapeetham, India
Divyabharathi R, Amrita Vishwa Vidyapeetham, India
Bagavathi C, Amrita Vishwa Vidyapeetham, India

# Table of Contents

*Script Fragmentation and Format: What Drives the English-Bengali Performance Gap in Open LLMs?*
Shimanto Bhowmik, Tawsif Tashwar Dipto, Md Sazzad Islam, Sheryl Hsu and Tahsin Reasat . . 1

*A Foundational Study of Multimodal Speech–Text Alignment in Malayalam*
Meharuniza Nazeem, Navaneeth S, Adithyakiran A S, Rajeev R R and Ashbi Biju . . . . . 18

*A Retrieval-Augmented Generation Approach for Enhancing Pedagogical Support in Education*
Ramkumar N and Rajendran S . . . . . 28

*One Head, Three Transcripts: A Controlled Oracle-to-ASR Audit of Bangla Speech–Text Sentiment Analysis*
Md Raihan, Nilam Rajak, Kuntal Ghosh and Sathyanarayanan Dhorali . . . . . 40

*MOEZHI-GURU: Multi-Perspective Misogynistic Meme Detection via Multimodal Mixture of Experts*
Sekar M, Pranava Mohan and A. Santhanavijayan . . . . . 55

*Axiom@MMLow 2026: Cascaded Prompting with QLoRA-Tuned Qwen2-VL for Hierarchical Tamil Political Meme Classification*
Md. Ajwad Hossain . . . . . 64

*TVKS@MMLow 2026: Factorized Stance × Target Classification of Tamil Political Memes with an Axis-Wise Multimodal Ensemble*
Kevin selva K and Thilagavathy R . . . . . 70

*Projukti Kavalan@MMLow 2026: Weighted Visual Frame and Cross-modal Fusion for Misogynistic Video Detection in Dravidian Language*
Palaniraj I and Dwaipayan Dutta . . . . . 76

*KEC@MMLow 2026: Multimodal Fusion for Hierarchical Tamil Political Meme Classification*
Asvitha RM and Amirthamozhi VS . . . . . 86

*CPSLab2@MMLow 2026: A Hierarchical Multimodal Framework with Stratified K-Fold Training for Political Meme Classification*
Anika Srivastava, Vishal Kumar, Harshita Sharma, Nitisha Aggarwal, Geetika Jain Saxena, Amit Pundir and Sanjeev Singh . . . . . 91

*CPS Lab 1@MMLow 2026: Hierarchical Multimodal Classification of Tamil Political Memes via Logit-Conditioned Stance-Target Prediction*
Vishal Kumar, Anika Srivastava, Harshita Sharma, Nitisha Aggarwal, Geetika Jain Saxena, Amit Pundir and Sanjeev Singh . . . . . 97

# Program

**Monday, November 9, 2026**

09:00 - 09:15 *Opening Remarks*

09:15 - 10:30 *Session 1: Oral*

*Script Fragmentation and Format: What Drives the English-Bengali Performance Gap in Open LLMs?*
Shimanto Bhowmik, Tawsif Tashwar Dipto, Md Sazzad Islam, Sheryl Hsu and Tahsin Reasat

*A Foundational Study of Multimodal Speech–Text Alignment in Malayalam*
Meharuniza Nazeem, Navaneeth S, Adithyakiran A S, Rajeev R R and Ashbi Biju

*One Head, Three Transcripts: A Controlled Oracle-to-ASR Audit of Bangla Speech–Text Sentiment Analysis*
Md Raihan, Nilam Rajak, Kuntal Ghosh and Sathyanarayanan Dhorali

*A Retrieval-Augmented Generation Approach for Enhancing Pedagogical Support in Education*
Ramkumar N and Rajendran S

*MOEZHI-GURU: Multi-Perspective Misogynistic Meme Detection via Multimodal Mixture of Experts*
Sekar M, Pranava Mohan and A. Santhanavijayan

10:30 - 11:00 *Coffee Break*

11:00 - 12:00 *Session 2: Poster*

*Axiom@MMLow 2026: Cascaded Prompting with QLoRA-Tuned Qwen2-VL for Hierarchical Tamil Political Meme Classification*
Md. Ajwad Hossain

*TVKS@MMLow 2026: Factorized Stance $\times$ Target Classification of Tamil Political Memes with an Axis-Wise Multimodal Ensemble*
Kevin selva K and Thilagavathy R

*Projukti Kavalan@MMLow 2026: Weighted Visual Frame and Cross-modal Fusion for Misogynistic Video Detection in Dravidian Language*
Palaniraj I and Dwaipayan Dutta

**Monday, November 9, 2026 (continued)**

*KEC@MMLow 2026: Multimodal Fusion for Hierarchical Tamil Political Meme Classification*
Asvitha RM and Amirthamozhi VS

*CPSLab2@MMLow 2026: A Hierarchical Multimodal Framework with Stratified K-Fold Training for Political Meme Classification*
Anika Srivastava, Vishal Kumar, Harshita Sharma, Nitisha Aggarwal, Geetika Jain Saxena, Amit Pundir and Sanjeev Singh

*CPS Lab 1@MMLow 2026: Hierarchical Multimodal Classification of Tamil Political Memes via Logit-Conditioned Stance-Target Prediction*
Vishal Kumar, Anika Srivastava, Harshita Sharma, Nitisha Aggarwal, Geetika Jain Saxena, Amit Pundir and Sanjeev Singh

12:00 - 12:30 *Closing Remarks*

12:30 - 14:00 *Lunch Break*

# Script Fragmentation and Format: What Drives the English-Bengali Performance Gap in Open LLMs?

**Shimanto Bhowmik**
Rochester Institute of Technology
sb6778@g.rit.edu

**Tawsif Tashwar Dipto**
Islamic University of Technology
tawsiftashwar@iut-dhaka.edu

**Md Sazzad Islam**
Stanford University
sazzad14@stanford.edu

**Sheryl Hsu**
Stanford University
sherylh@stanford.edu

**Tahsin Reasat**
Bengali.AI
reasat@bengali.ai

## Abstract

Bengali is spoken by more than 230 million people, yet no standardized instrument evaluates large language models (LLMs) on Bengali across the task categories used to benchmark frontier models. We release 8 English benchmarks translated into Bengali with a single consistent pipeline and use them to evaluate 10 open LLMs from 4 families on paired English and Bengali inputs. *Script fragmentation* is what subword tokenizers do to Bengali's alphasyllabary, whose written units are grapheme clusters spanning several Unicode code points: they cut the script into pieces smaller than a character, at a cost set by the vocabulary rather than the script itself. *Format* belongs to the evaluation, the answer shape that exact-match scoring demands regardless of whether the model knew the answer. Beyond confirming a substantial gap (macro LLM-judge score 0.79 in English versus 0.63 in Bengali), we show that part of it is a measurement artifact: exact-match accuracy conflates correctness with format adherence and because format failure is asymmetric across languages it distorts the gap for some models three to five fold and even reverses its sign for one reasoning-tuned model. On the fragmentation side, Bengali costs roughly five times more tokens per word than English, the 10 models share only 4 vocabularies and that cost varies twofold across them: the same Bengali word averages 7.9 tokens under the Llama 3 tokenizer but 4.0 under Mistral's Tekken. Under three of the four vocabularies the average Bengali token covers fewer bytes than a single code point. Bengali is the constant here and the vocabularies are not and so what Bengali text costs is set by tokenizer design rather than by the script. Fertility and sequence length correlate only weakly with scores ($r = -0.23$), so we present this as a cost and segmentation concern rather than a driver of the gap. The datasets, pipeline and evaluation code are released.

## 1 Introduction

Open large language models are now routinely compared on a standard set of English benchmarks covering commonsense reasoning, science questions, mathematics and broad knowledge (Grattafiori et al., 2024; Yang et al., 2025). For Bengali, one of the most widely spoken languages in the world, no comparable evaluation suite exists. This is not because Bengali lacks data or models: it has dedicated pretrained models such as BanglaBERT (Bhattacharjee et al., 2022), BanglaGPT (Salim et al., 2023), TituLLM (Nahin et al., 2025) and TigerLLM (Raihan and Zampieri, 2025) and roughly 30 billion tokens of curated pretraining text in the Sangraha corpus (Khan et al., 2024). What Bengali lacks is an *instrument*: a set of evaluation benchmarks that spans the same task categories as the English suites, is constructed consistently enough that scores are comparable across tasks and comes with an analysis of where its measurements can be trusted. Existing Bengali evaluations cover individual slices (science QA in BEnQA (Shafayat et al., 2024), the task selections of BenLLMEval (Kabir et al., 2023)) and translated collections such as Okapi (Lai et al., 2023) and Indic-Evals (Sarvam AI, 2024) cover several individual benchmarks, but no resource supports a controlled English–Bengali comparison across the full commonsense, science, math and multidomain spectrum. We argue that Bengali is best described not as data-poor but as *evaluation-poor* and this paper is about building and testing the missing instrument.



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 1–17
November 9, 2026 ©2026 Association for Computational Linguistics

The first of the two concepts is *script fragmentation*, and it starts from Bengali's writing system. Bengali is written in an alphasyllabary in which diacritics attach on either side of a base character and consonants combine into conjunct forms, producing multi-character grapheme clusters that constitute the writing system's natural units (Alam et al., 2021). In the word *chandra* (moon), for example, the cluster *ndra* is one written character built from five code points: three consonants joined by two virama signs. Bengali handwriting recognition works at this level, recovering grapheme clusters rather than individual code points (Alam et al., 2021); subword tokenization faces the text-side instance of the same segmentation problem. Byte-pair vocabularies trained predominantly on Latin-script text split Bengali into units smaller than its characters (Rust et al., 2021; Petrov et al., 2023); we measure this with standard tokenizer statistics (§4.3). The cut runs deep: under three of the 4 vocabularies the evaluated models share, the average Bengali token spans 2.2 - 2.6 bytes, less than a single three-byte Bengali code point, while the average English token spans 4.4 - 4.9 bytes. The depth is not fixed by the script: it varies twofold across those vocabularies, so fragmentation is a tokenizer choice. A benchmark suite with paired English and Bengali versions of identical content is precisely the setting in which this cost can be quantified, which is what we do here.

*Format*, the second concept, is a property of the evaluation rather than the language. Benchmark prompts instruct models to answer in a fixed shape (`Correct Answer: [option]` in our setup) and exact-match scoring accepts nothing else: a model that replies "the correct answer is (b)" or reasons before stating the answer is scored wrong despite knowing it. The response error rate (§4.2) measures this failure and §6.1 shows it strikes Bengali and English unequally.

Concretely, we translate 20 widely used English benchmarks into Bengali using a single consistent GPT-4o-mini pipeline, evaluate 8 of them with 10 open instruction-tuned models from the LLaMA, Qwen, Mistral and DeepSeek-R1 families in the range of 3B to 72B parameters, on the paired English and Bengali versions under three metrics: exact-match accuracy, an LLM-judge score and a response-format error rate. The headline gap is real: macro-averaged over models and tasks, English scores exceed Bengali scores by 0.17 - 0.18 on the 0 - 1 score scale under both metrics (LLM-judge: 0.79 vs. 0.63). The paired design also lets us decompose the gap; the decomposition is where the findings lie.

Our contributions are:

- **A uniformly constructed Bengali evaluation suite.** 8 benchmarks covering commonsense, science, math and multidomain knowledge, all translated with one pipeline and released with the pipeline and evaluation code. The suite complements existing Bengali resources: it adds WinoGrande, CommonsenseQA and OpenBookQA, for which we are not aware of public Bengali versions and, unlike prior translated collections assembled from heterogeneous sources, all datasets share one pipeline, prompt format and validation procedure (§3.4).
- **A paired English–Bengali evaluation of 10 open LLMs** across 4 families and three metrics, quantifying how the gap varies with model scale and family (§5).
- **Evidence that exact-match scoring conflates format adherence with capability.** When format-error rates are below 5% in both languages, exact-match and judge-based estimates of the cross-lingual gap agree within about 0.05; when format errors exceed 10%, exact match misstates the gap by 0.09 - 0.24 and can invert its sign. Reasoning-tuned models are penalized hardest. This extends known results on format sensitivity (Sclar et al., 2024; Zheng et al., 2024; Alzahrani et al., 2024) to the cross-lingual setting, where the artifact is asymmetric and therefore biases gap estimates rather than merely adding noise (§6.1).
- **A script-fragmentation characterization.** The script is fixed; the tokenizer decides what its text costs. Bengali fertility averages 6.7–7.4 tokens per word against $\sim$1.3 for English, but that pooled figure hides a twofold spread across the 4 vocabularies the 10 models share (7.9 for Llama 3, 4.0 for Tekken); per-instance sequence lengths inflate by up to 5.5$\times$, and three of the 4 vocabularies produce tokens spanning less than a single Bengali character while Tekken clears that boundary. We report a weak negative association between fragmentation and scores ($r = -0.23$) and are explicit that our evidence does not support a causal claim (§6.2).

## 2 Related Work

**Bengali and Indic evaluation resources.** Dedicated Bengali models include BanglaBERT (Bhattacharjee et al., 2022), BanglaGPT (Salim et al., 2023), TituLLM (Nahin et al., 2025) and TigerLLM (Raihan and Zampieri, 2025). On the evaluation side, BenLLMEval assembled a first LLM evaluation for Bengali (Kabir et al., 2023), BEnQA provides parallel English–Bengali science questions from Bangladeshi curricula (Shafayat et al., 2024) and the IndicLLMSuite provides pretraining and fine-tuning data across 22 Indian languages (Khan et al., 2024). Closest to our artifact, Okapi (Lai et al., 2023) released ChatGPT-translated versions of ARC, HellaSwag and MMLU in 26 languages including Bengali, Sarvam AI's Indic-Evals collection (Sarvam AI, 2024) releases Indic translations, including Bengali, of MMLU, BoolQ, ARC-Challenge and GSM8K and Global-MMLU (Singh et al., 2025) includes a Bengali split with partly human-verified translations. Section 3.4 details how our suite relates to these resources.

**Bootstrapping benchmarks by translation.** Translating English benchmarks is an established first step for languages without native evaluation suites: Turkish LLM development relied on translated benchmarks (Acikgoz et al., 2024) and the Aya project used translation alongside human templating across 100+ languages (Üstün et al., 2024). The approach has known costs, translationese, cultural mismatch and translation errors, that native construction, as in the Khayyam Challenge (Ghahroodi et al., 2024), avoids. We take the translation route with open eyes: it is the only way to obtain *paired* English–Bengali versions of identical items, which our measurement analysis requires (§3.5).

**Tokenization and script fragmentation.** Fertility and related statistics have long been used to compare tokenizers across languages (Rust et al., 2021) and tokenizer disparities translate into real cost and context-length penalties for non-Latin scripts (Petrov et al., 2023; Ahia et al., 2023). Tokenizer choice measurably affects downstream performance (Ali et al., 2024; Dagan et al., 2024) and subword fragmentation degrades robustness (Chai et al., 2024). For Indic scripts specifically, recent work proposes morphologically and orthographically grounded segmentation (Brahma et al., 2025; Vemula et al., 2025; Libovický and Helcl, 2024) and byte-level modeling for Bangla (Bhattacharyya and Bhattacharya, 2025). The visual counterpart of the segmentation problem is central to Bengali handwritten grapheme recognition (Alam et al., 2021). Our contribution is descriptive: paired statistics locating the fragmentation below the character level.

**Evaluation robustness and format sensitivity.** LLM scores are highly sensitive to prompt formatting (Sclar et al., 2024), option ordering and selection biases in MCQ (Zheng et al., 2024) and small benchmark perturbations (Alzahrani et al., 2024); prompt-sensitivity benchmarks quantify this variance systematically (Razavi et al., 2025; Polat et al., 2024). Unreliable answer extraction is a recognized failure mode of evaluation harnesses (Yu et al., 2025) and LLM-as-judge protocols are the standard mitigation, with known biases of their own (Zheng et al., 2023). We build on this literature rather than proposing new metrics: in a cross-lingual setting these known artifacts become *asymmetric* between languages, turning benign noise into systematic bias on gap estimates.

## 3 The Bengali Benchmark Suite

### 3.1 Benchmark Selection

Starting from the suites used to report results for open model families (Grattafiori et al., 2024), we translated 20 English benchmarks into Bengali (inventory in Appendix A) and selected 8 spanning 4 task categories with well-defined answers that support automatic scoring: **commonsense and reading comprehension** (HellaSwag, WinoGrande, CommonsenseQA, OpenBookQA, BoolQ), **science** (ARC, contributing ARC-Easy and ARC-Challenge splits), **math** (GSM8K) and **multidomain knowledge** (MMLU). Because ARC contributes two evaluation splits, result tables report 9 columns for the 8 benchmarks.

### 3.2 Translation Pipeline

All datasets were translated with `gpt-4o-mini-2024-07-18`. The model was prompted as a professional translator, instructed to preserve meaning, tone, idioms, proper nouns and, critically, the ground-truth answer values and to emit valid JSON so that translations could be machine-validated (Table 8). Automated checks validated JSON structure, key completeness and answer-key integrity for every item; entries that failed validation or were dropped by the

batched pipeline were detected by log analysis and retranslated until the datasets were complete. Translating all 20 benchmarks cost approximately $200, evidence that this bootstrap is affordable for other underserved languages.

### 3.3 Translator Selection

Before committing to a translation engine, we compared Google Translate, Azure Translator and `gpt-4o-mini-2024-07-18` in a blind review: reviewers(native Bengali speakers) saw translations of the same source items without knowing which system produced them and indicated which they preferred. GPT-4o-mini was preferred for accuracy and coherence and was used for the full pipeline. We did not log the number of reviewers, use a formal rubric or compute inter-rater agreement, so we treat this as an informal preference test that guided an engineering choice, not as a validation of translation quality (§3.5).

### 3.4 Position Relative to Existing Resources

Bengali versions of some of these benchmarks already exist: Okapi (Lai et al., 2023) includes ChatGPT-translated Bengali ARC, HellaSwag and MMLU, the Indic-Evals collection (Sarvam AI, 2024) provides Indic translations including Bengali of MMLU, BoolQ, ARC-Challenge and GSM8K, Global-MMLU (Singh et al., 2025) covers Bengali MMLU with partial human verification and BEnQA (Shafayat et al., 2024) provides parallel science QA. Our suite is complementary in three specific ways. First, it adds Bengali versions of WinoGrande, CommonsenseQA and OpenBookQA, for which we are not aware of public Bengali translations. Second, every dataset is produced by one pipeline with one prompt, one engine and one validation procedure, so differences between task scores reflect the tasks rather than heterogeneous translation provenance; this consistency is what makes the cross-task and cross-language analyses in §6 interpretable. Third, because every Bengali item is paired with its English source under identical formatting, the suite supports the controlled gap decomposition and paired tokenization statistics that are this paper's main analytical contributions.

### 3.5 Translation Quality

The suite is machine-translated and has not undergone item-level human validation; we state this up front rather than in passing. The automated pipeline guarantees structural integrity (well-formed items, intact answer keys, complete datasets) but not semantic fidelity. We therefore frame the suite as a *first-pass instrument with characterized error modes*: suitable for the relative, paired comparisons performed in this paper, where translation noise applies to all models equally and improvable through the released pipeline. Human validation on a stratified sample is the next step; the Limitations section returns to this point.

## 4 Experimental Setup

### 4.1 Models

We evaluate 10 open instruction-tuned models drawn from 4 families chosen to span three axes: parameter scale (3B - 72B), documented multilingual support and documented Bengali coverage in pretraining (Table 7, Appendix D). The LLaMA models represent widely deployed open models without documented Bengali pretraining, Qwen represents a multilingual family with documented Bengali exposure, Mistral documents no Bengali support (Mistral Small is multilingual but lists no Bengali) and the DeepSeek-R1 distillations, built on Qwen (14B) and Llama (70B) bases, represent reasoning-tuned models, a category that matters for measurement (§6.1). All models are evaluated zero-shot, without fine-tuning, using fixed prompt templates that instruct the model to answer in the format `Correct Answer: [option]`; full templates and inference details are in Appendix D.

### 4.2 Task Metrics

We score every model–dataset–language cell with three complementary metrics.

**Exact-match accuracy** counts a response as correct if it begins with the expected answer string (after normalization such as lowercasing and label mapping). This strictest, most common protocol is deliberately retained because its failure modes are part of what we study.

**Response Error Rate (RER)** is the fraction of responses that do not begin with *any* valid answer option for the item:

$$\mathrm{RER} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\left(\forall\, p \in P_i : \neg\, \mathrm{pre}(r_i, p)\right), \quad (1)$$

where $P_i$ is the set of valid (normalized) answer prefixes for item $i$ and $\mathrm{pre}(r_i, p)$ tests whether response $r_i$ begins with $p$. Its complement is the Response Adherence Rate,

$$\mathrm{RAR} = 1 - \mathrm{RER}. \quad (2)$$

RER isolates *format* failure from *knowledge* failure: a response with high RER cannot score under exact match regardless of whether the model knew the answer. This makes RER the diagnostic for the answer-extraction failures documented in evaluation-harness studies (Yu et al., 2025).

**LLM-judge score** is the fraction of responses that a few-shot GPT-based judge rules semantically equivalent to the ground truth, independent of surface format (Zheng et al., 2023). The judge is our estimate of capability with format effects removed; its own reliability limits are discussed in the Limitations section.

### 4.3 Tokenization Metrics

We characterize each model's tokenizer on the paired corpora using three standard statistics, computed over every prompt (system prompt plus item) in each dataset and language. Although we evaluate 10 models, they share only 4 distinct vocabularies: the Llama 3 tokenizer (the 4 LLaMA models and the DeepSeek-R1 Llama distill), the Qwen 2 tokenizer (both Qwen models and the DeepSeek-R1 Qwen distill) and the two Mistral vocabularies (Mistral 7B's 32k SentencePiece and Mistral Small's Tekken). All tokenizer statistics therefore characterize 4 vocabularies carried by 10 models. **Fertility** is the mean number of tokens per whitespace-separated word (Rust et al., 2021). **Tokens per instance** is the mean tokenized length of a full prompt, the quantity that determines context usage and inference cost (Ahia et al., 2023). **Bytes per token** is the mean UTF-8 byte length of the text divided by its token count (Dagan et al., 2024); higher values mean each token spans more text (coarser, more efficient tokenization), lower values mean finer fragmentation. Earlier drafts used ad hoc names for these quantities; we use the established names for comparability with the tokenization literature.

## 5 Results

### 5.1 The Cross-Lingual Gap

Table 1 summarizes performance macro-averaged over the 9 evaluation splits; Tables 4–6 give per-split scores. Averaged over all models, English exceeds Bengali by 0.180 under exact match and 0.169 under the LLM judge, so a substantial capability gap survives even after format effects are removed. The gap narrows sharply with scale within each family (row marginals of Figure 1): the

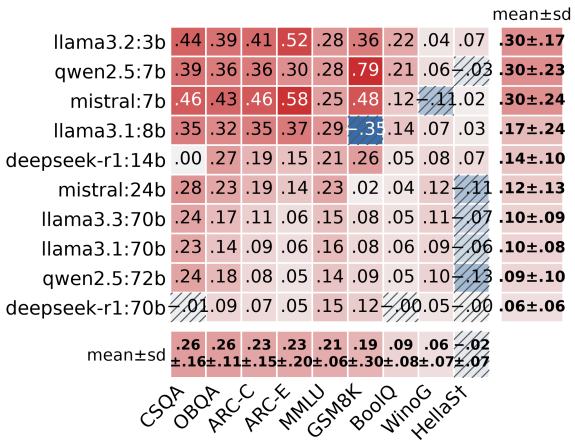



Figure 1: Per-model, per-split English−Bengali LLM-judge gap; rows and columns sorted by mean gap, marginals give mean±sd, hatched cells are negative (Bengali above English) and color encodes the printed value. †The judge is unreliable on HellaSwag (§6.1). The exact-match counterpart is Figure 7 (Appendix C).

LLaMA judge gap falls from 0.304 (3B) to 0.095–0.100 (70B), Qwen from 0.303 to 0.089 and Mistral from 0.299 to 0.125. The best Bengali performer overall is deepseek-r1:70b (judge 0.879), followed by the large Qwen and LLaMA models. Documented Bengali pretraining coverage (Table 7) shows no clear effect at small scale: qwen2.5:7b (documented coverage) and llama3.1:8b (none) appear to differ (judge gaps 0.303 vs. 0.173), but the difference vanishes once GSM8K, a split carrying known harness anomalies (see Limitations), is excluded entirely (0.242 vs. 0.239), so this pool supports no conclusion about coverage either way.

### 5.2 Where the Gap Concentrates

Figure 1 maps each model's judge gap by split and two regimes emerge: knowledge-heavy multiple choice carries the gap (CSQA, OBQA, ARC and MMLU average 0.21–0.26) while binary BoolQ and WinoGrande stay at or below 0.09. The column marginals separate two further stories: MMLU's gap is nearly model-independent (0.21±0.06), suggesting a property of the language pair rather than of any model, whereas GSM8K's similar mean hides extreme model dependence (0.19±0.30), spanning qwen2.5:7b's 0.79 collapse to llama3.1:8b's −0.35. The row marginals show mean and variance falling together with scale: a large gap is also an erratic gap. The exact-match view of the same matrix (Figure 7, Appendix C) is instead dominated by the format artifacts of §6.1.

| | Exact-match accuracy | | | LLM-judge | | | RER | |
|---|---|---|---|---|---|---|---|---|
| **Model** | EN | BN | Δ | EN | BN | Δ | EN | BN |
| llama3.1:8b | 0.682 | 0.401 | 0.281 | 0.669 | 0.496 | 0.173 | 0.020 | **0.190** |
| llama3.1:70b | 0.885 | 0.751 | 0.133 | 0.843 | 0.748 | 0.095 | 0.007 | 0.010 |
| llama3.2:3b | 0.658 | 0.322 | 0.336 | 0.657 | 0.353 | 0.304 | 0.017 | 0.064 |
| llama3.3:70b | 0.851 | 0.752 | 0.099 | 0.845 | 0.745 | 0.100 | 0.048 | 0.015 |
| qwen2.5:7b | 0.816 | 0.468 | 0.348 | 0.793 | 0.490 | 0.303 | 0.010 | 0.004 |
| qwen2.5:72b | 0.895 | 0.599 | 0.295 | 0.856 | 0.767 | 0.089 | 0.009 | **0.247** |
| mistral:7b | 0.170 | 0.112 | 0.058 | 0.627 | 0.328 | 0.299 | **0.733** | **0.721** |
| mistral:24b | 0.833 | 0.637 | 0.196 | 0.806 | 0.681 | 0.125 | 0.025 | 0.066 |
| deepseek-r1:14b | 0.750 | 0.515 | 0.236 | 0.908 | 0.767 | 0.141 | **0.140** | **0.212** |
| deepseek-r1:70b | 0.458 | 0.638 | −0.179 | 0.935 | 0.879 | 0.057 | **0.494** | **0.200** |
| Mean | 0.700 | 0.520 | 0.180 | 0.794 | 0.625 | 0.169 | 0.150 | 0.173 |
| Mean (excl. HellaSwag) | 0.703 | 0.540 | 0.163 | 0.830 | 0.638 | 0.192 | 0.147 | 0.157 |

Table 1: Macro-averages over the 9 evaluation splits, per model and language. Δ = EN − BN. **Bold** marks RER above 0.1; for those five models the two gap estimates (underlined) diverge by 0.09–0.24 and can invert in sign, while for the remaining models the two estimates agree within 0.05. The second mean row excludes HellaSwag, where the judge is unreliable (§6.1). Per-split scores are in Tables 4–6.

## 6 Analysis

### 6.1 Format Adherence Confounds Exact-Match Accuracy

The three metrics in Table 1 were chosen so that their disagreements are informative and they disagree systematically. Exact-match accuracy requires both knowing the answer and emitting it in the instructed format; the judge requires only the former; RER measures failure of the latter alone. Prior work established that format sensitivity adds noise to monolingual benchmarks (Sclar et al., 2024; Zheng et al., 2024; Alzahrani et al., 2024). In our paired setting the effect is worse than noise: RER differs *between languages* for the same model, so exact match does not merely blur the cross-lingual gap, it biases it. Shafayat et al. (2024) observe the same phenomenon qualitatively, noting that eliciting format-conforming output is markedly harder in Bengali than in English and fall back to manual answer assessment; our RER quantifies that failure mode at scale. Three patterns in Tables 1 and 5 make the case; Figure 2 previews them.

**Apparent incapability that is actually format failure.** mistral:7b scores 0.048 exact-match on English OpenBookQA while the judge scores it 0.678; its RER there is 0.914, meaning nine of ten responses never began with a valid option label. Its macro exact-match score (0.170 EN) would place it far below every other model, while its judge score (0.627 EN) sits modestly below the other 7–8B models (0.669–0.793) rather than at the catastrophic floor its exact-match score implies. An earlier draft concluded that Mistral consistently underperforms across languages; the RER column shows that conclusion was substantially a prefix-matching artifact and we correct it here. The genuine Mistral finding is different and appears only under the judge: proportionally, it loses 48% of its English judge score in Bengali (0.627 to 0.328), the largest relative drop among the 7–8B models (38% for qwen2.5:7b, 26% for llama3.1:8b), consistent with its undocumented Bengali support.

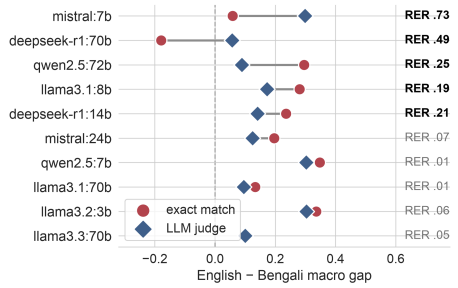



Figure 2: Exact-match vs. LLM-judge estimates of the macro English−Bengali gap per model, sorted by divergence; the right column gives each model's larger macro RER (bold above 0.1). Format failure separates the estimates by up to 0.24 and inverts the sign for deepseek-r1:70b; near-zero-RER models (grey) agree.

**Reasoning-tuned models are penalized hardest.** The DeepSeek-R1 distillations produce chain-of-thought before the final answer, which frequently breaks the required answer prefix. deepseek-r1:70b has an English RER of 0.494 and an English exact-match of 0.458, against a judge score of 0.935, the highest English judge score in the pool. Exact-match protocols thus rank

the strongest reasoner in the pool as one of the weakest models, a concrete instance of the answer-extraction failures that motivate harness-robustness work (Yu et al., 2025).

**Language-asymmetric format failure biases the gap estimate.** Because RER is not equal across languages, the bias does not cancel in the EN−BN difference. For qwen2.5:72b, Bengali RER (0.247) far exceeds English RER (0.009) and the exact-match gap (0.295) is more than triple the judge gap (0.089): most of this model's apparent Bengali deficit is formatting. llama3.1:8b shows the same pattern on OpenBookQA, where accuracy collapses from 0.790 (EN) to 0.172 (BN) while RER rises from 0.000 to 0.658. For deepseek-r1:70b the asymmetry runs the other way (EN RER 0.494 vs. BN 0.200) and exact match reports a *negative* gap of −0.179, Bengali above English, where the judge reports a positive gap of 0.057. Across the pool the pattern is monotone: the three models with RER below 0.05 in both languages show gap estimates agreeing within 0.05, the two models with intermediate RER (peaking near 0.07) diverge by at most 0.07 and the five models with RER above 0.1 in either language diverge by 0.09 to 0.24 (deepseek-r1:14b lowest, mistral:7b highest).

**Robustness to judge errors.** The judge itself is imperfect and its failure is systematic rather than random: on English HellaSwag, seven models score 0.583–0.905 exact match while the judge places them between 0.352 and 0.423; five of those seven have RER at or near zero, so format failure cannot explain the divergence and only the two chain-of-thought DeepSeek-R1 models escape (0.898 and 0.933). Our hypothesis is response-style sensitivity in the judge itself: HellaSwag asks for a continuation; terse label answers are judged not equivalent to the gold continuation while reasoning models restate it in full. The conclusions above do not depend on the split: excluding HellaSwag (second mean row of Table 1), qwen2.5:72b's exact-match gap (0.247) still more than doubles its judge gap (0.116) and the deepseek-r1:70b sign inversion persists (−0.188 vs. +0.064). The aggregate direction is itself unstable: the pooled exact-match gap exceeds the judge gap with HellaSwag included (0.180 vs. 0.169) and falls below it without (0.163 vs. 0.192); the per-split flip is visible by comparing Figures 1 and 7; the lesson is that exact match is unreliable in both directions, with large per-model biases that partially cancel in aggregate. The RER evidence is judge-independent throughout.

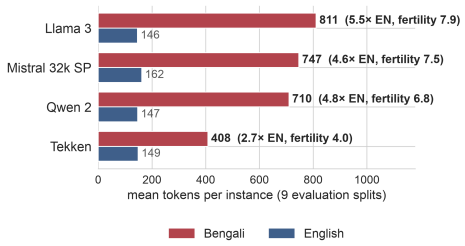



Figure 3: Bengali vs. English tokenization cost by vocabulary: mean tokens per instance over the 9 evaluation splits, annotated with the BN/EN ratio and Bengali fertility. Exact values incl. bytes per token: Table 3 (Appendix B).

The practical conclusion for low-resource evaluation is that a single exact-match number cannot be interpreted as a capability gap: report format-error rates alongside accuracy and treat gap estimates as unreliable wherever RER is non-negligible and asymmetric. This finding is invariant to our translation-quality caveats: translation noise affects both metrics equally; the divergences are metric-internal.

### 6.2 Script Fragmentation: Fixed Terrain, Variable Cost

The paired corpora let us measure exactly what current subword vocabularies do to Bengali text (Figure 4 and additional plots in Appendix B).

**Per-vocabulary cost.** Figure 3 is the concrete result: English cost is nearly identical across the 4 vocabularies (146–162 tokens per instance) while Bengali cost varies twofold. Mistral Small's Tekken vocabulary encodes the same Bengali text in half the tokens of Llama 3 (408 vs. 811 per instance; fertility 4.0 vs. 7.9), from a family with no documented Bengali or Indic support and at a vocabulary size nearly identical to Llama 3's (131k vs. 128k): vocabulary composition, not size or model scale, separates Bengali tokenization cost across open models. The absolute cost is a direct inference multiplier: under the dominant Llama 3 vocabulary the same content costs 5.5 times more context in Bengali (BoolQ: 1,115 tokens vs. 196; HellaSwag: 1,359 vs. 256), shrinking the effective context window for Bengali users independent of any accuracy effect. Pooled across models, Bengali fertility is 6.7–7.4 tokens per word against 1.25–1.4 for English (Figure 6), within documented cross-lingual disparities, which reach 15× for some languages (Petrov et al., 2023; Ahia et al., 2023).

**Sub-character fragmentation.** Every Bengali

code point occupies 3 bytes of UTF-8, so bytes per token tests whether a vocabulary's tokens reach the script's characters. Three of the 4 vocabularies fall below that boundary on Bengali text (2.2–2.6 bytes per token, Table 3, Appendix B): their average token covers *less than one character*. Tekken, at 4.4 bytes, is the one vocabulary that clears the boundary, and it is the same one that halves fertility: a single mechanism, Bengali-capable merges, explains both columns. English tokens span 4.4–4.9 bytes, four to five characters. This sub-character segmentation is the text-side counterpart of the challenge in visual processing of Bengali script (Alam et al., 2021); morphologically grounded Indic tokenization (Brahma et al., 2025; Vemula et al., 2025) and byte-level Bangla modeling (Bhattacharyya and Bhattacharya, 2025) are natural remedies our suite can help evaluate.

**Association with performance.** Across model–dataset cells, fertility and tokens-per-instance both correlate with the LLM-judge score at $r = -0.23$ (Appendix B). We do not read this as a performance effect. The correlation is weak, it is confounded with model family, scale and task, and it is pseudo-replicated: the 10 models draw on only 4 vocabularies, so the fertility values repeat and the effective sample is far smaller than 90 cells. What the data does support are the direct measurements: the fertility premium, the context-cost multiplier and the sub-character fragmentation, none of which depend on any link to accuracy.

## 7 Conclusion

We released a uniformly constructed Bengali suite of 8 benchmarks, evaluated 10 open LLMs on paired English and Bengali versions and used the pairing to answer the title's question. *Format* drives the measured gap, not the real one: language-asymmetric format failure lets exact match distort gap estimates threefold to fivefold or invert their sign, with reasoning-tuned models penalized most; what survives format-insensitive judging, roughly 0.17, is real and narrows sharply with scale. *Script fragmentation* is one phenomenon with a fixed terrain and a variable cost: subword tokenizers cut Bengali's grapheme-cluster script below the character, but how deep the cut runs is set by the vocabulary, not the script. Bengali pays a fivefold average fertility premium and its tokens fall below a single character under three of the 4 vocabularies, yet the premium varies twofold by vocabulary (Tekken halves it) and the association with scores stays weak, so vocabulary composition, not the script itself, sets the tax. The recurring lesson: documentation is not measurement; documented pretraining did not predict smaller gaps in this pool and the best Bengali tokenizer came from a family documenting no Bengali support. For practitioners: report format-error rates, corroborate exact match with a format-insensitive metric and expect reasoning-tuned models to be underrated by prefix matching. The suite and pipeline are a first-pass instrument; human validation, natively authored items and grapheme-aware tokenization come next.

## Limitations

Our datasets are machine translations that have not undergone systematic item-level human validation, and the engine comparison in §3.3 was an informal preference test. Translation errors, translationese and cultural mismatch may affect absolute Bengali scores; our analyses are paired and relative, which limits this exposure; stratified human validation is the most important future work.

We rely on a single LLM judge as our format-insensitive capability metric and do not corroborate it with a second, independent judge. LLM judges have documented biases and may err differently across languages (Zheng et al., 2023); §6.1 documents one split (English HellaSwag) where the judge disagrees with the gold key at zero RER, an exposure bounded by the exclusion check and judge-independent RER evidence; human agreement over judge verdicts and corroboration with an independent judge remain future work. A few cells score far below documented capability (GSM8K: llama3.1:8b in English, qwen2.5:7b in Bengali), suggesting residual harness effects; released outputs permit auditing them.

Our tokenization analysis is aggregate: we report per-vocabulary bytes per token and fertility, but do not measure how often individual grapheme clusters are split or relate cluster-level segmentation to per-item correctness, which we leave to future work. Finally, the model lineup predates the newest releases: scores age; protocols do not.

## Acknowledgements

We thank the Stanford AI Club and Phison Electronics Corporation for collaboration and computational resources.

## References


Emre Can Acikgoz, Mete Erdogan, and Deniz Yuret. 2024. Bridging the bosphorus: Advancing turkish large language models through strategies for low-resource language adaptation and benchmarking. *ArXiv*, abs/2405.04685.

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

Samiul Alam, Tahsin Reasat, Asif Shahriyar Sushmit, Sadi Mohammad Siddique, Fuad Rahman, Mahady Hasan, and Ahmed Imtiaz Humayun. 2021. A large multi-target dataset of common bengali handwritten graphemes. In *International Conference on Document Analysis and Recognition*, pages 383–398. Springer.

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, and 1 others. 2024. Tokenizer choice for LLM training: Negligible or crucial? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3907–3924, Mexico City, Mexico. Association for Computational Linguistics.

Norah Alzahrani, Hisham Abdullah Alyahya, Yazeed Alnumay, Sultan Alrashed, Shaykhah Alsubaie, Yusef Almushaykeh, and 1 others. 2024. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, Bangkok, Thailand. Association for Computational Linguistics.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327. Association for Computational Linguistics.

Pramit Bhattacharyya and Arnab Bhattacharya. 2025. BanglaByT5: Byte-level modelling for Bangla. *Preprint*, arXiv:2505.17102.

Maharaj Brahma, N J Karthika, Atul Singh, Devaraj Adiga, Smruti Bhate, Ganesh Ramakrishnan, Rohit Saluja, and Maunendra Sankar Desarkar. 2025. MorphTok: Morphologically grounded tokenization for Indian languages. *Preprint*, arXiv:2504.10335.

Yekun Chai, Yewei Fang, Qiwei Peng, and Xuhong Li. 2024. Tokenization falling short: On subword robustness in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1582–1599, Miami, Florida, USA. Association for Computational Linguistics.

Gautier Dagan, Gabriel Synnaeve, and Baptiste Rozière. 2024. Getting the most out of your tokenizer for pre-training and domain adaptation. *Preprint*, arXiv:2402.01035.

Omid Ghahroodi, Marzia Nouri, Mohammad Vali Sanian, Alireza Sahebi, Doratossadat Dastgheib, Ehsaneddin Asgari, Mahdieh Soleymani Baghshah, and Mohammad Hossein Rohban. 2024. Khayyam challenge (persianMMLU): Is your LLM truly wise to the persian language? In *First Conference on Language Modeling*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

M. Golam Kabir, Mohammed Saidul Islam, Md Tahmid Rahman Laskar, Mir Tafseer Nayeem, M Saiful Bari, and Enamul Hoque. 2023. Benllm-eval: A comprehensive evaluation into the potentials and pitfalls of large language models on bengali nlp. *ArXiv*, abs/2309.13173.

Mohammed Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh Khapra. 2024. Indicllmsuite: A blueprint for creating pre-training and fine-tuning datasets for indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, page 15831–15879. Association for Computational Linguistics.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning

from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.

Jindřich Libovický and Jindřich Helcl. 2024. Lexically grounded subword segmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7403–7420, Miami, Florida, USA. Association for Computational Linguistics.

Mistral AI Team. 2025. Mistral-Small-24B-Instruct-2501. Hugging Face Model Card, Apache-2.0 licensed. `https://huggingface.co/mistralai/Mistral-Small-24B-Instruct-2501` (accessed 2025-07-20).

Shahriar Kabir Nahin, Rabindra Nath Nandi, Sagor Sarker, Quazi Sarwar Muhtaseem, Md Kowsher, Apu Chandraw Shill, Md Ibrahim, Mehadi Hasan Menon, Tareq Al Muntasir, and Firoj Alam. 2025. Titullms: A family of bangla llms with comprehensive benchmarking. *Preprint*, arXiv:2502.11187.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2023. Language model tokenizers introduce unfairness between languages. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*.

Fina Polat, Ilaria Tiddi, and Paul Groth. 2024. Testing prompt engineering methods for knowledge extraction from text. *Semantic Web*.

Nishat Raihan and Marcos Zampieri. 2025. Tigerllm - a family of bangla large language models. *Preprint*, arXiv:2503.10995.

Amirhossein Razavi, Mina Soltangheis, Negar Arabzadeh, Sara Salamat, Morteza Zihayat, and Ebrahim Bagheri. 2025. Benchmarking prompt sensitivity in large language models. *Preprint*, arXiv:2502.06065.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Md. Shahidul Salim, Hasan Murad, Dola Das, and Faisal Ahmed. 2023. Banglagpt: A generative pretrained transformer-based model for bangla language. *2023 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, pages 56–59.

Sarvam AI. 2024. Indic-evals: Translated versions of popular LLM benchmarks. Hugging Face collection. Includes MMLU, BoolQ, ARC-Challenge, GSM8K, and TriviaQA-MCQ translated into 10 Indian languages including Bengali.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Sheikh Shafayat, H M Quamran Hasan, Minhajur Rahman Chowdhury Mahim, Rifki Afina Putri, James Thorne, and Alice Oh. 2024. BEnQA: A question answering benchmark for bengali and english. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David Ifeoluwa Adelani, Jian Gang Ngui, Daniel Vila-Suero, and 1 others. 2025. Global MMLU: Understanding and addressing cultural and linguistic biases in multilingual evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Saketh Reddy Vemula, Sandipan Dandapat, Dipti Misra Sharma, and Parameswari Krishnamurthy. 2025. Rethinking tokenization for rich morphology: The dominance of unigram over BPE and morphological alignment. *Preprint*, arXiv:2508.08424.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Qingchen Yu, Zifan Zheng, Shichao Song, Zhiyu Li, Feiyu Xiong, Bo Tang, and Ding Chen. 2025. xFinder: Large language models as automated evaluators for reliable evaluation. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, and 1 others. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023), Datasets and Benchmarks Track*.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer,

and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. *Preprint*, arXiv:2402.07827.

## A Translated Benchmark Inventory

Table 2 lists all 20 benchmarks translated by the pipeline. The 8 evaluated in this paper are OpenBookQA, ARC (Easy/Challenge), BoolQ, CommonsenseQA, GSM8K, HellaSwag, WinoGrande and MMLU; the remainder are released for future work.

## B Additional Tokenization Results

The tokens-per-instance columns of Table 3 are pipeline-computed over the full corpora; its fertility and bytes-per-token columns were measured by direct tokenization of Bengali text with each vocabulary's public tokenizer and agree with the pipeline-computed statistics shown here. Figure 5 reports tokens per instance by dataset and language and Figure 6 shows the scatter of fertility and tokens-per-instance against LLM-judge scores underlying the $r = -0.23$ association reported in §6.2.

## C Additional Results Figures

Figure 9 shows family-level LLM-judge score distributions and the standard deviation of scores across languages, complementing the per-model macro-averages in Table 1. Figure 8 plots macro scores by model size and language under both metrics. Tables 4–6 give the per-split scores underlying that table.

## D Reproducibility Details

Figure 10 gives a schematic overview of the construction and evaluation pipeline.

All models were evaluated zero-shot using their chat/instruct variants. Prompts instruct the model to output the answer as `Correct Answer: [option]` (with a Bengali-language equivalent instruction for Bengali inputs) and responses were scored as described in §4.2. Exact-match normalization lowercases responses and maps label variants (e.g., "Yes" to "yes") before prefix comparison. The LLM judge is a few-shot GPT-based classifier that receives the question, ground truth and model response and returns a Correct/Incorrect verdict. Evaluation uses each benchmark's released evaluation split as listed in Table 2 (the validation split where test labels are not public); exact per-cell counts ship with the released datasets. Tokenization statistics were computed over the concatenation of system prompt and item for every instance in each dataset, per model tokenizer. The released repository contains the translation pipeline, all prompt templates in both languages, inference scripts, judge prompts and the scripts that produce every table and figure in this paper. Datasets are released with train/dev/test splits mirroring their English sources.

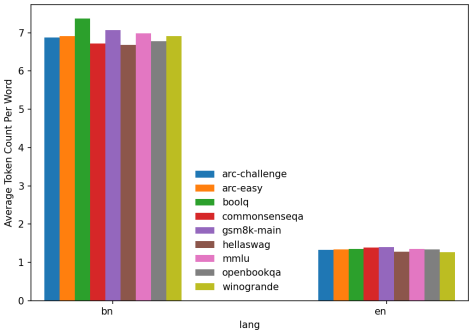
**Artifact availability:** The translated datasets, translation pipeline and evaluation code are publicly released: code on GitHub and datasets in our Hugging Face collection.

| Dataset | Train | Dev | Test | Answer format | Focus |
|---|---|---|---|---|---|
| OpenBookQA | 4957 | 500 | 500 | MCQ | Multi-step commonsense reasoning |
| ARC | 3370 | 869 | 3548 | MCQ | Grade-school science |
| BoolQ | 9427 | 3270 | – | Yes/No | Reading comprehension |
| CommonsenseQA | 9741 | 1221 | 1140 | MCQ | Commonsense reasoning |
| GSM8K | 7473 | – | 1319 | Number | Grade-school math |
| HellaSwag | 39905 | 10042 | 10003 | MCQ | Commonsense continuation |
| WinoGrande | 19482 | 1267 | 1767 | MCQ | Pronoun resolution |
| MMLU | 98487 | 1528 | 13869 | MCQ | Multidomain knowledge |
| BigBenchHard | – | – | var. | MCQ | Logical reasoning |
| AlpacaEval | – | – | 10465 | Instruction | Instruction following |
| Anthropic HH | 86372 | – | 35006 | Free-form | Safety, helpfulness |
| APPS | 5000 | – | 5000 | Code | Coding |
| BFCL | – | – | 250 | Structured | Function calling |
| Dolly | – | – | 7295 | Instruction | Varied NLP tasks |
| HumanEval | – | – | 164 | Code | Code generation |
| MATH | 8599 | – | 4999 | Exact match | Competition math |
| MMLU-Pro | – | 70 | 12032 | MCQ | Multidomain knowledge |
| MR-GSM8K | – | – | 12024 | Exact match | Math meta-reasoning |
| PIQA | 16113 | – | 3084 | MCQ | Physical commonsense |
| SIQA | 33410 | 1954 | – | MCQ | Social reasoning |
| TruthfulQA | – | – | 1634 | MCQ | Truthfulness |

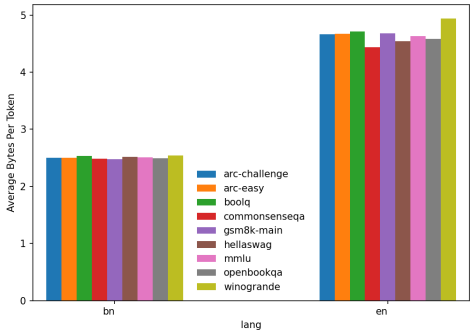
Table 2: All translated benchmarks. The top block lists the 8 evaluated in this paper; the bottom block lists the 12 additional translated benchmarks released for future work.

| Vocabulary | Models | BN | EN | BN/EN | Fert. | B/tok |
|---|---|---|---|---|---|---|
| Llama 3 | 4 LLaMA, R1-70B | 811 | 146 | 5.5 | 7.9 | 2.2 |
| Qwen 2 | 2 Qwen, R1-14B | 710 | 147 | 4.8 | 6.8 | 2.6 |
| Mistral 32k SP | Mistral 7B | 747 | 162 | 4.6 | 7.5 | 2.3 |
| Tekken | Mistral Small | 408 | 149 | 2.7 | 4.0 | 4.4 |

Table 3: Per-vocabulary tokenization of Bengali. Tokens per instance (BN, EN, ratio) are pipeline-computed over the full evaluation corpora; Bengali fertility (tokens per word) and bytes per token are measured by direct tokenization of Bengali text with each public tokenizer and agree with the pipeline statistics (Appendix B). A Bengali code point is 3 bytes: only Tekken's tokens clear the character boundary.




(a) Fertility (tokens per word).




(b) Bytes per token.

Figure 4: Paired tokenizer statistics by dataset and language. Bengali fertility is 6.7–7.4 tokens per word against roughly 1.3 for English (a); in (b), under three of the four vocabularies the average Bengali token spans 2.2 - 2.6 bytes, below the 3-byte Bengali code point (model-weighted mean 2.55; Tekken is the exception at 4.4), against 4.4 - 4.9 bytes for English.

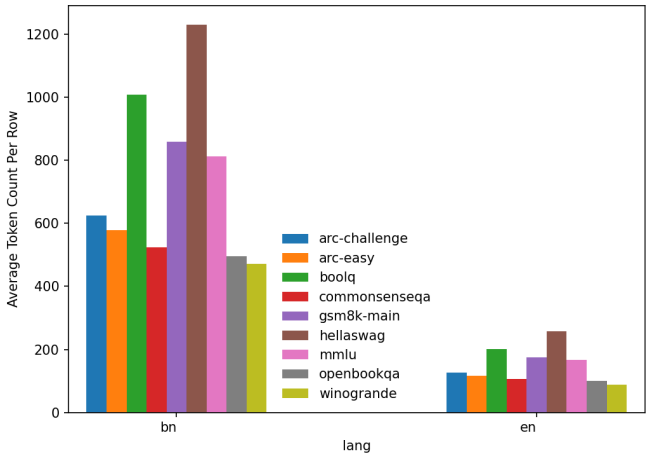
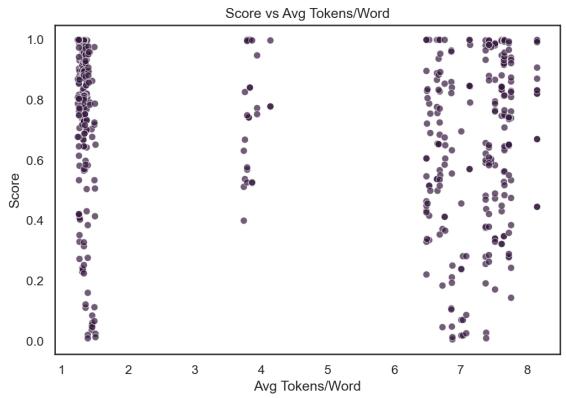
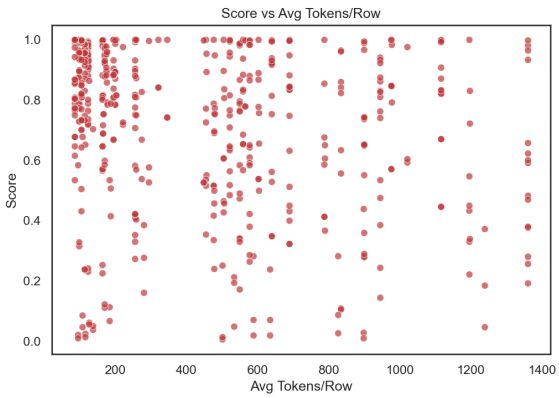



Figure 5: Tokens per instance by dataset and language.




(a) Fertility vs. LLM-judge score.




(b) Tokens per instance vs. LLM-judge score.

Figure 6: Scatter of tokenizer statistics against LLM-judge scores over model–dataset cells. Both associations are weak ($r = -0.23$).

| | Model | OBQA | CSQA | ARC-E | ARC-C | BoolQ | GSM8K | WinoG | HellaS | MMLU |
|---|---|---|---|---|---|---|---|---|---|---|
| **English** | llama3.1:8b | 0.790 | 0.735 | 0.888 | 0.788 | 0.809 | 0.111 | 0.616 | 0.753 | 0.647 |
| | llama3.1:70b | 0.938 | 0.816 | 0.969 | 0.934 | 0.882 | 0.923 | 0.805 | 0.882 | 0.814 |
| | llama3.2:3b | 0.730 | 0.701 | 0.832 | 0.720 | 0.669 | 0.586 | 0.535 | 0.583 | 0.567 |
| | llama3.3:70b | 0.896 | 0.771 | 0.941 | 0.916 | 0.885 | 0.931 | 0.804 | 0.709 | 0.802 |
| | qwen2.5:7b | 0.874 | 0.817 | 0.917 | 0.881 | 0.786 | 0.882 | 0.679 | 0.819 | 0.690 |
| | qwen2.5:72b | 0.960 | 0.849 | 0.969 | 0.943 | 0.893 | 0.909 | 0.809 | 0.905 | 0.817 |
| | mistral:7b | 0.048 | 0.014 | 0.056 | 0.038 | 0.719 | 0.416 | 0.011 | 0.162 | 0.068 |
| | mistral:24b | 0.900 | 0.811 | 0.937 | 0.911 | 0.817 | 0.785 | 0.773 | 0.810 | 0.754 |
| | deepseek-r1:14b | 0.774 | 0.645 | 0.733 | 0.723 | 0.872 | 0.859 | 0.766 | 0.811 | 0.571 |
| | deepseek-r1:70b | 0.316 | 0.432 | 0.238 | 0.231 | 0.839 | 0.923 | 0.647 | 0.273 | 0.226 |
| **Bengali** | llama3.1:8b | 0.172 | 0.423 | 0.529 | 0.433 | 0.671 | 0.387 | 0.519 | 0.193 | 0.282 |
| | llama3.1:70b | 0.790 | 0.650 | 0.922 | 0.846 | 0.822 | 0.811 | 0.648 | 0.624 | 0.650 |
| | llama3.2:3b | 0.330 | 0.287 | 0.349 | 0.323 | 0.446 | 0.145 | 0.485 | 0.257 | 0.280 |
| | llama3.3:70b | 0.764 | 0.643 | 0.918 | 0.835 | 0.833 | 0.827 | 0.635 | 0.659 | 0.652 |
| | qwen2.5:7b | 0.516 | 0.464 | 0.654 | 0.538 | 0.572 | 0.106 | 0.516 | 0.435 | 0.414 |
| | qwen2.5:72b | 0.336 | 0.609 | 0.835 | 0.779 | 0.848 | 0.824 | 0.355 | 0.222 | 0.587 |
| | mistral:7b | 0.006 | 0.048 | 0.019 | 0.019 | 0.594 | 0.011 | 0.240 | 0.046 | 0.026 |
| | mistral:24b | 0.538 | 0.577 | 0.842 | 0.743 | 0.780 | 0.754 | 0.570 | 0.401 | 0.527 |
| | deepseek-r1:14b | 0.500 | 0.457 | 0.568 | 0.500 | 0.792 | 0.557 | 0.552 | 0.339 | 0.367 |
| | deepseek-r1:70b | 0.490 | 0.611 | 0.755 | 0.749 | 0.872 | 0.764 | 0.565 | 0.381 | 0.552 |

Table 4: Exact-match accuracy per evaluation split, English (top block) and Bengali (bottom block).

| | Model | OBQA | CSQA | ARC-E | ARC-C | BoolQ | GSM8K | WinoG | HellaS | MMLU |
|---|---|---|---|---|---|---|---|---|---|---|
| **English** | llama3.1:8b | 0.000 | 0.001 | 0.040 | 0.015 | 0.000 | 0.121 | 0.000 | 0.000 | 0.000 |
| | llama3.1:70b | 0.010 | 0.008 | 0.013 | 0.005 | 0.000 | 0.022 | 0.002 | 0.000 | 0.001 |
| | llama3.2:3b | 0.000 | 0.003 | 0.041 | 0.019 | 0.000 | 0.073 | 0.000 | 0.000 | 0.018 |
| | llama3.3:70b | 0.042 | 0.065 | 0.043 | 0.026 | 0.000 | 0.022 | 0.000 | 0.223 | 0.009 |
| | qwen2.5:7b | 0.008 | 0.000 | 0.041 | 0.018 | 0.000 | 0.020 | 0.000 | 0.000 | 0.001 |
| | qwen2.5:72b | 0.008 | 0.000 | 0.019 | 0.009 | 0.001 | 0.045 | 0.000 | 0.000 | 0.000 |
| | mistral:7b | 0.914 | 0.975 | 0.938 | 0.950 | 0.023 | 0.211 | 0.979 | 0.722 | 0.886 |
| | mistral:24b | 0.000 | 0.002 | 0.039 | 0.020 | 0.000 | 0.039 | 0.000 | 0.125 | 0.000 |
| | deepseek-r1:14b | 0.190 | 0.192 | 0.250 | 0.235 | 0.011 | 0.039 | 0.056 | 0.007 | 0.282 |
| | deepseek-r1:70b | 0.672 | 0.494 | 0.761 | 0.757 | 0.067 | 0.028 | 0.249 | 0.669 | 0.747 |
| **Bengali** | llama3.1:8b | 0.658 | 0.016 | 0.061 | 0.051 | 0.000 | 0.258 | 0.005 | 0.407 | 0.257 |
| | llama3.1:70b | 0.012 | 0.003 | 0.002 | 0.001 | 0.002 | 0.065 | 0.000 | 0.001 | 0.005 |
| | llama3.2:3b | 0.018 | 0.002 | 0.001 | 0.000 | 0.000 | 0.465 | 0.041 | 0.018 | 0.032 |
| | llama3.3:70b | 0.010 | 0.016 | 0.004 | 0.003 | 0.000 | 0.058 | 0.001 | 0.035 | 0.009 |
| | qwen2.5:7b | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.039 | 0.000 | 0.000 | 0.000 |
| | qwen2.5:72b | 0.582 | 0.102 | 0.116 | 0.110 | 0.000 | 0.033 | 0.463 | 0.670 | 0.144 |
| | mistral:7b | 0.986 | 0.788 | 0.930 | 0.929 | 0.024 | 0.561 | 0.542 | 0.815 | 0.913 |
| | mistral:24b | 0.172 | 0.003 | 0.000 | 0.000 | 0.000 | 0.051 | 0.000 | 0.367 | 0.000 |
| | deepseek-r1:14b | 0.244 | 0.162 | 0.314 | 0.344 | 0.015 | 0.158 | 0.046 | 0.277 | 0.348 |
| | deepseek-r1:70b | 0.416 | 0.118 | 0.186 | 0.156 | 0.007 | 0.077 | 0.184 | 0.399 | 0.256 |

Table 5: Response Error Rate (RER, lower is better) per evaluation split, English (top block) and Bengali (bottom block). Note the strong model and language asymmetries: mistral:7b fails format in both languages, deepseek-r1:70b mainly in English, llama3.1:8b and qwen2.5:72b mainly in Bengali.

| | Model | OBQA | CSQA | ARC-E | ARC-C | BoolQ | GSM8K | WinoG | HellaS | MMLU |
|---|---|---|---|---|---|---|---|---|---|---|
| **English** | llama3.1:8b | 0.790 | 0.735 | 0.928 | 0.801 | 0.809 | 0.122 | 0.777 | 0.409 | 0.648 |
| | llama3.1:70b | 0.940 | 0.820 | 0.982 | 0.939 | 0.882 | 0.945 | 0.856 | 0.412 | 0.814 |
| | llama3.2:3b | 0.730 | 0.703 | 0.872 | 0.738 | 0.669 | 0.600 | 0.680 | 0.352 | 0.572 |
| | llama3.3:70b | 0.934 | 0.822 | 0.983 | 0.941 | 0.885 | 0.953 | 0.860 | 0.417 | 0.807 |
| | qwen2.5:7b | 0.876 | 0.817 | 0.957 | 0.899 | 0.786 | 0.896 | 0.789 | 0.423 | 0.690 |
| | qwen2.5:72b | 0.966 | 0.849 | 0.988 | 0.951 | 0.893 | 0.952 | 0.870 | 0.421 | 0.817 |
| | mistral:7b | 0.678 | 0.654 | 0.865 | 0.704 | 0.727 | 0.508 | 0.585 | 0.387 | 0.535 |
| | mistral:24b | 0.900 | 0.811 | 0.977 | 0.930 | 0.817 | 0.792 | 0.869 | 0.404 | 0.754 |
| | deepseek-r1:14b | 0.958 | 0.810 | 0.980 | 0.948 | 0.893 | 0.895 | 0.976 | 0.898 | 0.816 |
| | deepseek-r1:70b | 0.954 | 0.857 | 0.987 | 0.957 | 0.906 | 0.951 | 0.991 | 0.933 | 0.883 |
| **Bengali** | llama3.1:8b | 0.474 | 0.381 | 0.562 | 0.451 | 0.671 | 0.477 | 0.710 | 0.377 | 0.360 |
| | llama3.1:70b | 0.796 | 0.593 | 0.923 | 0.846 | 0.823 | 0.862 | 0.768 | 0.470 | 0.652 |
| | llama3.2:3b | 0.340 | 0.265 | 0.350 | 0.323 | 0.446 | 0.243 | 0.642 | 0.282 | 0.290 |
| | llama3.3:70b | 0.768 | 0.585 | 0.918 | 0.835 | 0.833 | 0.876 | 0.750 | 0.484 | 0.653 |
| | qwen2.5:7b | 0.516 | 0.429 | 0.654 | 0.538 | 0.572 | 0.110 | 0.726 | 0.451 | 0.414 |
| | qwen2.5:72b | 0.790 | 0.606 | 0.938 | 0.868 | 0.848 | 0.861 | 0.772 | 0.547 | 0.677 |
| | mistral:7b | 0.252 | 0.195 | 0.283 | 0.239 | 0.605 | 0.030 | 0.697 | 0.372 | 0.282 |
| | mistral:24b | 0.670 | 0.528 | 0.842 | 0.743 | 0.780 | 0.775 | 0.751 | 0.514 | 0.527 |
| | deepseek-r1:14b | 0.692 | 0.808 | 0.830 | 0.758 | 0.847 | 0.634 | 0.895 | 0.832 | 0.606 |
| | deepseek-r1:70b | 0.868 | 0.866 | 0.933 | 0.883 | 0.909 | 0.831 | 0.946 | 0.934 | 0.737 |

Table 6: LLM-judge score per evaluation split, English (top block) and Bengali (bottom block).

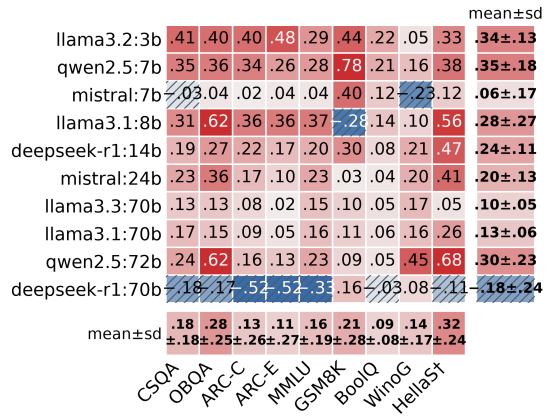



Figure 7: Exact-match counterpart of Figure 1 with identical row and column order. The format artifacts of §6.1 dominate: HellaSwag flips from −0.02 (judge) to +0.32 mean, deepseek-r1:70b's row turns negative because format failure suppresses its English scores and ARC-C/ARC-E swing from consistent judge gaps to standard deviations of 0.26–0.27.

| Model | Size | Multiling. | Bengali pretraining |
|---|---|---|---|
| LLaMA 3.1 (Grattafiori et al., 2024) | 8B | Limited | ✗ |
| LLaMA 3.1 (Grattafiori et al., 2024) | 70B | Limited | ✗ |
| LLaMA 3.2 (Grattafiori et al., 2024) | 3B | Limited | ✗ |
| LLaMA 3.3 (Grattafiori et al., 2024) | 70B | Limited | ✗ |
| Qwen 2.5 (Yang et al., 2025) | 7B | Yes | ✓ |
| Qwen 2.5 (Yang et al., 2025) | 72B | Yes | ✓ |
| Mistral (Jiang et al., 2023) | 7B | No | ✗ |
| Mistral Small (Mistral AI Team, 2025) | 24B | Yes | ✗ |
| DeepSeek-R1 Distill–Qwen (Guo et al., 2025) | 14B | Yes | ✓ |
| DeepSeek-R1 Distill–Llama (Guo et al., 2025) | 70B | Limited | ✗ |

Table 7: Evaluated models (chat/instruct variants). Bengali coverage follows each model's documentation. The DeepSeek-R1 models are reasoning distillations onto Qwen 2.5 (14B) and Llama 3.3 (70B) bases and inherit those bases' language coverage.

| **Role** | Content |
|---|---|
| **System** | You are a professional translator tasked with accurately translating text from English to Bengali. Your primary goal is to provide precise and culturally appropriate translations, regardless of the content's nature. |
| **User** | Translate the following English text into Bengali and ensure the output is valid JSON with all strings enclosed in double quotes: `<english_text> {{"input": {input}, "target": {target}}} </english_text>` Guidelines: 1. Translate accurately, maintaining meaning, tone and context. 2. Handle idiomatic expressions appropriately. 3. Preserve specialized terminology and proper nouns. 4. Translate sensitive content accurately without censorship. 5. Do not translate JSON keys, only values. 6. Ensure valid JSON output with double-quoted strings. Output within `<bangla_translation>` tags; notes in `<translator_notes>` tags. |

Table 8: Prompt template for English-to-Bengali translation.

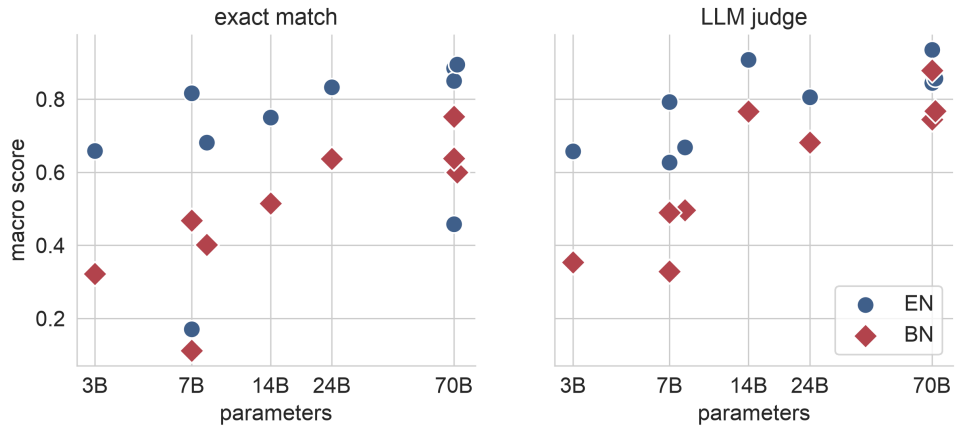
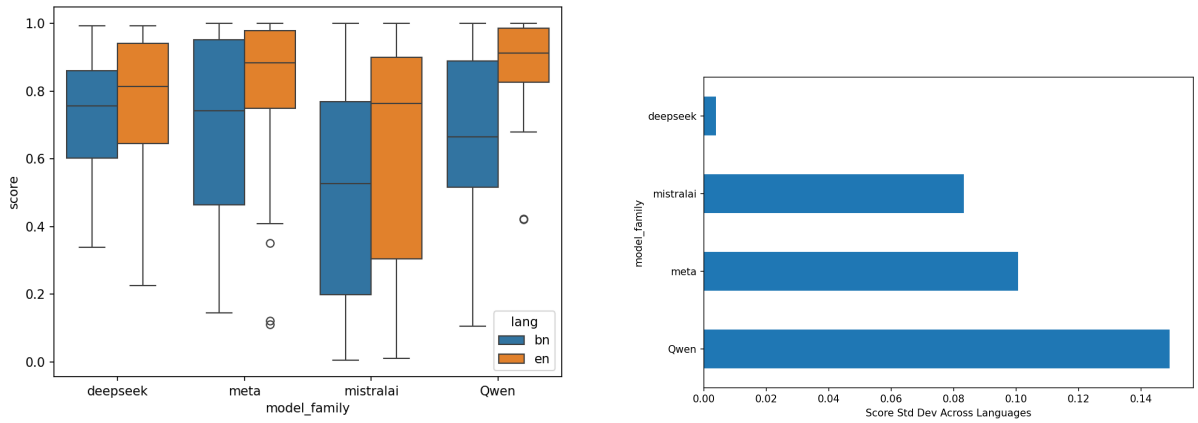



Figure 8: Macro scores by model size and language under exact match (left) and the LLM judge (right). Bengali scores rise faster with scale than English scores; the exact-match panel additionally shows the format-failure outliers of §6.1.




(a) LLM-judge score distributions by model family.

(b) Standard deviation of scores across languages by family.

Figure 9: Family-level score distributions and cross-language robustness. Lower deviation in (b) indicates more stable behavior across English and Bengali; the DeepSeek-R1 family is the most stable, consistent with Table 1.

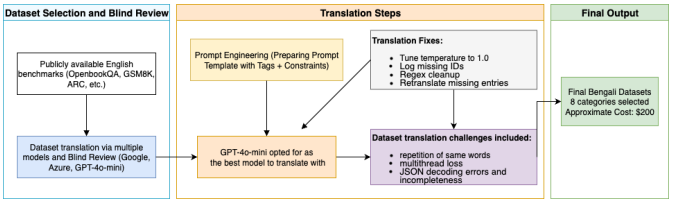



Figure 10: Overview of the construction and evaluation pipeline: benchmark selection, a blind three-way translator comparison, GPT-4o-mini translation with JSON-constrained outputs, automated validation with retranslation and paired English–Bengali evaluation of 10 open LLMs under three metrics.

# A Foundational Study of Multimodal Speech–Text Alignment in Malayalam

**Meharuniza Nazeem*** **Navaneeth S** **Adithyakiran A S** **Rajeev R R** **Ashbi Biju**
International Centre for Free and Open Source Solutions (ICFOSS)
Kerala, India

## Abstract

Speech–text retrieval is a multimodal alignment task in which spoken audio and written transcripts are mapped into a shared representation space. Standard retrieval metrics test whether an audio embedding recovers its transcript from a large candidate pool, but not whether the shared representation preserves small grammatical contrasts inside agglutinated words. We construct an automatic Malayalam benchmark with 12,611 controlled audio–text comparison rows derived from 6,611 held-out audio queries. The benchmark contains three morphosyntactic categories—case, tense, and negation—and a lexical numeral-control category. Alternatives are generated with the `mlmorph` finite-state analyser and generator and retained after a generate–reanalyse consistency check. A CLIP-style model without explicit hard negatives obtains 37.45% full-pool Recall@1 but only 53.66% pairwise accuracy, close to the 50% pairwise baseline. We diagnose a fixed-first-candidate loader defect that under-delivers later-generated categories and replace it with per-transcript category-aware, access-time resampling. At the matched epoch count, pairwise accuracy rises from 73.50% to 79.76%, and tense accuracy rises from 52.83% to 77.70%. The complete corrected procedure reaches 89.69% pairwise accuracy after extended training, while full-pool retrieval remains below the original checkpoint. A stratified human review of 797 rows retains or corrects 731 (91.7%); the final checkpoint obtains 74.01% on this deliberately reweighted and partly rewritten subset. These results show that global retrieval, controlled grammatical discrimination, candidate-delivery logic, and human benchmark validation are distinct components of multimodal evaluation.

*Corresponding author: `meharuniza@icfoss.org`

## 1 Introduction

Speech–text retrieval is a multimodal learning problem connecting spoken audio with written language. A speech encoder and a text encoder map the two modalities into a shared embedding space, enabling spoken-archive search, recording–caption alignment, and audio–language transfer. Standard evaluation asks whether the correct transcript appears among the highest-scoring candidates. This measures global retrieval, but it can conceal a language-specific failure: a model may identify the topic and most lexical content while ignoring a short segment that changes the utterance's grammatical interpretation.

Malayalam is a morphologically rich Dravidian language in which case, tense, polarity, number, and auxiliary structure can be expressed within an orthographic word. For example, പോയി *poyi* 'went' and പോയില്ല *poyilla* 'did not go' share most of their surface and acoustic material, yet differ in polarity. Similarly, സ്കൂളിൽ *skuulil* 'at school' and സ്കൂളിലേക്ക് *skuulilekkŭ* 'to school' differ in case-related material. A model that scores both alternatives almost equally may still achieve high Recall@K if all other candidates are unrelated.

The distinction is expressed differently in the two modalities. In written text, case, tense, and negation are represented by explicit orthographic material, whereas in speech the corresponding distinction is realized acoustically rather than as a symbolic morphological label. The controlled evaluation therefore asks whether acoustic evidence from the fixed utterance is aligned with the appropriate written morphological form in the shared representation space.

In this multimodal setting, dual encoders and contrastive objectives commonly map audio and text into a shared space. This paradigm builds on contrastive predictive coding and CLIP-style symmetric training (van den Oord et al., 2018; Radford



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 18–27
November 9, 2026 ©2026 Association for Computational Linguistics

et al., 2021), and has been adapted to speech–text retrieval, speech translation, and audio–language representation learning (Khurana et al., 2022; Shih et al., 2023; Ye et al., 2022; Elizalde et al., 2023). Pretrained multilingual speech and text models make such training feasible for Indian languages (Babu et al., 2022; Khanuja et al., 2021). However, ordinary in-batch negatives tend to be far from the correct transcript. They teach coarse discrimination without necessarily requiring the model to resolve a suffix-sized contrast.

This paper examines evaluation, data delivery, and benchmark validity. We first ask whether full-pool Recall@K is a reliable proxy for fine-grained sensitivity to Malayalam grammatical contrasts. We then inspect the path through which generated hard negatives reach the loss. Although the source file contained multiple categories, the original loader attached only the first generated candidate to each transcript. Because candidate generation scans from left to right and some triggers occur more frequently, this positional rule systematically altered the effective training distribution. Finally, because finite-state round-trip consistency does not guarantee sentence-level naturalness, we perform a stratified human review and re-evaluate all supplied checkpoints on the accepted or corrected subset.

We position this work as a foundational study in the sense that it establishes an initial evaluation framework and empirical reference point for Malayalam multimodal speech–text alignment, rather than claiming a comprehensive treatment of architectures or linguistic phenomena.

Our contributions are:

1. We construct a 12,611-row automatic multimodal diagnostic over 6,611 held-out audio queries, covering case, tense, negation, and a lexical numeral control.
2. We demonstrate a divergence between full-pool retrieval and controlled discrimination: the no-hard-negative model reaches 37.45% full-pool R@1 but only 53.66% pairwise accuracy.
3. We diagnose a fixed-first-candidate sampling defect and replace it with per-transcript category-aware retention and access-time resampling, without changing the loss architecture.
4. We separate matched-epoch evidence for the delivery correction from the extended-training result and report the unresolved gap between local discrimination and global retrieval.
5. We human-review a stratified 797-row sample, retain or correct 731 rows, and report checkpoint performance on this independently checked subset.

## 2 Related Work

### 2.1 Multimodal speech–text representation learning

Contrastive predictive coding learns representations by distinguishing observed future structure from sampled alternatives (van den Oord et al., 2018). CLIP scales a symmetric dual-encoder objective to image-text pairs and establishes the shared-embedding template used by many later multimodal systems (Radford et al., 2021). Wav2CLIP and SpeechCLIP transfer this idea to audio or speech by connecting pretrained audio features with language-aligned spaces (Wu et al., 2022; Shih et al., 2023). CLAP learns a general audio-text space from natural-language supervision (Elizalde et al., 2023). For multilingual speech, SAMU-XLSR aligns utterance-level speech representations with sentence embeddings (Khurana et al., 2022), while ConST adds cross-modal contrastive learning to speech translation (Ye et al., 2022). ATRI focuses on inconsistencies caused by multilingual audio-text data distributions (Yin et al., 2025). Our study differs by targeting controlled, within-language grammatical contrasts rather than broad semantic compatibility.

The backbones used here reflect two relevant lines of pretrained modelling. Wav2vec 2.0 and XLS-R learn contextual speech representations from unlabeled audio, with XLS-R extending the approach across languages at scale (Baevski et al., 2020; Babu et al., 2022). MuRIL provides text representations tailored to Indian languages and transliterated text (Khanuja et al., 2021). Benchmarks and datasets such as Common Voice, FLEURS, and IndicSUPERB illustrate the growing infrastructure for multilingual and Indian-language speech evaluation (Ardila et al., 2020; Conneau et al., 2023; Javed et al., 2023); our contribution is a targeted diagnostic rather than a new general speech corpus.

### 2.2 Negative sampling and dense retrieval

Dual encoders depend on the negative distribution. Dense Passage Retrieval shows the effectiveness of in-batch and retrieved negatives for scalable ranking (Karpukhin et al., 2020); RocketQA

further develops denoised and cross-batch negative strategies (Qu et al., 2021). Theoretical and empirical work shows that hard-negative choice changes the bias and optimization behaviour of contrastive objectives (Zhang and Stratos, 2021; Robinson et al., 2021). In our setting, the challenge precedes the scoring function: generated categories existed in storage, but the loader's positional choice prevented many from participating. This makes the sampler an experimentally consequential part of the model specification.

### 2.3 Targeted and minimal-pair evaluation

Aggregate test performance can hide narrow decision-boundary failures. Contrast sets perturb examples in controlled ways to expose such failures (Gardner et al., 2020); CheckList organizes behavioural tests around capabilities and perturbation templates (Ribeiro et al., 2020); Dynabench emphasizes iterative, failure-driven benchmarking (Kiela et al., 2021). BLiMP demonstrates how minimal pairs can isolate particular linguistic phenomena (Warstadt et al., 2020). Our benchmark imports this diagnostic logic into speech-text retrieval: the audio is held fixed, and the model must prefer the attested transcript over one controlled text alternative.

### 2.4 Morphological resources and finite-state control

Finite-state morphology provides reversible analysis and generation for morphologically complex languages (Beesley and Karttunen, 2003). For Malayalam, `mlmorph` implements a reversible finite-state analyser and generator covering inflectional, derivational, and phonological processes (Thottingal, 2019). Its reversibility supports the mutation protocol used here: analyse a word, alter a designated feature, generate a new surface form, and reanalyse it. This round trip checks consistency with the finite-state system; it is not equivalent to human validation of sentence naturalness.

## 3 Multimodal Model and Implementation

### 3.1 Cross-modal retrieval and controlled discrimination

Let $a_i = f_a(x_i)$ be the normalized embedding of Malayalam waveform $x_i$, and let $t_j = f_t(y_j)$ be the normalized embedding of transcript $y_j$. Global retrieval ranks all candidate transcripts by cosine similarity $a_i^\top t_j$. We report Recall@1, Recall@5, and Recall@10 over the 7,953-transcript held-out pool. The targeted benchmark instead supplies a controlled alternative $y_i^-$ for each true transcript $y_i^+$ and records whether

$$a_i^\top f_t(y_i^+) > a_i^\top f_t(y_i^-). \tag{1}$$

Pairwise accuracy is the proportion of rows satisfying this inequality; its random baseline is 50%. The full-pool and pairwise tasks use the same audio embeddings but test different properties of the shared multimodal space. The pairwise probe is inherently cross-modal: the audio is held fixed while the two text candidates differ only in a controlled linguistic feature. Correct discrimination therefore requires the audio embedding to preserve acoustic evidence sufficient to prefer the written form that corresponds to what was actually spoken.

### 3.2 Architecture and optimization

The model is a CLIP-style dual encoder. A `wav2vec2-xls-r-300m` branch consumes 16 kHz raw waveforms, and a shared `muril-base-cased` text branch encodes both true and controlled-alternative transcripts. Attention-masked mean pooling is used for text and temporal mean pooling for audio. Linear projection heads map both branches into a shared 256-dimensional space, followed by $\ell_2$ normalization. No explicit spectrogram branch is used: the audio modality enters as waveform samples.

Training minimizes symmetric audio-to-text and text-to-audio InfoNCE, with a learnable temperature. When a row has an attached hard negative $t_i^-$, a hinge term is added:

$$\mathcal{L}_{\text{HN}} = \max\Big(0, 0.2 - (a_i^\top t_i^+ - a_i^\top t_i^-)\Big). \tag{2}$$

Table 1 records the supplied implementation settings. The temperature is clamped after each update to prevent bf16 overflow. Gradient clipping was added when the corrected run was extended beyond epoch 2; it is therefore part of the final procedure, not of the matched-epoch sampler comparison.

## 4 Data and Agglutination-Aware Benchmark

### 4.1 Corpus and held-out pool

The processing pipeline begins with 89,200 Malayalam audio–transcript pairs aggregated from multiple publicly available speech datasets identified

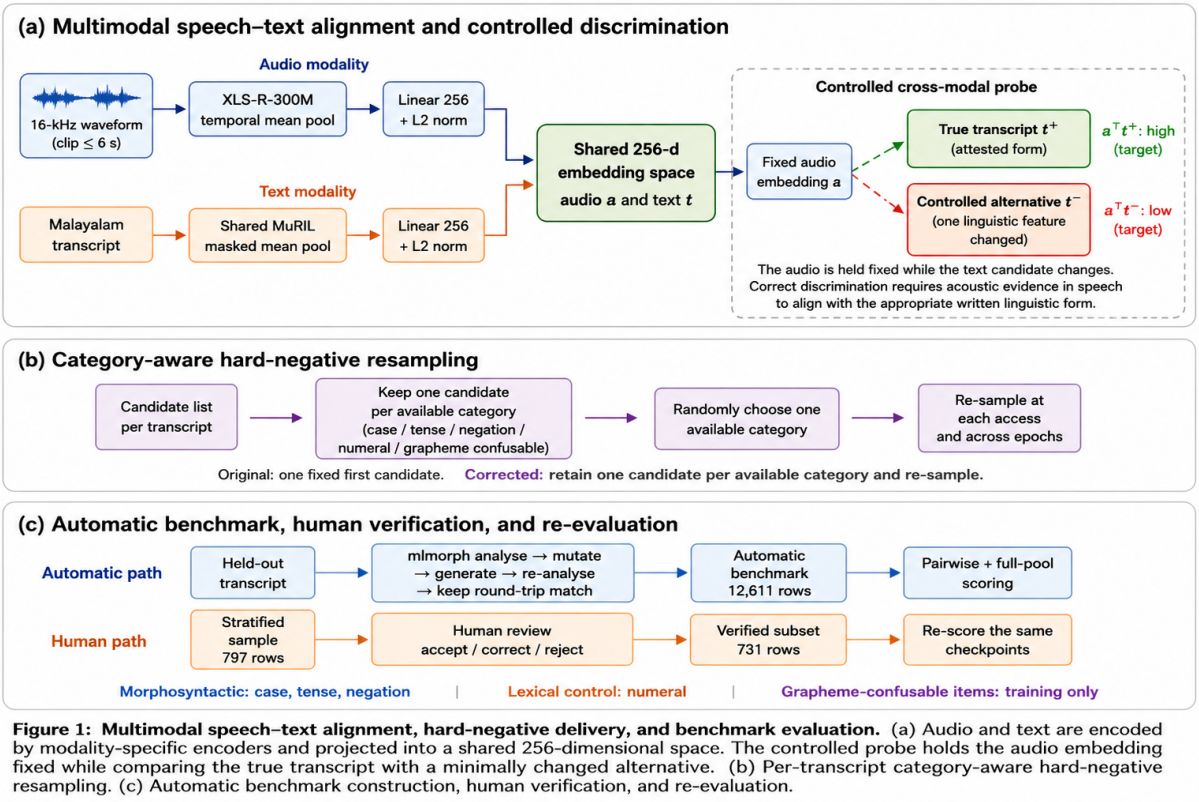



**Figure 1: Multimodal speech–text alignment, hard-negative delivery, and benchmark evaluation.** (a) Audio and text are encoded by modality-specific encoders and projected into a shared 256-dimensional space. The controlled probe holds the audio embedding fixed while comparing the true transcript with a minimally changed alternative. (b) Per-transcript category-aware hard-negative resampling. (c) Automatic benchmark construction, human verification, and re-evaluation.

Table 1: Implementation settings used in the reported experiments.

| Component | Setting |
|---|---|
| Audio encoder | `wav2vec2-xls-r-300m`; temporal mean pool |
| Text encoder | `muril-base-cased`; masked mean pool; shared for positive and negative text |
| Shared space | Linear projection to 256 dimensions; $\ell_2$ normalization |
| Loss | Symmetric InfoNCE + hard-negative hinge, margin 0.2 |
| Optimization | AdamW, learning rate $10^{-4}$, batch 32, bf16 |
| Audio input | 16 kHz waveform, maximum 6 seconds |
| Stabilization | Temperature clamped to $[0, \ln 100]$; gradient norm clipped at 1.0 in the extended corrected run |
| Memory | Gradient checkpointing on both encoders |

during corpus construction. Because the corpus combines data from several sources, the original provenance and source-specific licensing conditions are retained for each constituent dataset. Deduplication and quality control yield 55,059 unique transcript records used for linguistic annotation and hard-negative generation. The supplied split contains a 5,225-record anchor subset and uses speaker-disjoint partitioning; all reported benchmark evaluations use a held-out pool of 7,953 audio–transcript pairs.

The linguistic-processing record contains 760,630 generated alternatives before benchmark filtering. These inventory counts describe stored candidates; they do not imply that every category reached the loss under the original loader, which is the defect studied in Section 5.

### 4.2 Controlled candidate generation

For each content word, `mlmorph` analyses the surface form, applies an eligible category-specific mutation, regenerates a replacement, and reanalyses it. A candidate is retained when the generated analysis matches the intended mutation. Case candidates cycle among configured case analyses; tense candidates flip configured past/present analyses; negation is added or removed where licensed by the analyser; and numeral candidates use a fixed replacement lexicon.

A fifth training category, called `phon_conf` in the implementation, substitutes configured grapheme-confusable sets such as ല/ള/ഴ and ന/ണ. Because this operation is orthographic and is not independently pronunciation-validated, we refer to it as a *grapheme-confusable* category. It con-

Table 2: Automatic benchmark composition and stratified human verification. "Retained" includes rows accepted unchanged or after reviewer correction.

| Category | Auto. rows | Reviewed | Retained | Rate |
|---|---|---|---|---|
| Case | 4,890 | 150 | 148 | 98.7 |
| Tense | 1,870 | 150 | 142 | 94.7 |
| Negation | 5,504 | 150 | 138 | 92.0 |
| Numeral control | 347 | 347 | 303 | 87.3 |
| Total | 12,611 | 797 | 731 | 91.7 |
| Covered audio queries: 6,611 / 7,953 (83%) | | | | |

tributes to training but is excluded from the reported morphosyntactic benchmark. Round-trip validation establishes consistency with the finite-state analyser and generator; it does not by itself establish sentence naturalness or contextual appropriateness.

### 4.3 Automatic benchmark composition and metrics

Emitting one row per available query–category pair yields 12,611 automatic rows. A query can appear in more than one row, but each row contains one true transcript and one controlled alternative. Table 2 combines the automatic benchmark composition with the stratified human-verification outcome. Case, tense, and negation are morphosyntactic contrasts; numeral replacement is a lexical control. The automatic imbalance reflects trigger availability in held-out text rather than a balanced test design.

Pairwise accuracy is the primary diagnostic: for each row, the model must score the attested transcript above its controlled alternative. The random baseline is 50%. Full-pool Recall@K is computed for the same 6,611 covered audio queries against all 7,953 held-out transcripts. These metrics answer different questions and have different chance levels; their raw percentages should not be compared as if they shared a common scale.

### 4.4 Human-verification protocol

Finite-state consistency does not guarantee that a generated sentence is natural or that only the intended category changed. We therefore drew a stratified sample of 797 automatic rows: 150 each for case, tense, and negation, together with all 347 numeral rows. Four category-dedicated Malayalam reviewers assessed their assigned rows, and a lead reviewer independently checked the complete sample. Each pair was judged for sentence naturalness, agreement with the labelled change, and final disposition: accept unchanged, accept with a reviewer-written correction, or reject. The lead review supplied a second judgement for every item and the final accept/reject decision; reviewer correction text was retained where needed. As summarized in Table 2, of the 797 reviewed rows, 353 were accepted unchanged, 378 were retained after reviewer correction, and 66 were rejected; thus 731/797 rows (91.7%) were retained overall.

## 5 Per-Transcript Category-Aware Hard-Negative Resampling

### 5.1 Fixed-first-candidate defect

The hard-negative file stores multiple candidates in generation order, but the original loader attached only `negatives[0]` and reused that text for every epoch. Since generation scans words left to right and `phon_conf`/case triggers occur frequently, they commonly occupied position zero. Tense requires an eligible explicitly tensed verb and was under-delivered by the fixed-first selection rule. The held-out benchmark nevertheless contains 1,870 tense contrasts, allowing this failure mode to be evaluated. The defect changes the effective supervision distribution without changing the model or loss.

### 5.2 Corrected candidate-delivery procedure

We use a simple category-aware resampling correction to ensure that available contrast categories are not deterministically excluded by candidate order. We do not claim this procedure as a novel sampling algorithm; its role is to isolate the consequences of an experimentally consequential data-delivery defect. To correct the fixed-first-candidate defect, we reorganize the generated alternatives by transcript and category. For each transcript, at most one candidate is retained from every available category. Rather than attaching one fixed alternative when the dataset is loaded, the training dataset randomly selects one of the available category-specific candidates whenever the item is accessed. Consequently, a transcript with multiple available contrast types can expose different hard negatives across training epochs. The procedure does not globally equalize category frequencies because each transcript exposes only the categories licensed by its linguistic content; we therefore refer to it as per-transcript category-aware resampling rather than global category balancing.

The original and corrected epoch-2 checkpoints

provide the closest available comparison of this delivery change. The corrected epoch-6 checkpoint additionally includes four further epochs and gradient clipping, and is therefore treated as the complete corrected training procedure rather than a sampler-only ablation.

## 6 Experimental Setup

### 6.1 Systems and comparison logic

We evaluate four supplied checkpoints: (i) **Stage 1**, symmetric contrastive training without attached hard negatives; (ii) **Stage 2 original**, fixed first-candidate training stopped at epoch 2; (iii) **resampled epoch 2**, per-transcript category-aware resampling at the matched epoch count; and (iv) **resampled epoch 6**, the corrected run after four additional epochs, with gradient clipping during the extension.

Stage 1 reached 65.45% development R@1 after five epochs. The original Stage 2 development R@1 trajectory was 28.05%, 75.65%, and 82.20% at epochs 0–2. The corrected run developed more slowly, from 2.85% at epoch 0 to 41.35% at epoch 2 and 71.80% at epoch 6.

Three comparisons answer distinct questions. Stage 1 versus Stage 2 original describes the association between the supplied hard-negative stage and controlled discrimination. Original versus resampled epoch 2 is the closest available comparison of candidate delivery at a matched epoch count. Resampled epoch 2 versus epoch 6 measures progress under the corrected procedure. Original versus resampled epoch 6 is descriptive but cannot isolate sampling because duration and gradient clipping also differ. All reported checkpoints are from single training runs. Repeated-seed estimates were not available; therefore, the reported differences are descriptive and we do not report standard deviations, confidence intervals, or significance tests.

The 12,611-row automatic benchmark is the primary evaluation. The same checkpoints are then re-scored on the 731-row human-verified subset. Because the verification sample intentionally oversamples the 347-row numeral category and includes reviewer-corrected alternatives, its aggregate score is not directly comparable with the automatic benchmark's micro-average.

## 7 Results

### 7.1 Global retrieval and local discrimination diverge

The central result is visible without a separate table. Stage 1 retrieves the true transcript at rank 1 for 37.45% of benchmark queries in the 7,953-candidate pool, but scores the true transcript above its controlled alternative in only 53.66% of rows, just 3.66 points above the 50% pairwise baseline. Stage 2 original improves these values to 55.01% full-pool R@1 and 73.50% pairwise accuracy. Thus, non-trivial global retrieval can coexist with weak sensitivity to a controlled grammatical change; the fuller category-wise comparison appears in Table 3.

### 7.2 Matched-epoch effect of corrected delivery

At epoch 2, aggregate pairwise accuracy is 79.76% with the resampled procedure and 73.50% with the original. The most diagnostic change is tense, which rises from 52.83% to 77.70%. Negation rises from 77.25% to 90.44%. These are consistent with the purpose of exposing alternatives that were available in storage but systematically underdelivered by fixed-first selection.

The matched checkpoint is not uniformly superior. Case is 70.12% versus 77.16%, numeral is 57.06% versus 73.78%, and full-pool R@1 is 18.35% versus 55.01%. The convergence trajectories in Figure 2 indicate that the corrected negative distribution is associated with slower global-retrieval learning over the observed epochs. The epoch-2 result should therefore be interpreted as a redistribution toward controlled discrimination, not as a globally superior checkpoint.

### 7.3 Extended corrected training

At epoch 6, aggregate pairwise accuracy reaches 89.69%. Category scores are 83.78% for case, 86.52% for tense, 97.33% for negation, and 68.88% for the numeral control. Relative to the original checkpoint, the supplied differences are +6.62, +33.69, +20.08, and $-4.90$ points, respectively, with +16.19 points overall. Full-pool R@1/R@5/R@10 reaches 42.47/82.08/89.05%, below the original 55.01/90.06/94.01% checkpoint. Because this checkpoint additionally includes four epochs and gradient clipping, these values characterize the complete corrected training procedure rather than a sampler-only ablation.

Table 3: Automatic-benchmark pairwise accuracy and full-pool recall. All values are percentages. Δ is the supplied original-to-resampled-epoch-6 difference.

| Metric | Orig. | Resp.-2 | Resp.-6 | Δ |
|---|---|---|---|---|
| Case | 77.16 | 70.12 | **83.78** | +6.62 |
| Tense | 52.83 | 77.70 | **86.52** | +33.69 |
| Negation | 77.25 | 90.44 | **97.33** | +20.08 |
| Numeral control | **73.78** | 57.06 | 68.88 | −4.90 |
| **All pairwise** | 73.50 | 79.76 | **89.69** | +16.19 |
| Full-pool R@1 | **55.01** | 18.35 | 42.47 | −12.54 |
| Full-pool R@5 | **90.06** | 50.81 | 82.08 | −7.98 |
| Full-pool R@10 | **94.01** | 64.04 | 89.05 | −4.96 |

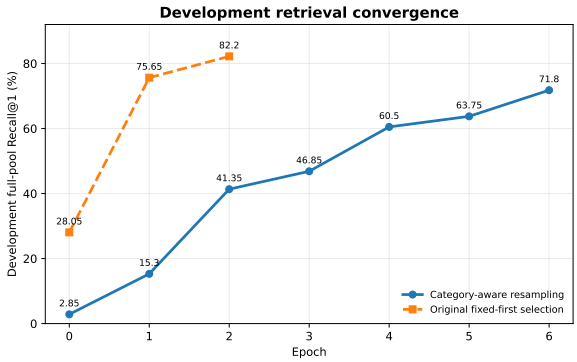



Figure 2: Development full-pool Recall@1 by epoch. These are observed trajectories from separate runs, not repeated-run uncertainty estimates.

**Progress within the corrected run.** From resampled epoch 2 to epoch 6, every supplied metric increases. Pairwise accuracy moves from 79.76% to 89.69%. Full-pool R@1/R@5/R@10 moves from 18.35/50.81/64.04% to 42.47/82.08/89.05%, corresponding to supplied gains of +24.12, +31.27, and +25.01 points. Pairwise accuracy gains 9.93 points. This shows that the severe epoch-2 global-retrieval deficit is at least partly associated with training progress. It does not establish that the corrected model would match or exceed the original after convergence, because no later checkpoint was supplied.

### 7.4 Human-verified evaluation

The verification outcome is summarized together with benchmark composition in Table 2. Across category reviewer–lead pairs, supplied raw agreement ranges from 82% to 95%, while Cohen's $\kappa$ ranges from 0.11 to 0.50; exact per-category values were not supplied. The lower end of the $\kappa$ range indicates weak chance-corrected agreement for at least one category; therefore, the 91.7% figure should be interpreted as a lead-adjudicated retention rate rather than evidence of uniformly strong annotator consensus. Most disagreements concern "accept unchanged" versus "accept with correction," rather than accept versus reject.

Table 4: Final checkpoint on the full automatic benchmark and the stratified human-verified subset. The columns are not directly comparable micro-averages because category mixture and text provenance differ.

| Category | Automatic | Verified | Δ |
|---|---|---|---|
| Case | 83.78 | 63.51 | −20.27 |
| Tense | 86.52 | 81.69 | −4.83 |
| Negation | 97.33 | 84.78 | −12.55 |
| Numeral control | 68.88 | 70.63 | +1.75 |
| All pairwise | 89.69 | 74.01 | −15.68 |

Table 4 re-scores the final checkpoint on the 731 retained rows. The 74.01% aggregate should not be read as a paired 15.68-point regression from the 89.69% automatic micro-average: the verified sample includes every numeral row but only 150 rows from each larger category, and 378 alternatives were rewritten by reviewers. Reweighting the automatic category scores to the verified sample's category proportions gives 80.69%. On the 353 rows accepted unchanged, observed accuracy is 75.35%, compared with a supplied category-mix-adjusted expectation of 74.98% (a +0.37-point residual). This supports the narrower conclusion that automatically generated rows confirmed by humans behave consistently with the full automatic benchmark, while reviewer corrections produce a different and generally harder diagnostic subset.

## 8 Analysis and Discussion

### 8.1 Global retrieval is not a morphology test

Full-pool retrieval rewards separation from thousands of candidates that often differ in topic, length, and lexical content. A model can exploit those broad cues without assigning a robust margin to a suffix-level contrast. Stage 1 combines 37.45% full-pool R@1 with 53.66% pairwise accuracy, demonstrating that the capabilities separate in this setting. Recall@K remains operationally necessary, but it should be accompanied by targeted diagnostics when a system is expected to preserve fine-grained grammar, as in contrast sets and minimal-pair suites for text NLP (Gardner et al., 2020; Ribeiro et al., 2020; Warstadt et al., 2020).

### 8.2 Candidate delivery is part of the method

The source file contained category diversity, but fixed position-zero selection converted it into a narrower effective distribution. At the matched epoch, tense rises from 52.83% to 77.70% under corrected

delivery, consistent with the diagnosis that later-triggering candidates were under-delivered. The corrected procedure does not equalize global category counts; it merely ensures that every category available for a transcript can reach the hinge loss. Reproducible reporting must therefore include when negative selection occurs, how many candidates are retained, and, where available, the category distribution actually observed in mini-batches rather than only the stored inventory.

### 8.3 What human verification changes

The 91.7% retained rate shows that the finite-state procedure generates a large proportion of usable controlled alternatives in the sampled data, but the 378 reviewer corrections also show that round-trip consistency is not sufficient for a high-precision linguistic benchmark. The verified subset is intentionally not a miniature random copy of the 12,611-row benchmark: it over-represents numeral controls and replaces many generated texts. Consequently, the verified aggregate should be interpreted as a secondary diagnostic of robustness to human-curated contrasts. The close match between observed and mix-adjusted expected accuracy on the 353 unchanged rows argues against the specific concern that obviously invalid automatic rows alone inflated the reported automatic score. It does not validate every unsampled row in the full benchmark.

### 8.4 Observed local–global behavior

The resampled model becomes substantially stronger on controlled pairs while global retrieval develops more slowly. At epoch 6, automatic pairwise accuracy is 89.69%, but full-pool R@1 remains below the original checkpoint. The supplied trajectory is still increasing, so incomplete convergence is plausible; an optimization trade-off between close negative discrimination and broad ranking is also possible. The available measurements do not distinguish these explanations. Numeral is the only category below the original at epoch 6. Its 347 automatic rows make scarcity a plausible concern, but no repeated runs or category-size ablation were supplied, so the cause remains unresolved.

### 8.5 Practical implications

The targeted failure mode is operationally relevant: an audio–text system may retrieve a broadly related recording while remaining insensitive to a case marker, tense marker, negation, or numeral that changes the intended interpretation. This matters in spoken-archive search, transcript quality assurance, call-centre compliance, dictation review, and confirmation workflows involving amounts or dates. The benchmark does not certify deployment readiness, but it demonstrates why full-pool retrieval should be paired with controlled contrast evaluation in morphologically rich languages.

## 9 Conclusion

We introduce an automatic multimodal Malayalam speech–text diagnostic that separates global retrieval from fine-grained controlled discrimination. A no-hard-negative model reaches 37.45% full-pool Recall@1 while remaining close to chance on one-change comparisons, showing that aggregate retrieval is not a proxy for sensitivity to grammatical contrasts. We identify a fixed-first-candidate loader defect that restricted which generated negatives reached training. Per-transcript category-aware retention with access-time resampling improves matched-epoch aggregate and tense pairwise accuracy, and the complete extended corrected procedure reaches 89.69% automatic pairwise accuracy. A stratified human review retains or corrects 731 of 797 sampled rows and shows that reviewer-edited contrasts form a harder secondary evaluation set. The remaining full-pool retrieval gap and the difference between automatic and human-curated evaluation remain open. More broadly, targeted linguistic diagnostics, exact negative-delivery logic, and human validation should be considered jointly when evaluating multimodal speech–text models for morphologically rich languages.

## 10 Limitations

The full 12,611-row benchmark is automatically generated; human review covers a stratified 797-row sample rather than every row. The lead reviewer adjudicated all sampled items, but the supplied record does not provide per-category agreement values beyond ranges, blinded reviewer details, or a fully independent third adjudication. The verified subset deliberately over-represents numeral controls and contains reviewer-corrected alternatives, so its micro-average is not directly comparable with the full automatic benchmark. Round-trip validation establishes finite-state consistency, not human naturalness, dialectal acceptability, or

semantic felicity. The benchmark contains three morphosyntactic categories and one lexical control and is not a complete analysis of Malayalam morphology. The grapheme-confusable category remains outside the reported benchmark. All results use one audio backbone and one text backbone. The original and corrected trajectories are separate runs, and the extended corrected run additionally uses gradient clipping. No repeated-run estimates were supplied, and the corrected model was not trained to a demonstrated plateau. Because each training condition was observed under a single seed, the magnitude and stability of the reported improvements across random initializations remain unquantified.

## References


Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2020. Common Voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association. https://aclanthology.org/2020.lrec-1.520/.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2022. XLS-R: Self-supervised cross-lingual speech representation learning at scale. In *Proceedings of Interspeech 2022*, pages 2278–2282. https://doi.org/10.21437/Interspeech.2022-143.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems 33*, pages 12449–12460.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI Publications, Stanford, California.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. FLEURS: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop*, pages 798–805. https://doi.org/10.1109/SLT54892.2023.10023141.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023. CLAP: Learning audio concepts from natural language supervision. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. https://doi.org/10.1109/ICASSP49357.2023.10095889.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323. https://doi.org/10.18653/v1/2020.findings-emnlp.117.

Tahir Javed, Kaushal Santosh Bhogale, Abhigyan Raman, Pratyush Kumar, Anoop Kunchukuttan, and Mitesh M. Khapra. 2023. IndicSUPERB: A speech processing universal performance benchmark for Indian languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):12942–12950. https://doi.org/10.1609/aaai.v37i11.26521.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550.

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. *Computing Research Repository*, arXiv:2103.10730.

Sameer Khurana, Antoine Laurent, and James Glass. 2022. SAMU-XLSR: Semantically-aligned multimodal utterance-level cross-lingual speech representation. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1493–1504. https://doi.org/10.1109/JSTSP.2022.3192714.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124. https://doi.org/10.18653/v1/2021.naacl-main.324.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North*

*American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847. https://doi.org/10.18653/v1/2021.naacl-main.466.

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*, pages 8748–8763.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912. https://doi.org/10.18653/v1/2020.acl-main.442.

Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *International Conference on Learning Representations*. https://arxiv.org/abs/2010.04592.

Yi-Jen Shih, Hsuan-Fu Wang, Heng-Jui Chang, Layne Berry, Hung-yi Lee, and David Harwath 2022. SpeechCLIP: Integrating speech with pretrained vision and language model. In *2022 IEEE Spoken Language Technology Workshop.*

Santhosh Thottingal. 2019. Finite state transducer based morphology analysis for Malayalam language. In *Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages*, pages 1–5.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *Computing Research Repository*, arXiv:1807.03748.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392. https://doi.org/10.1162/tacl_a_00321.

Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. 2022. Wav2CLIP: Learning robust audio representations from CLIP. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4563–4567. https://doi.org/10.1109/ICASSP43922.2022.9747669.

Rong Ye, Mingxuan Wang, and Lei Li. 2022. Cross-modal contrastive learning for speech translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5099–5113. https://doi.org/10.18653/v1/2022.naacl-main.376.

Yuguo Yin, Yuxin Xie, Wenyuan Yang, Dongchao Yang, Jinghan Ru, Xianwei Zhuang, Liming Liang, and Yuexian Zou. 2025. ATRI: Mitigating multilingual audio text retrieval inconsistencies by reducing data distribution errors. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5491–5504, Vienna, Austria. Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.acl-long.273.

Wenzheng Zhang and Karl Stratos. 2021. Understanding hard negatives in noise contrastive estimation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1090–1101. https://doi.org/10.18653/v1/2021.naacl-main.86.

# A Retrieval-Augmented Generation Approach for Enhancing Pedagogical Support in Education

**Ramkumar N**
Coimbatore Institute of Technology
ramkumar@cit.edu.in

**Rajendran S**
Coimbatore Institute of Technology
asrajendrayadav@gmail.com

## Abstract

The manual creation of high-quality study materials such as summaries, question–answer sets, and flashcards remains a time-consuming task for both educators and learners. While large language models can generate such content rapidly, they frequently suffer from hallucinations and lack grounding in the source material. This paper presents a Retrieval-Augmented Generation (RAG) framework designed to address these limitations by transforming educational documents into pedagogically structured learning resources. The system integrates document preprocessing, semantic chunking, embedding-based retrieval, and a locally deployed language model to produce concise summaries, targeted questions, and flashcards. A cross-sectional evaluation involving 48 undergraduate students and 12 educators was conducted to assess usability, accuracy, cognitive support, and overall pedagogical value. Findings indicate that the AI Study Agent is perceived as supportive of comprehension and revision activities, and objective evaluation supports factual consistency relative to non-retrieval generation. Challenges related to output accuracy, oversimplification, and user trust remain consistent with limitations identified in existing GenAI literature. The study concludes that while the system demonstrates strong potential to support teaching and learning workflows, its perceived effectiveness relies on responsible use, educator oversight, and continued refinement to ensure pedagogical alignment and reliability.

## 1 Introduction

Generative Artificial Intelligence has catalyzed profound transformations across education. Large Language Models now power intelligent tutoring systems, automated assessment, and adaptive learning platforms, offering unprecedented scalability in content creation (Dwivedi et al., 2023; Kasneci et al., 2023). Within this ecosystem, automated creation of structured study materials (summaries, question–answer sets, and flashcards) represents a particularly promising frontier. These resources serve as foundational scaffolds for active recall and self-assessment, strategies empirically validated for durable learning (Dunlosky et al., 2013).

Manual production of such materials, however, remains labor-intensive and scales poorly with increasing class sizes. Pure generative models offer a tempting shortcut, yet introduce significant liabilities: ungrounded outputs, factual hallucinations, and lack of domain-specific fidelity can undermine the comprehension they aim to support (Ji et al., 2023). Existing commercial tools frequently lack pedagogical alignment, transparency regarding source provenance, and local deployment options essential for institutional data sovereignty. Many operate as opaque black boxes with limited configurability, hindering trustworthy integration into formal learning workflows. This paper addresses these challenges through the design, implementation, and empirical evaluation of a Retrieval-Augmented Generation (RAG) framework for pedagogical support. The contributions are fourfold: (1) a fully articulated RAG pipeline encompassing document ingestion, semantic chunking, embedding-based retrieval, and local LLM inference; (2) generation of structured pedagogical outputs—summaries, question–answer sets, and flashcards—aligned with cognitive load reduction and active learning principles; (3) cross-sectional empirical evaluation with students and educators capturing both objective metrics and subjective perceptions; and (4) insights on residual hallucination, oversimplification, and trust calibration that inform responsible AI deployment in education.

## 2 Related Work

Generative Artificial Intelligence has transformed educational content creation, with Large Language



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 28–39
November 9, 2026 ©2026 Association for Computational Linguistics

Models now powering tutoring systems, assessment generation, and personalized feedback (Kasneci et al., 2023; Dwivedi et al., 2023). Critical scholarship documents significant risks: hallucinations, erosion of epistemic vigilance, and the potential for generative outputs to supplant rather than scaffold knowledge construction (Ji et al., 2023; Sullivan et al., 2023). The persuasive surface validity of model outputs can induce unwarranted confidence among learners, particularly when domain expertise is insufficient to detect subtle distortions.

Research on automated study materials spans summarization (Allahyari et al., 2017; Nenkova and McKeown, 2011), question generation (Kurdi et al., 2020; Alsubait, 2015), and flashcard creation (Dunlosky et al., 2013; Karpicke and Roediger, 2008). Existing tools frequently operate as opaque, proprietary services that prioritize lexical fluency over epistemic accuracy, generating plausible but potentially distorted content (Chiu et al., 2023; Bender et al., 2021). Many lack explicit pedagogical alignment, offering limited configurability for chunking strategies, retrieval parameters, or prompt engineering essential for fine-tuning outputs to specific educational contexts.

Retrieval-Augmented Generation addresses hallucination by grounding outputs in verifiable source documents (Lewis et al., 2020; Guu et al., 2020). Educational applications of RAG have shown potential for improving the factual consistency and contextual grounding of generated learning materials, while recent work has also explored large language models for educational question generation (Lamsiyah et al., 2024). Existing work has generally focused on individual educational artifacts or tutoring interactions. Comparatively less attention has been given to the *unified* generation of summaries, question–answer sets, and flashcards within a single, locally deployable hybrid RAG framework that is evaluated by both learners and educators. This gap motivates the present work: an integrated system that prioritizes institutional data sovereignty, pedagogical alignment, and transparent source grounding.

## 3 Theoretical Framework

The following three theories motivate both the system architecture and the evaluation framework.

**Constructivist Learning Theory** positions AI-generated materials as scaffolds for active knowledge construction rather than inert knowledge repositories (Wood et al., 1976). The RAG pipeline operationalizes this through question–answer sets demanding active retrieval, flashcards cueing spaced rehearsal, and summaries preserving conceptual density for inferential reasoning. Source-grounded retrieval mitigates the risk of passive acceptance by positioning outputs as verifiable derivatives of authoritative texts.

**Cognitive Load Theory** distinguishes intrinsic, extraneous, and germane demands on working memory (Sweller, 1988). Structured summaries reduce extraneous load by eliminating redundancy and signaling hierarchical relationships; targeted questions channel resources toward germane processing. The chunking strategy and prompt engineering reflect this rationale, while the evaluation assesses whether perceived cognitive support aligns with theoretical predictions. Oversimplification and residual inaccuracies are treated as risks that may paradoxically increase load by forcing learners to reconstruct missing connections.

**Human–AI Interaction Theory** frames trust as a calibrated response shaped by reliability and transparency (Lee and See, 2004; Hoff and Bashir, 2015). Local deployment and source attribution support appropriate trust calibration, while the evaluation instruments measure perceived transparency and usability as distinct yet interdependent constructs. Educators are positioned as curators rather than passive consumers, retaining agency to validate and contextualize outputs.

## 4 System Design and Architecture

### 4.1 Overall Architecture

The system architecture implements a Retrieval-Augmented Generation pipeline that transforms raw educational documents into structured pedagogical artifacts through modular, sequential stages. Figure 1 illustrates the complete workflow. Document ingestion and preprocessing feed parallel retrieval streams: a dense semantic retriever employing sentence-transformer embeddings with cosine similarity, and a sparse lexical retriever employing BM25 scoring. The dual retrieval streams are fused via reciprocal rank fusion (RRF) to produce a consolidated context window of the top-$k$ most relevant chunks. This context, together with a task-specific prompt template, conditions the locally deployed large language model to generate one of three output types: narrative summaries, question–answer sets, or flashcards. Modularity ensures that

each functional component can be independently optimized, replaced, or audited without compromising broader system integrity. The pipeline operates entirely within local infrastructure, thereby eliminating external API dependencies and ensuring that sensitive educational materials remain within institutional boundaries.

### 4.2 Document Ingestion and Preprocessing

Document ingestion accommodates multiple formats including PDF, DOCX, and plain text, reflecting the heterogeneous document ecosystems typical of academic environments. Preprocessing routines perform structural normalization by removing headers, footers, and extraneous metadata while preserving semantic paragraph boundaries. Text chunking employs a hybrid strategy that combines recursive character splitting with semantic boundary detection; individual chunks are constrained to approximately 512 tokens with 64-token overlaps. This chunk size represents a deliberate compromise between retrieval granularity and contextual coherence. Smaller fragments improve precision yet risk severing conceptual relationships that span multiple sentences, whereas excessively large chunks dilute semantic specificity and introduce irrelevant context into generation prompts. Overlap mechanisms mitigate boundary effects by ensuring that transitional passages appear in adjacent chunks, thereby preserving thematic continuity across segment boundaries.

### 4.3 Embedding and Vector Storage

Chunk embeddings are computed using the `sentence-transformers/all-MiniLM-L6-v2` model, a 22.7M-parameter sentence-transformer that produces 384-dimensional dense vectors via mean-pooling over the final hidden states (Reimers and Gurevych, 2019). Input preprocessing truncates sequences to a maximum of 256 word-piece tokens and applies lowercase normalization and whitespace collapsing. No additional domain-specific fine-tuning is performed. Vector similarity is computed using cosine distance in the $L_2$-normalized embedding space.

Dense vector representations are stored in ChromaDB 0.5.0 with HNSW indexing (efConstruction = 128, M = 16), enabling approximate nearest-neighbor search with persistent local storage (Team, 2022). The selection of ChromaDB reflects a design priority for lightweight deployment without container orchestration overhead.

Table 1: Chunk Size and Overlap Ablation ($n = 30$ documents)

| Configuration | Precision@3 | Recall@5 | ROUGE-L F1 | Factual Consistency |
|---|---|---|---|---|
| $c = 256, o = 32$ | 0.71 | 0.64 | 0.62 | 3.78 |
| $c = 256, o = 64$ | 0.73 | 0.66 | 0.63 | 3.82 |
| $c = 256, o = 128$ | 0.72 | 0.65 | 0.61 | 3.75 |
| $c = 512, o = 32$ | 0.76 | 0.69 | 0.68 | 4.05 |
| $c = 512, o = 64$ | **0.78** | **0.71** | **0.71** | **4.12** |
| $c = 512, o = 128$ | 0.77 | 0.70 | 0.69 | 4.08 |
| $c = 1024, o = 64$ | 0.72 | 0.67 | 0.65 | 3.91 |
| $c = 1024, o = 128$ | 0.71 | 0.66 | 0.64 | 3.88 |
| $c = 1024, o = 256$ | 0.70 | 0.65 | 0.62 | 3.84 |

The choice of chunk size ($c = 512$ tokens) and overlap ($o = 64$ tokens) was validated through an ablation study rather than selected heuristically. Table 1 reports retrieval and generation performance across nine configurations on a held-out validation set of 30 documents. Retrieval quality is measured by Precision@3 and Recall@5 against manually annotated relevance judgments. Generation quality is measured by ROUGE-L F1 and expert-rated factual consistency (1–5 scale) for summarization outputs.

The configuration $c = 512$, $o = 64$ achieved the optimal balance: smaller fragments ($c = 256$) improved retrieval precision marginally but degraded generation coherence due to severed conceptual relationships, while larger chunks ($c = 1024$) diluted semantic specificity and introduced irrelevant context into generation prompts. The 64-token overlap was sufficient to preserve boundary continuity without excessive storage redundancy. This configuration was fixed for all subsequent experiments.

### 4.4 Retrieval and Generation

Dense retrieval identifies the top-$m = 10$ candidates via cosine similarity over the 384-dimensional embedding space, capturing paraphrastic and conceptual relationships that elude exact term matching. Sparse retrieval identifies the top-$m = 10$ candidates via BM25 with default parameters ($k_1 = 1.5$, $b = 0.75$), ensuring precise recovery of rare terminological and definitional content that dense representations may dilute. The two ranked lists are fused through reciprocal rank fusion with constant $k_{\text{RRF}} = 60$:

$$\text{RRF}(d) = \sum_{r \in R} \frac{1}{k_{\text{RRF}} + \text{rank}_r(d)}$$

where $R$ denotes the set of retrievers and $\text{rank}_r(d)$ the rank of document $d$ in retriever $r$. The fused list is truncated to $k = 5$ chunks, which

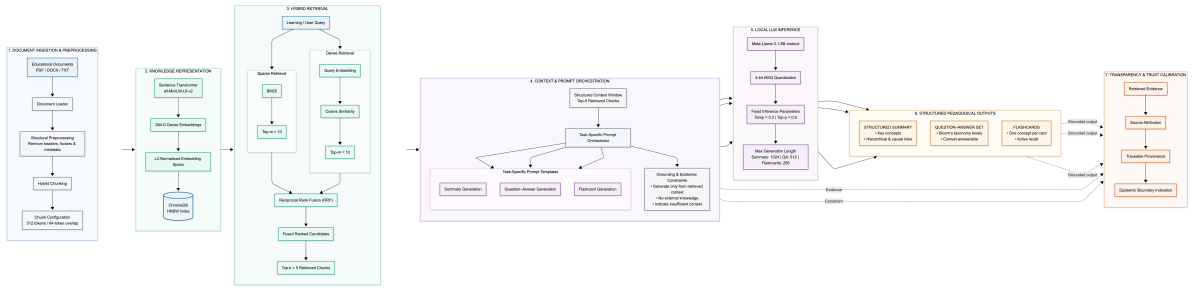


Figure 1: System Architecture. End-to-end pipeline from document ingestion to grounded pedagogical outputs (summaries, QA pairs, and flashcards), with hybrid retrieval, local LLM inference, and explicit transparency mechanisms.

are concatenated into a structured context window. This value of $k = 5$ was selected based on the context-window constraints of the generation model and the empirical observation that additional chunks beyond the fifth contributed diminishing relevance while increasing the risk of context dilution. The Full System significantly outperforms both unimodal retrieval baselines and the hybrid configuration without RRF, confirming that the fusion mechanism captures complementary signals rather than merely aggregating redundant ones.

Retrieved chunks are injected into task-specific prompt templates that constitute a secondary contribution. Three distinct templates govern output generation, each engineered to align with specific pedagogical objectives:

- *Summarization*: instructs condensation while preserving hierarchical concept relationships and key terminology, with explicit constraints against omission of causal or comparative relationships;
- *Question generation*: demands varied cognitive levels ranging from factual recall to analytical inference, with explicit constraints on answerability from the provided context and prohibitions against tautological or overly generic items;
- *Flashcard generation*: enforces concise question-answer pairs suitable for spaced repetition, with a maximum of 128 tokens per card face and requirements that each pair target a single, verifiable concept.

All templates include explicit grounding instructions that constrain the model to synthesize responses exclusively from retrieved context and to indicate when a request cannot be answered from the provided material. This design directly operationalizes the theoretical commitment to transparency: by mandating that the generator acknowledge epistemic boundaries, the system mitigates the risk of confabulation that persists even under retrieval augmentation.

Generation is performed by `meta-llama/Llama-3.1-8B-Instruct`, deployed locally via 4-bit AWQ quantization (group size 128, zero point quantization). Inference parameters are fixed across all tasks: temperature $= 0.3$, top-$p$ $= 0.9$, repetition penalty $= 1.1$, maximum generation length $= 1024$ tokens for summaries, $512$ tokens for question-answer sets, and $256$ tokens for flashcards. The low temperature prioritizes factual consistency and structural adherence over lexical diversity, an appropriate trade-off for educational content where precision supersedes creativity. Local deployment constitutes a core architectural decision motivated by three interlocking considerations: institutional data sovereignty, which precludes transmission of curricular materials to third-party APIs; elimination of network latency and external service dependencies, which ensures reliable classroom deployment; and mitigation of vendor lock-in, which preserves institutional autonomy over pedagogical infrastructure. These considerations reflect the theoretical mandate from Human-AI Interaction Theory that trust and transparency depend not merely on interface design but on underlying architectural control.

### 4.5 Output Types

The system produces three structured output types through a unified retrieval-augmented pipeline, ensuring consistent source grounding while accommodating distinct pedagogical functions.

**Summaries** condense source documents into

narrative overviews that preserve core arguments and terminological precision. The summarization template instructs the model to maintain hierarchical concept relationships and causal links, avoiding the aggressive compression that strips contextual qualifiers necessary for advanced understanding.

**Question–answer sets** comprise paired items spanning multiple cognitive levels, from definitional recall to application and synthesis. The generation template demands varied cognitive demands aligned with a modified Bloom's taxonomy, with explicit constraints prohibiting tautological items or questions unanswerable from the provided context. These sets support both formative self-assessment and instructor-led review.

**Flashcards** adhere to a strict question-answer format optimized for active recall and spaced repetition protocols. Each card targets a single verifiable concept with a maximum of 128 tokens per face, ensuring concision suitable for rapid review cycles.

The tripartite design reflects the theoretical commitment to reducing extraneous cognitive load while supporting active knowledge construction.

# 5 Methodology

## 5.1 Participants and Procedure

A cross-sectional design was employed to capture user interactions and perceptions at a single temporal point. The sample comprised $N = 60$ participants purposively recruited from a single higher-education institution offering programmes in artificial intelligence and data science: $n = 48$ undergraduate students (24 male, 22 female, 2 non-binary; $M_{\text{age}} = 20.7$, $SD = 1.4$) distributed across second-year ($n = 16$), third-year ($n = 20$), and final-year ($n = 12$) study levels, and $n = 12$ educators (7 male, 5 female; $M_{\text{age}} = 38.4$, $SD = 6.2$) with 2 to 19 years of teaching experience ($M = 9.3$, $SD = 5.1$). All students were enrolled in or had recently completed a Fundamentals of Artificial Intelligence module, ensuring sufficient domain familiarity to evaluate output accuracy and pedagogical relevance. Inclusion required functional English proficiency and basic digital literacy; exclusion comprised prior exposure to the prototype or involvement in its development. Students reported moderate prior GenAI familiarity ($M = 3.1$, $SD = 1.0$ on a 5-point scale); educators reported lower familiarity ($M = 2.4$, $SD = 1.1$), with only 25% having integrated generative AI into formal instruction.

Participants completed a single 45–60 minute session with the prototype loaded with Fundamentals of AI materials. Students performed document upload, summary generation, question–answer set generation, flashcard creation, and output review. Educators performed identical interactions and additionally evaluated outputs against professional standards for pedagogical alignment, terminological precision, and classroom suitability. Task completion triggered administration of the structured questionnaire. All interactions and temporal markers were captured via automated logs.

## 5.2 Measures and Analysis

Data comprised interaction logs, a structured questionnaire, and expert artifact evaluations. Logs recorded uploads, generation requests, refinement iterations, and session duration. The questionnaire operationalized three theoretical constructs across four domains—Perceived Usefulness, Accuracy and Reliability, Perceived Cognitive Support, and Usability—with 16 five-point Likert items (1 = Strongly Disagree, 5 = Strongly Agree). Pilot testing ($n = 15$) confirmed internal consistency ($\alpha \geq .87$ for all domains). Complete items appear in Appendix A.

Objective artifact quality was evaluated by two domain experts. Summaries ($n = 40$) were rated on factual consistency (claim-level support against the source), completeness (retention of key concepts), relevance (alignment with inferred information need), and compression ratio (summary word count / source word count). Question–answer sets ($n = 120$) were assessed on correctness, answerability, relevance, and cognitive-level validity (match to intended Bloom's taxonomy level). Flashcards ($n = 80$) were assessed on factual correctness, pedagogical usefulness, and coverage (proportion of key terms represented). Inter-rater reliability was substantial ($\kappa \geq .82$).

Ablation analysis compared five system variants on a held-out corpus of 50 document sections: Vanilla LLM, Dense-only RAG, Sparse-only RAG, Hybrid without RRF, and the Full System. Each of the 50 document sections was processed independently by all five system variants, yielding repeated measurements for each document and enabling within-document comparisons. Two experts rated factuality on a 1–5 scale. Metrics included citation precision (verifiable claims against retrieved context; against full source for Vanilla LLM), hallucination rate (unsupported plus contra-

Table 2: Retrieval Performance Metrics Across System Configurations

| Metric | Dense-only | Sparse-only | Hybrid (no RRF) | Full System |
|---|---|---|---|---|
| Precision@3 | 0.62 | 0.58 | 0.71 | **0.78** |
| Recall@5 | 0.54 | 0.51 | 0.63 | **0.71** |
| MRR | 0.68 | 0.64 | 0.74 | **0.82** |
| nDCG@5 | 0.59 | 0.56 | 0.67 | **0.76** |

dicted claims), and ROUGE-L F1 against human-written reference summaries prepared by graduate teaching assistants instructed to condense source documents to approximately 30% length while preserving key concepts. ROUGE-L served as a lexical overlap metric, distinct from expert-rated factual consistency. Quantitative analysis employed descriptive statistics and independent-samples $t$-tests for subgroup comparisons. Effect sizes are reported as Cohen's $d$ with 95% confidence intervals. Optional open-text responses were analysed thematically to identify recurrent patterns (Braun and Clarke, 2006). Three evaluation subsets were used. A 30-document validation set was used for chunk-size/overlap selection, a manually annotated retrieval set was used to evaluate retrieval effectiveness, and a separate 50-document held-out corpus was used for the five-way system ablation.

## 6 Results

### 6.1 System Performance

Retrieval effectiveness was evaluated against 150 manually annotated query–chunk relevance judgments ($\kappa = .82$). The Full System achieved Precision@3 of 0.78, Recall@5 of 0.71, MRR of 0.82, and nDCG@5 of 0.76 (Table 2), outperforming both unimodal baselines and the hybrid configuration without RRF. The MRR gain of 0.08 attributable to RRF indicates that fusion elevates the most relevant chunks beyond the simple concatenation of retrieval streams.

Generated artifacts demonstrated strong overall quality across the three output types. Summaries ($n = 40$) achieved mean factual consistency of 4.12 ($SD = 0.68$), completeness of 3.85 ($SD = 0.74$), and relevance of 4.05 ($SD = 0.61$), with a mean compression ratio of 0.32 ($SD = 0.08$) (Table 3). The completeness score was marginally lower than consistency, reflecting occasional omission of secondary concepts during condensation.

Question–answer sets ($n = 120$) demonstrated strong relevance ($M = 4.21$, $SD = 0.58$) and answerability ($M = 4.08$, $SD = 0.66$), with lower correctness ($M = 3.94$, $SD = 0.79$) (Table 4). Discrepancies concentrated in items requiring inferential reasoning beyond explicit source statements. Cognitive-level validity reached 0.74 ($SD = 0.18$); misclassifications predominantly involved remember-level items generated when understand-level items were solicited.

Table 3: Objective Evaluation of Generated Summaries ($n = 40$)

| Criterion | Scale | $M$ | $SD$ |
|---|---|---|---|
| Factual Consistency | 1–5 | 4.12 | 0.68 |
| Completeness | 1–5 | 3.85 | 0.74 |
| Relevance | 1–5 | 4.05 | 0.61 |
| Compression Ratio | Proportion | 0.32 | 0.08 |

Table 4: Objective Evaluation of Generated Question–Answer Sets ($n = 120$)

| Criterion | Scale | $M$ | $SD$ |
|---|---|---|---|
| Correctness | 1–5 | 3.94 | 0.79 |
| Answerability | 1–5 | 4.08 | 0.66 |
| Relevance | 1–5 | 4.21 | 0.58 |
| Cognitive-Level Validity | Proportion correct | 0.74 | 0.18 |

Flashcards ($n = 80$) achieved factual correctness of 4.05 ($SD = 0.71$) and pedagogical usefulness of 3.88 ($SD = 0.82$), with coverage of 0.68 ($SD = 0.14$) (Table 5). The majority of key terms were represented, though specialized terminology in dense conceptual clusters was occasionally overlooked. Interaction logs recorded mean generation latency of 2.3 s for summaries, 1.1 s for question–answer sets, and 0.8 s for flashcards. Refinement requests constituted 12.4% of total interactions.

Participants reported an estimated 68.1% reduction in preparation time relative to *recalled* manual baselines ($M = 47.3$ minutes vs. $M = 15.1$ minutes, 95% CI $[65.7\%, 70.5\%]$). These figures reflect self-reported estimates rather than observed performance; no controlled manual baseline condition was administered. Interaction logs corroborated the AI-assisted duration ($M = 13.8$ minutes, $SD = 3.9$). We therefore interpret the reported time reduction as an indication of perceived efficiency gains contingent on successful workflow integration, not as causal evidence of productivity improvement.

Table 6 reports the distribution of individual reduction ratios.

### 6.2 User Perceptions

The final sample comprised $N = 60$ participants: $n = 48$ students and $n = 12$ educators. The instrument demonstrated strong internal consistency

Table 5: Objective Evaluation of Generated Flashcards ($n = 80$)

| Criterion | Scale | $M$ | $SD$ |
|---|---|---|---|
| Factual Correctness | 1–5 | 4.05 | 0.71 |
| Pedagogical Usefulness | 1–5 | 3.88 | 0.82 |
| Coverage | Proportion | 0.68 | 0.14 |

Table 6: Distribution of Self-Reported Time Reduction Ratios ($N = 60$)

| Statistic | Value |
|---|---|
| Minimum | 45.2% |
| 25th Percentile | 62.1% |
| Median | 69.3% |
| 75th Percentile | 74.5% |
| Maximum | 84.6% |
| Mean ($SD$) | 68.1% (9.4%) |

($\alpha \geq .87$ for all domains).

Students reported significantly higher perceived usefulness, $t(58) = 3.02$, $p = .003$, $d = 0.97$ [0.32, 1.61], and perceived cognitive support, $t(58) = 2.62$, $p = .010$, $d = 0.84$ [0.20, 1.47], than educators. No significant group difference emerged for perceived accuracy and reliability, $t(58) = 1.15$, $p = .255$, $d = 0.37$ [−0.27, 1.00], or usability, $t(58) = 1.89$, $p = .064$, $d = 0.61$ [−0.03, 1.24]. Both groups identified summarization as the most beneficial feature.

### 6.3 Ablation Analysis

Table 7 reports performance across the five system variants. A one-way repeated-measures ANOVA documented significant condition effects for factuality, $F(4, 196) = 24.37$, $p < .001$, $\eta_p^2 = .33$; citation precision, $F(4, 196) = 41.52$, $p < .001$, $\eta_p^2 = .46$; hallucination rate, $F(4, 196) = 28.91$, $p < .001$, $\eta_p^2 = .37$; and ROUGE-L F1, $F(4, 196) = 19.84$, $p < .001$, $\eta_p^2 = .29$.

Bonferroni-corrected comparisons revealed that the Full System significantly outperformed the Vanilla LLM on all metrics (all $p < .001$, $d$ = 1.78–2.34). It also surpassed both Dense-only and Sparse-only configurations (all $p \leq .003$, $d$ = 0.82–1.21), confirming that hybrid retrieval captures complementary signals. Most critically, the Full System exceeded the Hybrid without RRF variant on factuality, $t(49) = 2.72$, $p = .009$, $d = 0.62$, and citation precision, $t(49) = 3.14$, $p = .003$, $d = 0.78$, demonstrating that reciprocal rank fusion contributes measurably beyond simple concatenation.

Table 7: Ablation Study Results Across System Variants ($N = 50$ documents)

| Variant | Factuality $M$ ($SD$) | Citation Precision $M$ ($SD$) | Hallucination Rate $M$ ($SD$) | ROUGE-L F1 $M$ ($SD$) |
|---|---|---|---|---|
| Vanilla LLM | 2.84 (0.91) | 0.12 (0.08) | 0.34 (0.11) | 0.42 (0.10) |
| Dense-only RAG | 3.62 (0.72) | 0.68 (0.12) | 0.18 (0.09) | 0.58 (0.09) |
| Sparse-only RAG | 3.44 (0.81) | 0.64 (0.14) | 0.21 (0.10) | 0.55 (0.11) |
| Hybrid without RRF | 3.88 (0.64) | 0.74 (0.10) | 0.14 (0.07) | 0.63 (0.08) |
| **Full System (Hybrid + RRF)** | **4.28 (0.52)** | **0.86 (0.08)** | **0.08 (0.06)** | **0.71 (0.07)** |

## 7 Discussion

### 7.1 Interpretation of Findings

The ablation study confirms that hybrid retrieval with reciprocal rank fusion yields measurable gains in factual consistency beyond both non-retrieval generation and unimodal retrieval baselines. The Full System achieved a hallucination rate of 0.08, representing a 76.5% reduction from the Vanilla LLM baseline, with disproportionate attenuation of unsupported claims. These improvements are attributable to the architecture itself. Dense retrieval excels at capturing paraphrastic and conceptual relationships, while sparse retrieval preserves precision for rare terminological and definitional content. Reciprocal rank fusion exploits the complementary strengths of both modalities, re-ranking candidates to elevate the most relevant chunks beyond what either stream achieves in isolation. The divergence between objective factual consistency and subjective trust ratings merits attention. The objective hallucination rate of 0.08 contrasts with perceived accuracy ratings of 3.45 on a five-point scale, indicating that participants maintained appropriate epistemic caution despite retrieval grounding. This pattern aligns with Human-AI Interaction Theory: transparency mechanisms, including source attribution and traceable provenance, enable users to calibrate trust without surrendering critical judgment.

### 7.2 Pedagogical Implications

Participant perceptions support the theoretical framing of the system as a scaffold rather than an autonomous instructional authority. Students reported significantly higher perceived usefulness and cognitive support than educators, a pattern consistent with Constructivist Learning Theory and its emphasis on learner agency. For students, structured sum-

maries and question sets functioned as temporary supports that reduced extraneous cognitive load and freed working memory resources for schema construction. Educators, by contrast, exhibited greater circumspection regarding accuracy and pedagogical alignment, reflecting their institutional responsibility for curriculum fidelity. This tension underscores that AI-generated materials facilitate active engagement only when deployed under human oversight. Self-reported time estimates indicated substantial perceived reduction in preparation time, though the absence of a controlled manual baseline precludes causal inference. The system should be understood as offering potential for reduced preparation effort contingent on workflow integration, not as empirically demonstrated productivity enhancement.

## 8 Limitations

The cross-sectional design captures perceptions at a single point in time and therefore cannot assess longitudinal effects such as sustained trust calibration or actual changes in knowledge retention. Claims regarding comprehension benefits and cognitive support consequently reflect participant perception rather than measured learning gains. The evaluation was restricted to a single higher-education institution and a single course domain. Performance may differ in disciplines with distinct epistemological structures or denser technical terminology. The sample size ($N = 60$) further limits generalizability. Despite retrieval grounding, residual hallucination and oversimplification of nuanced concepts persist. Educators consistently emphasized the need for human oversight. These limitations reinforce that the system is best deployed as a pedagogical scaffold under educator supervision rather than as an autonomous instructional authority.

## 9 Conclusion

This study presented a locally deployable Retrieval-Augmented Generation framework for generating structured educational materials, including summaries, question–answer sets, and flashcards. The architecture integrates hybrid dense–sparse retrieval with reciprocal rank fusion, local quantized inference, and task-specific prompt engineering. The 76.5% reduction in hallucination rate, driven primarily by attenuation of unsupported claims, provides direct empirical support for the central architectural motivation. A cross-sectional evaluation with 60 participants yielded converging evidence from objective system metrics and subjective user perceptions. Students reported significantly higher perceived usefulness and cognitive support than educators, while both cohorts shared comparable levels of perceived accuracy and trust calibration. Residual challenges, including occasional inaccuracies, oversimplification of complex topics, and persistent trust calibration concerns, reinforce that the system functions most effectively as a pedagogical scaffold under educator oversight rather than as an autonomous instructional authority. Future work should prioritize longitudinal experimental designs with pre/post knowledge assessments and delayed retention measures to establish whether perceived benefits translate into measurable learning outcomes. Multi-domain evaluation is essential to determine generalizability beyond artificial intelligence education. Technical priorities include improved hallucination mitigation through chain-of-verification generation, confidence indicators, and expanded personalization features that accommodate heterogeneous learning levels. Hybrid human–AI curation workflows, wherein educators actively validate and contextualize generated materials, represent the most promising direction for ensuring pedagogical alignment and ethical soundness in AI-supported learning environments.

## AI Usage Statement

The authors used generative AI only for auxiliary writing support, including grammar correction, and wording refinement. The technical contributions were developed and validated by the authors. The authors take full responsibility for the content of the paper.

## Ethics Statement

This study involved human participants (students and educators) who interacted with the system and completed questionnaires. All participants provided informed consent prior to participation. No personally identifiable information was collected, and all interaction logs were anonymized. The educational materials used in the evaluation were institutionally approved course content. The system is designed as a pedagogical support tool and is not intended to replace human educators. We emphasize that generated outputs should always be reviewed by instructors before classroom use.

## References


Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. 2017. A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*.

Tahani Alsubait. 2015. A review of ontology-based question generation techniques. *Journal of King Saud University-Computer and Information Sciences*, 27(2):145–153.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.

Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101.

Thomas K. F. Chiu, Qing Xia, Xiaoqin Zhou, Ching Sing Chai, and Miao Cheng. 2023. Systematic literature review on opportunities, challenges, and future research recommendations of artificial intelligence in education. *Computers and Education: Artificial Intelligence*, 4:100118.

John Dunlosky, Katherine A. Rawson, Elizabeth J. Marsh, Mitchell J. Nathan, and Daniel T. Willingham. 2013. Improving students' learning with effective learning techniques: Promising directions from cognitive and educational psychology. *Psychological Science in the Public Interest*, 14(1):4–58.

Yogesh K. Dwivedi, Nir Kshetri, Laurie Hughes, Emma L. Slade, Anand Jeyaraj, Arpan Kumar Kar, Abdullah M. Baabdullah, Alex Koohang, Vishnupriya Raghavan, Manju Ahuja, et al. 2023. So what if ChatGPT wrote it? Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *International Journal of Information Management*, 71:102642.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. *Proceedings of the 37th International Conference on Machine Learning*, pages 3929–3938.

Kevin A. Hoff and Masooda Bashir. 2015. Trust in automation: Integrating empirical evidence on factors that influence trust. *Human Factors*, 57(3):407–434.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Jeffrey D. Karpicke and Henry L. Roediger. 2008. The critical importance of retrieval for learning. *Science*, 319(5865):966–968.

Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert, Darya Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. 2023. Chatgpt for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274.

Gihan Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, 30:121–204.

Salima Lamsiyah, Abdelkader El Mahdaouy, Aria Nourbakhsh, and Christoph Schommer. 2024. Fine-tuning a large language model with reinforcement learning for educational question generation. In *Artificial Intelligence in Education*, pages 424–438. Springer Nature Switzerland.

John D. Lee and Katrina A. See. 2004. Trust in automation: Designing for appropriate reliance. *Human Factors*, 46(1):50–80.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ani Nenkova and Kathleen McKeown. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2–3):103–233.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992.

Michael Sullivan, Ann Kelly, and Peter McLaughlan. 2023. Chatgpt in higher education: Considerations for academic integrity and student learning. *Journal of Applied Learning and Teaching*, 6(1):31–37.

John Sweller. 1988. Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2):257–285.

Chroma Team. 2022. Chroma: The ai-native open-source embedding database. `https://github.com/chroma-core/chroma`. Accessed: 2024.

David Wood, Jerome S. Bruner, and Gail Ross. 1976. The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, 17(2):89–100.

# A Questionnaire Items

Table 8 presents the 16 questionnaire items. All items employ a five-point Likert scale (1 = Strongly Disagree, 5 = Strongly Agree).

# B Local Deployment Configuration

Table 9 reports hardware specifications and inference performance.

# C Prompt Templates

The following task-specific prompt templates were used for all generation tasks. Each template includes explicit grounding instructions and epistemic boundary constraints.

## C.1 Summary Generation Prompt

```
You are an educational summarization assistant. Your task is to produce a concise summary of the provided educational text.
```

**Instructions:**

- Condense the text to approximately 30% of its original length.
- Preserve all key concepts, technical terminology, and hierarchical relationships.
- Maintain causal, comparative, and definitional links.
- Do not introduce information not present in the context.
- If the context lacks sufficient detail for a complete summary, indicate which aspects are omitted.

**Output format:** A single coherent paragraph with section headings where appropriate.

**Context:** {retrieved_chunks}

**Summary:**

## C.2 Question–Answer Generation Prompt

```
You are an educational assessment designer. Your task is to generate question-answer pairs from the provided educational text.
```

**Instructions:**

- Generate exactly {n} question–answer pairs.
- Vary cognitive levels across Bloom's taxonomy: remember, understand, apply, analyze.
- Each question must be unambiguously answerable from the provided context.
- Do not create tautological questions (e.g., "What is X?" followed by "X is X").
- Do not generate questions requiring external knowledge.
- Provide a concise, accurate answer for each question.

**Output format:**

```
Q1: [Question]
A1: [Answer]
...
Q{n}: [Question]
A{n}: [Answer]
```

**Context:** {retrieved_chunks}

## C.3 Flashcard Generation Prompt

```
You are a flashcard designer for spaced repetition study. Your task is to generate flashcards from the provided educational text.
```

**Instructions:**

- Generate flashcards targeting discrete, verifiable concepts.
- Each card must contain exactly one question and one answer.
- Maximum length: 128 tokens per card face (question or answer).
- Optimize for active recall: questions should prompt retrieval of a specific fact, definition, or relationship.
- Do not include information not present in the context.
- If a concept cannot be adequately captured in 128 tokens, split it across multiple cards.

**Output format:**

```
Q: [Question]
A: [Answer]
```

**Context:** {retrieved_chunks}

# D Evaluation Rubrics

Two domain experts independently rated generated artifacts using the rubrics below. Discrepancies were resolved through discussion.

## D.1 Summary Evaluation

The evaluation criteria and rating scales for summaries are listed in Table 10.

## D.2 Question–Answer Set Evaluation

The evaluation criteria and rating scales for question–answer sets are listed in Table 11.

## D.3 Flashcard Evaluation

The evaluation criteria and rating scales for flashcards are listed in Table 12.

Table 8: Questionnaire Structure: Theoretical Constructs, Domains, and Items

| **Theoretical Foundation** | **Domain** | **Item** |
|---|---|---|
| Constructivist Learning Theory | Perceived Usefulness | 1. The generated materials helped me identify key concepts I might have overlooked.<br>2. The system encouraged me to think more deeply about the source content.<br>3. I felt actively involved in constructing my understanding while using the materials.<br>4. The outputs served as a useful starting point for my own further exploration of the topic. |
| Human–AI Interaction Theory | Accuracy and Reliability | 5. The generated content was factually consistent with the uploaded documents.<br>6. I could trace most claims back to specific parts of the source material.<br>7. I felt confident that the outputs did not contain fabricated information.<br>8. The system provided sufficient transparency for me to judge the reliability of its outputs. |
| Cognitive Load Theory | Perceived Cognitive Support | 9. The summaries helped me grasp the main ideas without re-reading the entire document.<br>10. The structured format (summaries, questions, flashcards) reduced the mental effort needed to study the material.<br>11. The system helped me focus on important concepts rather than getting lost in details.<br>12. The generated questions prompted me to think about relationships between concepts rather than memorize isolated facts. |
| Human–AI Interaction Theory | Usability | 13. The interface was intuitive and easy to navigate.<br>14. The generation process was predictable and transparent.<br>15. I understood how to refine or regenerate outputs when needed.<br>16. The system responded in a timeframe that supported my workflow. |

Table 9: Local Deployment Configuration and Inference Performance

| **Category** | **Metric** | **Value** |
|---|---|---|
| Hardware | GPU | NVIDIA RTX 4090 (24 GB) |
| | System RAM | 64 GB |
| | CPU | AMD Ryzen 9 7950X |
| Model | Model | Llama 3.1 8B Instruct |
| | Quantization | 4-bit AWQ |
| Memory | Model memory | $\approx$4.7 GB VRAM |
| | Peak inference memory | 6.2 GB VRAM |
| Latency | Summaries | 2.3 s ($SD = 0.4$) |
| | Question–answer sets | 1.1 s ($SD = 0.2$) |
| | Flashcards | 0.8 s ($SD = 0.2$) |
| Throughput | Sequential load | 18 requests/min |

Table 10: Summary Evaluation Rubric

| **Criterion** | **Definition and Scale** |
|---|---|
| Factual Consistency | Proportion of claims fully supported by the source document. Rated 1–5: 1 = majority unsupported/contradicted; 3 = mixed support; 5 = all claims fully supported. |
| Completeness | Degree to which key concepts from the source are retained. Rated 1–5: 1 = majority omitted; 3 = most present with minor omissions; 5 = all critical concepts retained. |
| Relevance | Alignment between summary content and the inferred information need of a learner studying this material. Rated 1–5: 1 = minimal alignment; 3 = adequate; 5 = precise alignment. |
| Compression Ratio | Objective measure: summary word count divided by source word count. |

Table 11: Question–Answer Set Evaluation Rubric

| **Criterion** | **Definition and Scale** |
|---|---|
| Correctness | Factual accuracy of the provided answer. Rated 1–5: 1 = incorrect; 3 = partially correct; 5 = fully accurate. |
| Answerability | Whether the question can be answered unambiguously from the source context. Rated 1–5: 1 = unanswerable; 3 = answerable with inference; 5 = directly and unambiguously answerable. |
| Relevance | Alignment with the source topic and pedagogical utility. Rated 1–5. |
| Cognitive-Level Validity | Proportion of items matching their intended Bloom's taxonomy classification (remember, understand, apply, analyze). |

Table 12: Flashcard Evaluation Rubric

| **Criterion** | **Definition and Scale** |
|---|---|
| Factual Correctness | Accuracy of the answer component. Rated 1–5. |
| Pedagogical Usefulness | Suitability for spaced repetition and active recall. Rated 1–5: 1 = unsuitable; 3 = adequate; 5 = optimally structured. |
| Coverage | Proportion of key terms from the source represented across the full set of generated flashcards. |

# One Head, Three Transcripts: A Controlled Oracle-to-ASR Audit of Bangla Speech–Text Sentiment Analysis

**Md Raihan**[1] **Nilam Rajak**[2] **Kuntal Ghosh**[3] **Sathyanarayanan Dhorali**[1]
[1]Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany
[2]ERP Professionals, Germany [3]WallStreetDocs (WSD), United Kingdom
md.raihan@fau.de, nilam.rajak@erp-professionals.de
kuntal.ghosh@wsd.com, sathyanarayanan8998@gmail.com

## Abstract

Speech sentiment systems are routinely benchmarked with gold transcripts, but a deployed cascade must read whatever its recognizer produces. We audit that gap on BanglaMUSE, a Bangla read-speech sentiment corpus, using sentence-grouped cross-validation and a controlled substitution design in which one trained classifier is evaluated three times while only the test-time transcript changes. Replacing gold transcripts with ASR output costs 0.109 macro-F1 with Bengali XLS-R CTC and 0.341 with our Whisper large-v3 configuration, a high-error stress case. Retraining the classifier on matched recognizer output recovers 57–81% of the gap, but re-selecting only the decision threshold on ASR validation data, with weights and probabilities unchanged, already recovers 28–49%. Adding the speech embedding improves over ASR text, yet replacing it with acquisition/nuisance features gives a similar gain for gold-trained heads. These features alone predict sentiment at 0.669 macro-F1 (group-permutation null mean 0.495, $p = 0.001$), and a speech-only classifier that reaches 0.940 falls to 0.621 when both speaker and sentence are unseen. These speech-only and fusion results therefore do not provide clean evidence of prosodic sentiment cues. Low-resource speech–text benchmarks should report oracle and deployable conditions, separate threshold effects from retraining, and test acquisition variables before interpreting audio.

## 1 Introduction

Many speech sentiment and emotion systems evaluate the text branch on human transcripts. This is reasonable for studying fusion in isolation, but a deployed cascade consumes recognizer output. Prior work has shown that this mismatch can materially affect affective prediction on English benchmarks (Sheikh et al., 2018; Culnan et al., 2021; Li et al., 2024); controlled evidence remains scarce for low-resource languages, where ASR errors are larger and gold text is a less realistic deployment assumption.

Bangla makes a useful case study. It is spoken by more than 230 million people yet remains under-resourced in speech technology (Nayeem et al., 2026). Its writing system also makes normalization consequential for ASR scoring: visually coherent grapheme clusters can span multiple Unicode code points, so we use symmetric normalization and grapheme-cluster CER in addition to WER. We compare Whisper, a general multilingual recognizer (Radford et al., 2023), with a community Bengali CTC model fine-tuned from XLS-R-300M (Babu et al., 2022).

We audit the gap on BanglaMUSE (Nayeem et al., 2026), a corpus of 1,000 sentiment-labelled Bangla sentences each read by four speakers. Our design is deliberately narrow. We fix the encoders, freeze them, and use one lightweight classifier head, because the question is not which architecture wins but how much the evaluation protocol itself inflates reported performance. The central device is a controlled substitution: the *same trained head* is evaluated on gold and on automatic transcripts, so the measured difference is attributable to transcript source rather than to two separately fitted models.

Four questions organize the work. **RQ1:** how accurately do a general multilingual recognizer and a Bangla-specific one transcribe this corpus? **RQ2:** holding the classifier fixed, how much does text-only performance change when gold test transcripts are replaced by automatic ones? **RQ3:** does the same hold for speech–text fusion? **RQ4:** how much of any gap is recovered by training on matched recognizer output, and how much of that recovery needs only a different decision threshold? A fifth question turned out to matter as much: whether speech-input results can be interpreted cleanly as evidence of prosodic sentiment. They cannot, because acquisition conditions are strongly



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 40–54
November 9, 2026 ©2026 Association for Computational Linguistics

aligned with the labels.

Our contribution is an evaluation and audit study rather than a new method: (i) a sentence-disjoint evaluation protocol for BanglaMUSE with automated leakage assertions; (ii) a controlled oracle-to-ASR substitution audit across two Bangla-capable recognizers with sentence-clustered uncertainty; (iii) an operational adaptation analysis that separates decision-threshold adaptation from head retraining; and (iv) an acquisition audit—a nuisance-feature fusion control, a permutation null and a speaker-transfer diagnostic—showing that speech-input results on this corpus are confounded with recording conditions.

## 2 Related Work

Imperfect transcripts are known to matter for affective prediction. Sheikh et al. (2018) study sentiment analysis with imperfect ASR and acoustic views on CMU-MOSI/MOSEI. Culnan et al. (2021) compare gold and automatic transcripts across emotion, sarcasm and personality detection and report partial mitigation from audio together with a weak link between WER and downstream performance. Wu et al. (2022) propose multimodal refinement of sentiment-word representations and evaluate it on CMU-MOSI transcripts produced by three ASR systems. Li et al. (2024) benchmark eleven recognizers on IEMOCAP (Busso et al., 2008), CMU-MOSI (Zadeh et al., 2016) and MSP-Podcast (Lotfian and Busso, 2019) using text-only and six bimodal fusion techniques, and propose an ASR-error-robust framework. We therefore do not claim that oracle-to-ASR degradation, training on recognizer output, or holding a classifier fixed is new. We instead ask how large this gap is on a low-resource Bangla dataset, how much can be recovered, and which alternative explanations survive a controlled same-head analysis.

A related line trains downstream systems to tolerate recognition errors: simulated ASR errors can be injected when training spoken language understanding systems (Simonnet et al., 2018), and interventional speech-noise injection improves generalization across recognizers (Jung et al., 2024). Our matched-training conditions use real recognizer output instead and test only this simplest form of the idea.

Self-supervised speech representations (Baevski et al., 2020; Hsu et al., 2021) and their cross-lingual extensions (Babu et al., 2022) underpin most current low-resource recognizers, either fine-tuned with a CTC objective (Graves et al., 2006) or used as encoders. Large multilingual systems also broaden Bangla coverage: Whisper uses large-scale weak supervision (Radford et al., 2023), whereas MMS combines multilingual self-supervised pre-training with supervised ASR adaptation across more than 1,000 languages (Pratap et al., 2024). Bhogale et al. (2023) find that recognizer rankings on one Indic benchmark can correlate weakly with rankings on another, and published Bangla error rates differ in normalization, punctuation handling and language-model use, so we report our own operating points.

Models routinely exploit incidental dataset structure rather than the intended signal (Torralba and Efros, 2011; Geirhos et al., 2020), as documented for annotation artifacts in NLI (Gururangan et al., 2018) and spurious source tags in image classification (Lapuschkin et al., 2019). Our acquisition audit applies this concern to speech: we ask whether recording conditions, rather than content or delivery, carry the label.

## 3 The BanglaMUSE Corpus and Its Preparation

BanglaMUSE (Nayeem et al., 2026) contains 1,000 sentiment-labelled Bangla sentences, balanced 500 positive and 500 negative, each read by four speakers (two female, two male). We use it because every recording comes with its exact reference text and a sentence-level sentiment label and every sentence is read four times, which a transcript-substitution audit requires. Prominent public Bangla affective-speech alternatives, such as SUBESCO (Sultana et al., 2021) and BanglaSER (Das et al., 2022), annotate categorical emotions on acted readings of a small fixed set of sentences.

The delivered corpus required repair, which we performed by content-derived rules rather than by hand (Appendix A.1). The prepared corpus contains 3,996 usable recordings (1,997 positive, 1,999 negative; 4.08 hours) over 998 sentence groups; two sentence pairs with identical token multisets are merged into shared groups. Looser paraphrases are deliberately not merged: 60 sentence pairs have token Jaccard similarity of at least 0.7, and 13 of those carry opposite labels, so such overlap may inflate *or* deflate absolute scores; Section 6.2 shows the reported gaps are insensitive to these items. The recordings mix 16, 44.1 and 48 kHz, mono and

stereo, and two codecs. All audio is resampled to 16 kHz mono, but as Section 6.5 shows, acquisition traces survive.

## 4 Evaluation Protocol

All recordings of a sentence group share one outer fold. We use 5-fold stratified group cross-validation, grouped on sentence group and stratified on sentiment (about 200 groups and 800 recordings per test fold), with a deterministic group-disjoint 10% inner validation split for early stopping and model selection (Appendix A.2). An utterance-random split would test the text branch on sentences whose exact wording it had seen through another speaker. Because this failure is silent, 44 automated assertions confirm zero sentence-group and normalized-transcript overlap between every training, validation and test partition, and exactly one out-of-fold prediction per recording and condition.

A single frozen normalizer is applied identically to gold text and to both recognizers: Unicode NFC, removal of model-specific and zero-width characters, Bangla digit mapping, whitespace collapsing, and punctuation removal. Punctuation is removed because Whisper emits it and CTC models do not; retaining it would charge the CTC system for a stylistic difference rather than a recognition error. WER is computed over whitespace tokens and CER over extended grapheme clusters, since one visual unit of Bangla routinely spans several code points. A narrowly scoped post-hoc canonicalization check is reported in Appendix A.18.

Unless stated otherwise, scores are pooled recording-level out-of-fold macro-F1. Four speakers reading the same sentence are not independent observations, so the bootstrap resamples sentence *groups* with all their recordings (10,000 iterations), shares each resample across conditions, and reports paired 95% intervals on differences between conditions (Efron and Tibshirani, 1994; Dror et al., 2018).

The four substitution gaps, the two fusion-versus-ASR-text comparisons (C6−C3, C7−C4) and the four matched-training recovery comparisons were declared before the primary analysis; the high-similarity exclusion, X1–X2 and the C1-versus-fusion comparisons were post hoc. The nuisance-fusion control, threshold-only adaptation, lexical baseline, permutation null, seed variability, recovery intervals and speaker-transfer diagnostic were added later. Their settings were fixed before they were run, but after the primary results were known, so we label them *secondary* throughout.

## 5 Experimental Setup

**ASR-A** is the multilingual Whisper large-v3 model of Radford et al. (2023),[1] run through Hugging Face `transformers` 5.14.1 with beam 2, early stopping, at most 200 new tokens, Bengali and transcription forced, and sampling disabled. Every clip is shorter than 8.4 s, so Whisper's long-form options (conditioning on previous text and the compression-ratio, log-probability and no-speech thresholds) are inactive (Appendix A.16). Beam search was preferred over greedy decoding on a 300-utterance diagnostic sample drawn from this corpus, so this choice used evaluation references: greedy decoding emits romanized Latin-script output for 31% of that sample and reaches 0.744 WER, against 0.692 for beam 2 and 0.679 for beam 5. We retained beam 2 as the primary, compute-motivated operating point; beam 5 is reported only to show decoding sensitivity. Whisper's tokenizer needs a median of 72 new tokens for these sentences; a 48-token budget silently truncates 99.5% of them.

**ASR-B** is Bengali XLS-R CTC[2], a community Bengali fine-tune of XLS-R-300M (Babu et al., 2022) with a CTC head (Graves et al., 2006), decoded greedily and with no external language model; its WER is therefore not comparable to published LM-rescored numbers. Neither recognizer was fine-tuned on BanglaMUSE: each recording was transcribed once by the unadapted recognizers, and the transcripts are indexed by the downstream folds.

The text branch uses the frozen `csebuetnlp/banglabert` checkpoint (Bhattacharjee et al., 2022) with attention-mask-aware mean pooling over the final layer. The speech branch uses the frozen `facebook/wav2vec2-xls-r-300m` checkpoint (Babu et al., 2022) with mask-aware temporal mean and standard-deviation pooling. Embeddings are computed once and reused across every fold, seed and condition. ASR-B is a fine-tune of the same self-supervised backbone as the speech encoder; thus, in the Bengali XLS-R condition, the recognizer and speech encoder do not provide acoustic-backbone diversity.

[1] `openai/whisper-large-v3`, revision `06f233fe`.

[2] `arijitx/wav2vec2-xls-r-300m-bengali`, revision `45ed7c70`.

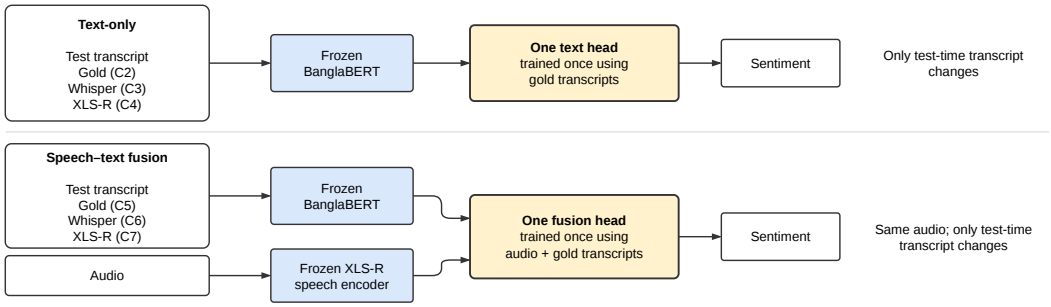



Figure 1: Controlled substitution design. C2–C4 reuse one text head trained on gold transcripts, and C5–C7 reuse one fusion head trained on audio and gold transcripts; only test-time transcripts change, with audio fixed for fusion. The secondary control N5–N7 follows the same design with acquisition/nuisance features in place of the speech encoder.

The classifier is a nonlinear head, LayerNorm $\rightarrow$ Linear(256) $\rightarrow$ GELU $\rightarrow$ Dropout(0.2) $\rightarrow$ Linear(2), trained with AdamW (Loshchilov and Hutter, 2019) (learning rate $10^{-3}$, weight decay $10^{-4}$, batch size 64, at most 60 epochs, early-stopping patience 8) on unweighted cross-entropy; the epoch with the highest validation macro-F1 is retained. Embeddings are L2-normalized per sample, and fusion concatenates the normalized speech and text embeddings. The primary heads fit no scaler or other data statistic; the secondary nuisance and lexical baselines fit their preprocessing on training partitions only. Results average the probabilities of three seeds (13, 42, 87) before applying the rule $p(\text{positive}) \geq 0.5$.

**Condition definitions:** C1 is an independently trained speech-only head. C2, C3 and C4 reuse one gold-trained text head evaluated on gold, Whisper and Bengali XLS-R transcripts; C5–C7 reuse one gold-trained fusion head in the same way (Figure 1). Training and evaluation occur within a single process, so head reuse is a property of the code rather than of a checkpoint round-trip. Gold-trained heads are early-stopped on gold validation transcripts. A1–A4 are text and fusion heads trained and early-stopped on the matching recognizer's transcripts; X1 and X2 reuse the A1 and A2 heads on the *other* recognizer's transcripts (post hoc; Appendix A.5). The secondary conditions are defined where they are used. We keep three comparison types apart: *controlled substitutions* (same head; the test transcript changes), *threshold-only adaptation* (same weights and probabilities; a validation-selected cutoff changes) and *system-level comparisons* (different fitted heads or inputs).

| System | WER | CER | Sub. | Del. | Ins. |
|---|---|---|---|---|---|
| Whisper | 0.687 | 0.360 | 0.590 | 0.012 | 0.086 |
| Bengali XLS-R | 0.276 | 0.111 | 0.230 | 0.028 | 0.017 |

Table 1: ASR quality on BanglaMUSE: Whisper large-v3 and Bengali XLS-R CTC. Neither recognizer was fine-tuned on the corpus. All rates are corpus-level (edits pooled over recordings).

## 6 Results

### 6.1 Recognition quality

Table 1 reports RQ1. Whisper large-v3 has a WER of 0.687, 2.5$\times$ the 0.276 WER of Bengali XLS-R CTC; CER gives the same ranking (0.360 versus 0.111). Whisper's insertion rate is about 5.1$\times$ that of the CTC system (0.086 versus 0.017), but repetition loops do not explain it: only 2 of 3,996 Whisper hypotheses show an immediate $n$-gram loop or an excessive compression ratio, and they contain none of the insertions. Romanized output matters more: 9.8% of Whisper hypotheses contain Latin script, including 11 of the 20 with the highest WER (Appendix A.16). We treat the two systems as tested operating points, not as representative of Bangla ASR.

### 6.2 Controlled oracle-to-ASR substitution

Table 2 and Figure 2 give the principal result, addressing RQ2 and RQ3. Within each column, the same trained head is reused, so only the supplied test-time transcript source changes. For text alone, gold-transcript evaluation overstates performance under ASR input by 0.109 macro-F1 with Bengali XLS-R and 0.341 with Whisper; the corresponding fusion gaps are 0.086 and 0.304. Every interval excludes zero, and every gap keeps its sign and approximate size in each of the three training

| | Text only | | Text + nuisance (secondary) | | Text + speech | |
|---|---|---|---|---|---|---|
| Test transcript | Macro-F1 | Gap [95% CI] | Macro-F1 | Gap [95% CI] | Macro-F1 | Gap [95% CI] |
| Gold | C2 0.976 | – | N5 0.980 | – | C5 0.981 | – |
| Whisper | C3 0.635 | 0.341 [0.316, 0.365] | N6 0.692 | 0.289 [0.266, 0.311] | C6 0.677 | 0.304 [0.282, 0.328] |
| Bengali XLS-R | C4 0.867 | 0.109 [0.095, 0.124] | N7 0.890 | 0.090 [0.078, 0.103] | C7 0.895 | 0.086 [0.074, 0.099] |

Table 2: Principal conditions and acquisition/nuisance control. Within each column, the same gold-trained head is reused and only the test transcript changes. Gaps are paired sentence-clustered bootstrap differences from the gold condition. Full scores are in Appendix Table 6.

seeds (C2−C3 ranges from 0.311 to 0.353; Appendix A.10). Excluding from evaluation every sentence with token Jaccard similarity of at least 0.7 to another sentence (84 sentences, 333 recordings) moves the four gaps by at most 0.004, with every interval still excluding zero (Appendix Table 7); this check changes the scored items, not the training data.

Adding the speech embedding improves macro-F1 over ASR text by 0.042 [0.030, 0.053] under Whisper (C6−C3) and by 0.028 [0.020, 0.035] under Bengali XLS-R (C7−C4). Because text-only and fusion use different trained heads, these are system-level comparisons. As a secondary control, we replaced the speech embedding with the 17-dimensional acquisition/nuisance feature vector of Section 6.5, whose imputation, encoding and standardization are fitted on training partitions only (N5–N7; Appendix A.6). This text+nuisance head improves over ASR text by 0.057 [0.045, 0.068] and 0.023 [0.016, 0.030], and its direct differences from speech fusion are small (C6−N6 = −0.015 [−0.025, −0.004]; C7−N7 = 0.005 [−0.001, 0.011]) and change sign across training seeds. The measured acquisition/nuisance features produce a fusion gain of similar size. We therefore cannot attribute the speech-fusion gain specifically to prosody or other sentiment-bearing acoustics. This control also does not tell us which cues the speech encoder actually uses, and the acquisition/nuisance feature vector is not guaranteed to be content-free: its trailing 0.15 s window lies within 20 dB of the utterance's level in 47% of recordings.

The oracle score itself is near ceiling. A secondary TF-IDF (word uni- and bigrams over whitespace tokens) and logistic-regression baseline on gold text (Appendix A.8), fitted on training partitions only under the same folds, reaches 0.854, which is 0.122 [0.101, 0.145] below the frozen-encoder head (C2): simple lexical features separate the classes well under this protocol, though less well than the pretrained encoder. This does not rule out lexical shortcuts, and the high-similarity check above is not a family-disjoint retraining.

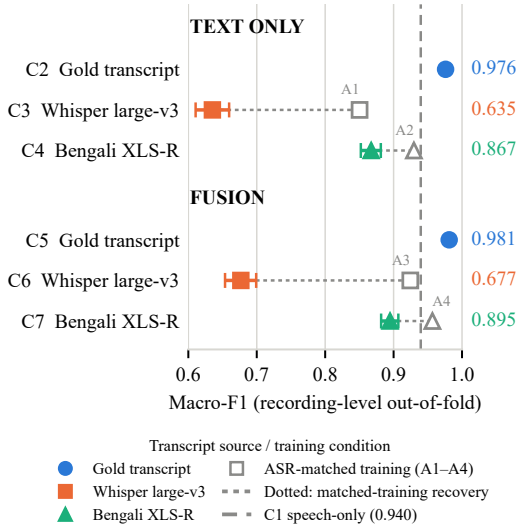



Figure 2: Macro-F1 for C2–C7. Error bars show 95% paired sentence-clustered bootstrap intervals for oracle-to-ASR gaps, mapped to scores relative to C2 or C5. Hollow markers show ASR-matched heads (A1–A4); dotted segments show recovery. The dashed line is the independently trained speech-only head C1.

### 6.3 Degradation tracks transcription quality

Is the degradation simply item difficulty? Across bins fixed before downstream performance was inspected, substituted-text accuracy for Bengali XLS-R falls from 0.971 at zero WER to 0.738 in the highest-WER bin, while oracle-text accuracy on the *same items* stays between 0.971 and 0.982 and speech-only accuracy between 0.930 and 0.946 (Table 3); Whisper shows the same pattern across its positive-WER bins. Normalized transcripts are exact for 801 of 3,996 XLS-R but only 10 Whisper recordings, so the table omits Whisper's zero-WER bin; its 10 items remain in every other analysis. This is an observed association between transcription quality and downstream accuracy, not a causal dose-response: the near-ceiling oracle and the con-

| ASR | Bin | $n$ | WER | Oracle | ASR text | ASR fusion | Speech |
|---|---|---|---|---|---|---|---|
| Whisper | low | 1392 | 0.424 | 0.971 | 0.747 | 0.787 | 0.940 |
| Whisper | mid | 1642 | 0.732 | 0.975 | 0.645 | 0.678 | 0.938 |
| Whisper | high | 952 | 1.044 | 0.985 | 0.601 | 0.623 | 0.942 |
| XLS-R | WER = 0 | 801 | 0.000 | 0.971 | 0.971 | 0.981 | 0.945 |
| XLS-R | low | 1152 | 0.167 | 0.982 | 0.909 | 0.927 | 0.930 |
| XLS-R | mid | 1092 | 0.333 | 0.978 | 0.865 | 0.891 | 0.940 |
| XLS-R | high | 951 | 0.579 | 0.971 | 0.738 | 0.791 | 0.946 |

Table 3: Accuracy by transcription-quality bin. Bins are zero WER followed by tertiles of positive WER; Whisper's zero-WER bin ($n = 10$) is omitted. WER is the unweighted mean of per-recording WERs; Table 1 reports corpus-level WER. WER can exceed 1 when insertions make the total edit count larger than the number of reference words.

founded speech-only condition are imperfect controls for item difficulty. Fusion beats text alone in every bin shown (descriptive error patterns: Appendix A.15).

### 6.4 How much of the gap is recovered, and how

Retraining the head on matched ASR transcripts (A1–A4) recovers 57–81% of the gap (Table 4). The gold-trained heads, however, also shift their decisions under ASR input: the predicted-positive rate falls from 50.0% on gold text to 18.7% on Whisper and 38.9% on XLS-R text against a true rate of 50%, and C3's recall on positive recordings is only 0.358. We therefore asked, in a secondary analysis, how much of the recovery needs only a new threshold. For each fold we re-selected the decision threshold of the unchanged gold-trained head on its inner-validation ASR transcripts (seed-averaged probabilities; grid 0.000–1.000; ties broken towards 0.5) and applied it to the test fold; weights, probabilities and test labels are untouched (Appendix A.7).

Threshold-only adaptation restores balanced predictions and recovers 28–49% of the gap, for example 0.149 [0.124, 0.174] macro-F1 for Whisper text (T3−C3) with thresholds between 0.010 and 0.099. A threshold selected on *gold* validation data gains at most 0.008, so the effect comes from adapting the cutoff to ASR input rather than from optimizing a cutoff as such. Head retraining still adds 0.032 to 0.097 over threshold-only adaptation in every setting, and 0.031 to 0.097 when the matched heads also receive their own tuned thresholds, which barely change them; the retraining gain therefore does not come from the cutoff alone. All conditions use the same test labels, so the change is not a change in the label prior. Threshold adaptation also leaves the predicted probabilities unchanged, so we do not interpret it as calibration. With the encoders frozen, the remaining gain comes from retraining the classifier head. For each setting, let $O$ be the oracle score, $B$ the gold-trained head's score on ASR input with the default threshold, $T$ its score after ASR-validation threshold adaptation, and $A$ the ASR-matched head's score with the default threshold. We use the decomposition $O-B = (T-B)+(A-T)+(O-A)$ only as an accounting device; it should not be interpreted as a causal separation of mismatch and information loss.

Recovery is larger for fusion (72–81%) than for text (57–63%). A secondary matched text+nuisance head (NA3, NA4) also improves over matched text (by 0.042 and 0.012), but less than matched speech fusion (A3−NA3 = 0.032 [0.023, 0.040]; A4−NA4 = 0.016 [0.009, 0.022]; positive in every seed). This control therefore does not reproduce the whole matched-fusion gain; unmeasured acquisition effects and lexical information in the speech signal remain possible explanations, and the larger fusion recovery is not evidence of complementary sentiment acoustics. Finally, recovery partly transfers across recognizers: a head trained on XLS-R transcripts raises Whisper-test macro-F1 from 0.635 to 0.815, against 0.850 under matched training (post hoc; Appendix A.5).

### 6.5 Acquisition audit

Speech-only reaches 0.940 macro-F1, which invites a prosodic reading—that expressive delivery carries the sentiment. That reading is not available on this corpus. The corpus is label-blocked: sentences 1–500 are positive and 501–1000 negative. Using the filesystem creation timestamps retained in the delivered files as a proxy for acquisition order, speakers m1, f2, m2 and f1 show only 1, 2, 9 and 11 label switches, respectively, versus roughly 500 per speaker under random interleaving (Appendix Table 15). These timestamps are not authoritative acquisition logs, but their pattern is consistent with class-blocked recording.

A logistic regression given **only** the acquisition and nuisance features listed in Table 5 reaches 0.669 macro-F1. In a secondary whole-pipeline permutation test that shuffles labels between sentence groups and regenerates the folds (1,000 permutations; Appendix A.9), the null scores average 0.495 (95% of them between 0.476 and 0.512) and none reaches the observed score

| | | Gold-trained head | | | Matched head | | Share of gap recovered (%) | |
|---|---|---|---|---|---|---|---|---|
| Setting | Oracle | Default | ASR thr. | Gold thr. | Default | ASR thr. | Matched head | Threshold only |
| Whisper text | 0.976 | 0.635 (18.7) | 0.784 (51.9) | 0.635 | 0.850 | 0.845 | 63.1 [58.6, 67.6] | 43.6 [37.9, 48.9] |
| XLS-R text | 0.976 | 0.867 (38.9) | 0.898 (47.7) | 0.870 | 0.930 | 0.929 | 57.4 [47.2, 67.4] | 28.2 [18.1, 37.8] |
| Whisper fusion | 0.981 | 0.677 (21.1) | 0.827 (48.3) | 0.685 | 0.924 | 0.924 | 81.3 [77.8, 85.0] | 49.4 [44.4, 54.3] |
| XLS-R fusion | 0.981 | 0.895 (41.0) | 0.921 (47.8) | 0.897 | 0.957 | 0.952 | 71.6 [62.7, 80.5] | 30.5 [20.7, 39.7] |

Table 4: Decision rule versus retraining (RQ4). Macro-F1 for gold-trained heads and ASR-matched heads (A1–A4), with default or per-fold validation-selected thresholds. Parentheses: predicted-positive rate (%). Recovery shares $(A-B)/(O-B)$ (retraining) and $(T-B)/(O-B)$ (threshold only) are computed within each bootstrap resample; brackets show 95% intervals.

| Model | Shared speakers | Held-out speaker and sentence |
|---|---|---|
| Speech-only head (C1) | 0.940 | 0.621 |
| Acquisition + nuisance LR | **0.669** | 0.333 |
| metadata only | 0.521 | – |
| extended exploratory | 0.699 | – |

Table 5: Acquisition audit (pooled out-of-fold macro-F1). The primary logistic regression (LR) combines metadata with encoding- and signal-level features (Appendix A.6); the exploratory variant adds duration, RMS, and spectral centroid. Group-permutation null for the primary LR: mean 0.495, $p = 0.001$.

($p = 0.001$, the smallest attainable). This supports predictability relative to that null only; it neither removes session effects nor licenses inference to independently recorded sessions. Metadata alone reaches only 0.521, whereas adding the encoding- and signal-level features raises performance to 0.669. This is the speech analogue of the artifact literature discussed in Section 2.

Our folds share speakers between training and test. In a secondary joint speaker-and-sentence holdout, each recording is predicted by models trained on the other three speakers and on other sentence groups (20 partitions; Appendix A.11). The speech-only head then falls from 0.940 to 0.621 (0.555–0.686 per speaker), and the acquisition/nuisance classifier from 0.669 to 0.333: the feature–label mapping learned from three speakers inverts or collapses on the fourth. The high speech-only score therefore does not carry over to an unseen speaker reading unseen sentences under this protocol, which also trains on fewer speakers. With four speakers this is descriptive, as are the per-speaker results in Appendix A.13, and neither drop by itself identifies an acquisition mechanism.

We report this as an evaluation lesson rather than a criticism of the resource: sequential collection is practical, but class-blocked collection can make nuisance conditions label-predictive. The caveat applies to every speech-input condition, including fusion (Section 6.2), and reaches the ASR-text conditions, because transcripts are produced from the audio; Appendix Table 12 shows no aggregate class-level WER pattern that explains the class asymmetry, but finer-grained effects remain possible. The controlled substitutions avoid a refitting confound; the adaptation and cross-recognizer comparisons are system-level evidence.

## 7 Discussion

The recognizer ranking by WER is preserved downstream, but not proportionally: WER differs by 2.5× whereas the text-only deployment gap differs by 3.1×, and within each recognizer accuracy declines as WER rises, which supports reporting downstream impact alongside WER (Culnan et al., 2021; Bhogale et al., 2023). Threshold adjustment alone recovers a substantial part of the improvement obtained with matched-head retraining. For this reason, we recommend reporting a threshold-only baseline alongside retraining results. For low-resource benchmark design, we recommend reporting oracle and deployable transcript conditions with controlled substitution rather than separately fitted per-condition models, and testing acquisition predictability, nuisance-feature fusion and speaker transfer before interpreting audio. When collection is sequential, interleaving classes or releasing session structure lets such effects be audited.

## 8 Conclusion

On BanglaMUSE, the oracle-to-ASR macro-F1 gap is 0.109 for Bengali XLS-R and 0.341 for the high-error Whisper configuration. Matched retraining recovers 57–81% of the gap, re-selecting only the decision threshold already recovers 28–49%, and a residual oracle advantage remains. Speech fusion performs better than ASR text alone, but

a control using only acquisition/nuisance features also improves performance. This matters because the corpus is label-blocked, and its filesystem timestamps are consistent with class-blocked recording. The nuisance features predict sentiment well above the permutation null, while the speech-only score falls from 0.940 to 0.621 when both speaker and sentence are unseen. Together, these results make it difficult to interpret the speech-input performance as evidence of prosodic sentiment cues. Low-resource speech–text evaluations should report oracle and deployable conditions, separate threshold effects from retraining, and audit acquisition variables before interpreting audio.

## Limitations

The corpus has four speakers, uses read rather than spontaneous speech, and carries binary sentence-level labels. All conclusions apply to BanglaMUSE, these two recognizer configurations, these models and this protocol, and should not be read as claims about Bangla speech sentiment in general. The main protocol shares speakers between training and test folds, and the speaker-and-sentence holdout has only four speakers.

Every speech-input condition is confounded with acquisition conditions, and ASR output inherits recording conditions from the audio. The measured acquisition/nuisance feature vector is incomplete and not guaranteed to be content-free, so the nuisance-fusion control tests one acquisition-based explanation rather than identifying what the speech encoder uses. The permutation null does not provide inference to independently recorded sessions, and the timestamp reconstruction is a proxy rather than a verified acquisition log.

A changed threshold identifies neither label shift nor calibration, and it is selected on the validation split already used for early stopping; the split into threshold, retraining and residual parts is operational, not causal. Bootstrap intervals are conditional on the fitted heads, splits and thresholds and exclude training-seed variation, which we report separately. Neither the lexical baseline nor the high-similarity check establishes independence from paraphrase families. Neither recognizer is at its ceiling (XLS-R runs without a language model, Whisper uses beam 2, and decoding was chosen on a sample of this corpus). The speech encoder shares a backbone with ASR-B, punctuation removal discards a potentially informative cue from all conditions, and the secondary analyses were specified after the primary results were known.

## Ethical Considerations

We use the public BanglaMUSE data release, licensed CC BY 4.0 (Nayeem et al., 2025), and collect no new personal data. Speakers are identified only by pseudonymous codes and we make no attempt at identification or at inferring speaker attributes. Results describe only the four speakers in this corpus; they must not be extrapolated to dialectal, spontaneous, conversational or code-switched Bangla, nor used to support claims about groups of speakers, and the per-speaker results are descriptive only, since speaker is confounded with recording condition. The acquisition audit reports a property of the released resource, not a judgement of the people who built it: class-blocked sequential collection is a practical choice for low-resource corpus building, and our recommendation is a methodological check, not a standard that existing resources have failed.

All experiments ran on a single consumer GPU; the dominant costs were approximately 1.3 hours of Whisper inference and 13 minutes for Bengali XLS-R. Code, the split definitions with their leakage assertions, per-recording out-of-fold predictions and the secondary-analysis scripts are released under open licenses (MIT for code, CC BY 4.0 for data) at `https://github.com/raihansaa/one-head-three-transcripts`.

## Acknowledgments

Generative AI tools were used to assist with drafting and language refinement throughout the manuscript, as well as structural feedback, literature search, suggestions for candidate analyses, and implementation of analysis code. The authors reviewed and revised all AI-assisted material, independently verified cited sources and reported results, and take full responsibility for the work.

## References

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2022. XLS-R: Self-supervised cross-lingual speech representation learning at scale. In *Interspeech 2022*, pages 2278–2282.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

Kaushal Bhogale, Sai Sundaresan, Abhigyan Raman, Tahir Javed, Mitesh M. Khapra, and Pratyush Kumar. 2023. Vistaar: Diverse benchmarks and training sets for Indian language ASR. In *Interspeech 2023*, pages 4384–4388.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359.

John Culnan, Seongjin Park, Meghavarshini Krishnaswamy, and Rebecca Sharp. 2021. Me, myself, and ire: Effects of automatic transcription quality on emotion, sarcasm, and personality detection. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 250–256, Online. Association for Computational Linguistics.

Rakesh Kumar Das, Nahidul Islam, Md. Rayhan Ahmed, Salekul Islam, Swakkhar Shatabda, and A. K. M. Muzahidul Islam. 2022. BanglaSER: A speech emotion recognition dataset for the Bangla language. *Data in Brief*, 42:108091.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Bradley Efron and Robert J. Tibshirani. 1994. *An Introduction to the Bootstrap*. Chapman & Hall.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 369–376.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

YeonJoon Jung, Jaeseong Lee, Seungtaek Choi, Dohyeon Lee, Minsoo Kim, and Seung-won Hwang. 2024. Interventional speech noise injection for ASR generalizable spoken language understanding. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20642–20655, Miami, Florida, USA. Association for Computational Linguistics.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2019. Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications*, 10(1):1096.

Yuanchao Li, Peter Bell, and Catherine Lai. 2024. Speech emotion recognition with ASR transcripts: A comprehensive study on word error rate and fusion techniques. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 518–525. IEEE.

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

Reza Lotfian and Carlos Busso. 2019. Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings. *IEEE Transactions on Affective Computing*, 10(4):471–483.

Md Darun Nayeem, Zarin Rafa, Tasnuva Tasnim Nova, Yasin Rahman, Abdul Mumeet Pathan, and Md Masudul Islam. 2026. BanglaMUSE: A multimodal Bangla sentiment dataset of text–audio pairs for speech and sentiment analysis. *Data in Brief*, 65:112458.

Md Darun Nayeem, Zarin Rafa, Tasnuva Tasnim Nova, Yasin Rahman, Abdul Mumeet Pathan, and Nusrat Sultana. 2025. A multimodal Bangla text–audio dataset for sentiment analysis. Mendeley Data. Version 1; CC BY 4.0.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, Alexei Baevski, Yossi Adi, Xiaohui Zhang, Wei-Ning

Hsu, Alexis Conneau, and Michael Auli. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR.

Imran Sheikh, Sri Harsha Dumpala, Rupayan Chakraborty, and Sunil Kumar Kopparapu. 2018. Sentiment analysis using imperfect views from spoken language and acoustic modalities. In *Proceedings of Grand Challenge and Workshop on Human Multimodal Language (Challenge-HML)*, pages 35–39, Melbourne, Australia. Association for Computational Linguistics.

Edwin Simonnet, Sahar Ghannay, Nathalie Camelin, and Yannick Estève. 2018. Simulating ASR errors for training SLU systems. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Sadia Sultana, M. Shahidur Rahman, M. Reza Selim, and M. Zafar Iqbal. 2021. SUST Bangla emotional speech corpus (SUBESCO): An audio-only emotional speech corpus for Bangla. *PLOS ONE*, 16(4):e0250173.

Antonio Torralba and Alexei A. Efros. 2011. Unbiased look at dataset bias. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1521–1528.

Yang Wu, Yanyan Zhao, Hao Yang, Song Chen, Bing Qin, Xiaohuan Cao, and Wenting Zhao. 2022. Sentiment word aware multimodal refinement for multimodal sentiment analysis with ASR errors. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1397–1406, Dublin, Ireland. Association for Computational Linguistics.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intelligent Systems*, 31(6):82–88.

# A Supporting Analyses

## A.1 Corpus preparation

The delivered archive contained 4,735 audio files. Of these, 737 were byte-identical extra copies of base recordings and were dropped after verifying that their hashes matched. Identifiers appeared in three inconsistent forms—zero-padded, unpadded, and one file with a five-digit identifier—and were parsed as integers and re-padded, which collapsed the variants automatically. 244 recordings were delivered as 16 kHz Ogg/Opus audio, 242 of them carrying `.mp3` filenames; we decode by content rather than by extension. The same audio payload appears under two different sentence identifiers whose texts differ by one word, making the reference assignment ambiguous; both recordings were excluded rather than forcing an arbitrary choice. Two further nominal recordings are absent from the delivery. Thus the 4,000 nominal recording slots comprise 3,996 usable recordings, the two ambiguous recordings and two missing recordings. Two sentence pairs have identical token multisets and differ only in word order; placing such near-identical lexical content in different folds would create an unusually strong overlap shortcut, so each pair is merged into a shared group by this objective rule, declared in advance. The 998 groups result from these merges; recording exclusions do not determine the group count. No label conflicts were found across the four recordings of any sentence.

## A.2 Reproducibility

Outer folds come from `StratifiedKFold` (5 folds, shuffled, seed 42) over the 998 sentence groups, stratified on the label; each outer-training partition holds out a label-stratified 10% of its groups for inner validation (seed 1000 + fold). The heads use seeds 13, 42 and 87 and the bootstrap seed 20260807. The neural-head architecture and optimizer settings in Section 5 were fixed before the primary experiments and were not tuned; validation macro-F1 was used only to select the checkpoint epoch. Gold-trained heads are early-stopped on gold validation transcripts, matched heads on the matching recognizer's validation transcripts, and the threshold-only conditions select their cutoff on the gold-trained head's seed-averaged probabilities for the ASR validation transcripts. Software: Python 3.12, PyTorch 2.11 (CUDA 12.8), `transformers` 5.14.1 and scikit-learn 1.9.0 on one consumer GPU (RTX 5060 Laptop). Checkpoint revisions: whisper-large-v3 `06f233fe`, wav2vec2-xls-r-300m-bengali `45ed7c70`, banglabert `9ce791f3`, wav2vec2-xls-r-300m `1a640f32`. On this machine, re-training every primary head reproduces all 155,844 stored test probabilities bit for bit. The code release includes the secondary-analysis specification, fixed before those analyses were run, and a manifest of all settings and result hashes.

### A.3 All conditions

Table 6 lists every condition with its accuracy, class-wise F1 and predicted-positive rate.

### A.4 High-similarity exclusion

Table 7 reports the high-similarity exclusion check discussed in Section 6.2: the four controlled gaps are essentially unchanged and every paired interval still excludes zero.

### A.5 ASR-trained and cross-recognizer conditions

Table 8 reports the ASR-matched conditions A1–A4 and the post-hoc cross-recognizer conditions X1–X2. Training on Bengali XLS-R transcripts raises Whisper-test macro-F1 from 0.635 to 0.815 (X2−C3 = 0.180 [0.152, 0.207]) against 0.850 under matched training; training on Whisper transcripts raises Bengali XLS-R-test performance from 0.867 to 0.909 (X1−C4 = 0.042 [0.026, 0.058]) against 0.930. Matching still adds a significant increment (A1−X2 = 0.035 [0.020, 0.051]; A2−X1 = 0.021 [0.011, 0.031]), but a smaller one. Substantial transfer shows that the adaptation benefit is not wholly recognizer-specific for these two systems. These are comparisons between separately trained heads, not controlled substitutions, and two recognizers are not enough to infer transfer to arbitrary ASR systems.

### A.6 Nuisance-fusion control

The acquisition/nuisance feature vector is the primary Table 5 feature set: eight numeric features (bytes per second, bitrate, clipping fraction, DC offset, and the RMS and 10th-percentile amplitude of the leading and trailing 0.15 s) and one-hot codec, sample rate, channel count and container format, 17 dimensions in total. Median imputation, one-hot categories and standardization are fitted on each fold's training partition and frozen for validation, test and every transcript substitution; the standardized vector is then L2-normalized per row and concatenated with the text embedding in place of the speech embedding. No fold had missing values, zero-variance training columns, unseen categories or zero-norm rows. The resulting head has 785 inputs and 203,300 parameters, against 2,816 inputs and 727,298 parameters for speech fusion, so the two are not architecture-identical. As a pipeline check, the Table 5 logistic regression on this normalized vector reaches 0.653, close to its original 0.669. The edge windows are not guaranteed to be silent: the leading window lies a median 69 dB below the utterance's RMS, but the trailing window lies within 20 dB of it for 47% of recordings and within 10 dB for 22%. Per seed, C6−N6 is −0.009, 0.009 and −0.036 and C7−N7 is −0.002, 0.019 and −0.004, whereas A3−NA3 (0.033, 0.034, 0.040) and A4−NA4 (0.014, 0.014, 0.012) are positive in every seed.

### A.7 Threshold-only adaptation

For each outer fold, the three seeds' inner-validation probabilities are averaged and the threshold $t$ on the grid $\{0.000, 0.001, \ldots, 1.000\}$ that maximizes validation macro-F1 under the rule $p \geq t$ is chosen, breaking ties by the smallest $|t - 0.5|$ and then the smaller $t$. The threshold is applied to that fold's seed-averaged test probabilities, which are exactly those of the corresponding default condition. The selected thresholds range from 0.010 to 0.099 for T3, 0.119 to 0.268 for T4, 0.002 to 0.042 for T6 and 0.039 to 0.177 for T7; the gold-validation thresholds range from 0.343 to 0.623 and the matched-head thresholds from 0.177 to 0.636. Table 9 gives every term of the operational decomposition with paired intervals.

### A.8 Lexical baseline

The secondary lexical baseline uses a TF-IDF vectorizer with whitespace tokens, no lowercasing, word uni- and bigrams, min_df = 1, smoothed IDF and L2 normalization, fitted on the training partition only (about 4,200 features per fold). As for C2, training and validation rows are one per sentence group and test rows are recordings. Logistic regression (lbfgs) selects $C \in \{0.1, 1, 10\}$ on inner-validation macro-F1 (ties to the smaller $C$); the selected values were 1, 10, 0.1, 10 and 10, and every fit converged. The baseline reaches 0.854 macro-F1 ($F1_-$ 0.866, $F1_+$ 0.841).

### A.9 Permutation null

Each of 1,000 permutations shuffles the 998 sentence-group labels between groups, broadcasts them to the groups' recordings, regenerates the outer folds with the original stratified splitter and seed, and refits the unchanged Table 5 pipeline; the statistic is pooled out-of-fold macro-F1. Scikit-learn's `permutation_test_score` with sentence groups would shuffle labels only within groups, which randomizes nothing here. The same procedure on the true labels reproduces the original folds and gives 0.669. The null has mean 0.495,

| ID | Condition | Macro-F1 | Acc. | $F1_{-}$ | $F1_{+}$ | Pred. pos. (%) |
|---|---|---|---|---|---|---|
| C1 | Speech only | 0.940 | 0.940 | 0.940 | 0.940 | 50.2 |
| C2 | Gold text | 0.976 | 0.976 | 0.976 | 0.976 | 50.0 |
| C3 | Gold-trained text, Whisper test | 0.635 | 0.671 | 0.749 | 0.521 | 18.7 |
| C4 | Gold-trained text, XLS-R test | 0.867 | 0.869 | 0.882 | 0.852 | 38.9 |
| C5 | Gold fusion | 0.981 | 0.981 | 0.981 | 0.981 | 49.7 |
| C6 | Gold-trained fusion, Whisper test | 0.677 | 0.704 | 0.770 | 0.583 | 21.1 |
| C7 | Gold-trained fusion, XLS-R test | 0.895 | 0.896 | 0.904 | 0.885 | 41.0 |
| A1 | Whisper-trained text | 0.850 | 0.850 | 0.853 | 0.847 | 48.0 |
| A2 | XLS-R-trained text | 0.930 | 0.930 | 0.929 | 0.930 | 50.3 |
| A3 | Whisper-trained fusion | 0.924 | 0.924 | 0.925 | 0.924 | 49.7 |
| A4 | XLS-R-trained fusion | 0.957 | 0.957 | 0.957 | 0.957 | 49.9 |
| X1 | Whisper-trained text, XLS-R test (post hoc) | 0.909 | 0.909 | 0.906 | 0.911 | 52.9 |
| X2 | XLS-R-trained text, Whisper test (post hoc) | 0.815 | 0.816 | 0.801 | 0.828 | 57.3 |
| N5 | Gold-trained text+nuisance, gold test | 0.980 | 0.980 | 0.980 | 0.980 | 50.1 |
| N6 | Gold-trained text+nuisance, Whisper test | 0.692 | 0.714 | 0.774 | 0.609 | 23.3 |
| N7 | Gold-trained text+nuisance, XLS-R test | 0.890 | 0.891 | 0.900 | 0.880 | 40.8 |
| NA3 | Whisper-trained text+nuisance | 0.893 | 0.893 | 0.892 | 0.893 | 50.3 |
| NA4 | XLS-R-trained text+nuisance | 0.941 | 0.941 | 0.941 | 0.942 | 50.7 |
| T3 | C3 head, ASR-validation threshold | 0.784 | 0.784 | 0.780 | 0.788 | 51.9 |
| T4 | C4 head, ASR-validation threshold | 0.898 | 0.898 | 0.900 | 0.895 | 47.7 |
| T6 | C6 head, ASR-validation threshold | 0.827 | 0.827 | 0.830 | 0.824 | 48.3 |
| T7 | C7 head, ASR-validation threshold | 0.921 | 0.921 | 0.923 | 0.919 | 47.8 |
| | C3/C4/C6/C7 heads, gold-validation threshold | 0.635 / 0.870 / 0.685 / 0.897 | | | | |
| | A1–A4 heads, ASR-validation threshold | 0.845 / 0.929 / 0.924 / 0.952 | | | | |
| | TF-IDF + logistic regression, gold text | 0.854 | 0.855 | 0.866 | 0.841 | 41.5 |

Table 6: All conditions (pooled recording-level out-of-fold scores). N, NA, T and the lexical baseline are secondary. Predicted-positive rates are in %; the true rate is 50.0%.

standard deviation 0.009 and maximum 0.524, so $p = (1 + 0)/(1{,}000 + 1) = 0.001$.

### A.10 Seed variability

Table 10 scores each training seed separately, pooled over the five outer folds, next to the probability-averaged ensemble used elsewhere. The three seeds are not independent replications of the test set, and the bootstrap intervals in the paper do not include this variation.

### A.11 Speaker-transfer diagnostic

For each speaker $s$ and outer fold $f$, the test set is speaker $s$'s recordings of fold $f$'s test groups; training and inner validation use the other three speakers' recordings of fold $f$'s training and inner-validation groups. The speech head uses the C1 recipe and seeds; the logistic regression, which has no validation step, is fitted on the other speakers' recordings of all of fold $f$'s training groups. Every recording receives exactly one prediction (Table 11). Under this holdout the logistic regression predicts positive for 1.0% of f1's positive and 50.0% of its negative recordings, and for 23.7% and 79.0% of m2's, while it labels almost all of f2's and most of m1's recordings positive.

### A.12 ASR quality by sentiment class

Table 12 reports ASR quality by sentiment class; Appendix A.14 uses it to assess the per-class asymmetry of the Whisper conditions.

### A.13 Per-speaker descriptive results

Table 13 lists per-speaker recognition and downstream results. Speaker is confounded with recording condition in this corpus, so these numbers are descriptive only.

### A.14 Post-hoc system-level observations

Both substituted fusion systems score below the speech-only system (C1−C6 = 0.263 [0.240, 0.286]; C1−C7 = 0.045 [0.031, 0.059]). These comparisons were not predeclared and, unlike C5→C6, contrast two *separately trained* systems, so they answer a deployment question—is this fusion system better than simply using speech?—rather than a controlled substitution question. Given Section 6.5, C1 is additionally affected by acquisition confounding, so we report the observation as directional and build no claim on it. There is no numerical contradiction with C6 > C3: the fusion system scores higher than the text-only system while still scoring below C1. As Section 6.2

| Comparison | Full-set gap [95% CI] | Excluded-set gap [95% CI] | Change |
|---|---|---|---|
| C2−C3 | 0.341 [0.316, 0.365] | 0.344 [0.318, 0.370] | +0.004 |
| C2−C4 | 0.109 [0.095, 0.124] | 0.109 [0.094, 0.125] | +0.001 |
| C5−C6 | 0.304 [0.282, 0.328] | 0.309 [0.285, 0.332] | +0.004 |
| C5−C7 | 0.086 [0.074, 0.099] | 0.087 [0.075, 0.101] | +0.001 |

Table 7: Post-hoc robustness check after excluding from evaluation all items with a token-Jaccard-similar neighbour ($\geq 0.7$). The same 10,000 sentence-grouped paired bootstrap is used; changes are computed from unrounded gaps.

| ID | Train transcripts | Test transcripts | Branch | Macro-F1 | Gain over gold-trained [95% CI] |
|---|---|---|---|---|---|
| A1 | Whisper | Whisper | text | 0.850 | 0.215 [0.192, 0.239] |
| A2 | Bengali XLS-R | Bengali XLS-R | text | 0.930 | 0.062 [0.048, 0.077] |
| A3 | Whisper | Whisper | fusion | 0.924 | 0.248 [0.225, 0.270] |
| A4 | Bengali XLS-R | Bengali XLS-R | fusion | 0.957 | 0.062 [0.050, 0.074] |
| X2 | Bengali XLS-R | Whisper | text | 0.815 | 0.180 [0.152, 0.207] |
| X1 | Whisper | Bengali XLS-R | text | 0.909 | 0.042 [0.026, 0.058] |

Table 8: ASR-matched adaptation (A1–A4) and post-hoc cross-recognizer transfer (X1–X2). X1 and X2 reuse the A1 and A2 text heads on the other recognizer's transcripts. Gains use unrounded scores and may differ by 0.001 from subtraction of the displayed values.

| | Whisper text | XLS-R text | Whisper fusion | XLS-R fusion |
|---|---|---|---|---|
| $T - B$ (threshold only) | 0.149 [0.124, 0.174] | 0.031 [0.019, 0.043] | 0.151 [0.129, 0.172] | 0.026 [0.017, 0.036] |
| $A - T$ (added by retraining) | 0.067 [0.050, 0.083] | 0.032 [0.020, 0.044] | 0.097 [0.084, 0.111] | 0.036 [0.026, 0.045] |
| $O - A$ (residual) | 0.126 [0.109, 0.142] | 0.046 [0.035, 0.058] | 0.057 [0.045, 0.068] | 0.025 [0.016, 0.033] |
| $A_{\text{thr}} - T$ | 0.061 [0.045, 0.078] | 0.031 [0.019, 0.043] | 0.097 [0.084, 0.111] | 0.031 [0.022, 0.041] |
| Gold-val. threshold $- B$ | 0.000 [−0.005, 0.005] | 0.003 [0.000, 0.007] | 0.008 [0.005, 0.012] | 0.002 [0.000, 0.004] |
| $(A - T)/(O - B)$ (%) | 19.5 [14.7, 24.7] | 29.3 [19.3, 39.5] | 31.9 [27.5, 36.7] | 41.2 [30.9, 52.1] |

Table 9: Operational decomposition of each oracle gap $O - B$ (secondary). $O$: oracle; $B$: gold-trained head, default rule; $T$: the same head with an ASR-validation threshold; $A$: matched head, default rule; $A_{\text{thr}}$: matched head with its own threshold. Paired 95% intervals from the shared bootstrap resamples; terms are rounded separately.

shows, this system-level gain cannot be attributed to sentiment-bearing acoustics, because measured acquisition/nuisance features reproduce a similar improvement.

The Whisper conditions show a large per-class asymmetry (C3: F1 0.749 negative against 0.521 positive), which is substantially reduced by threshold-only adaptation (Section 6.4 and Table 6). Aggregate class-conditioned errors are mixed: positive utterances have lower WER than negative ones (0.676 versus 0.699) but higher CER (0.367 versus 0.352), with the WER advantage driven by fewer insertions (0.069 versus 0.103; Table 12).

### A.15 Error-pattern diagnostic

Among the 1,725 recording–recognizer cases that are correct under oracle text and wrong under an ASR-substituted text condition, 1,267 involve Whisper and 458 involve Bengali XLS-R. A heuristic edit-pattern categorization flags 887 substitution-only cases, 561 with inserted content, 227 with deleted content and 130 with residual Latin-script material, whereas detected negation changes are rare (6 added and 3 lost). These counts are descriptive: the flags are neither mutually exclusive nor exhaustive, so they need not sum to 1,725, and they are not causal explanations. They nevertheless indicate that the observed failures are associated much more often with broad lexical corruption than with a negation-specific polarity reversal.

### A.16 Whisper decoding diagnostic and settings

Table 14 reports the decoding diagnostic behind the Whisper configuration choices described in Section 5, with the secondary beam-2 row. The sample is re-drawn with the diagnostic's own code and seed (75 utterances per speaker); its stored example outputs match. The other three rows used a 128-token budget: in cached greedy transcripts of the full corpus, 4 of the 300 sampled outputs come within one token of that budget, so the Bengali-forced greedy row may include a few truncated outputs, and 5

| | Seeds 13 / 42 / 87 | SD | Ens. |
|---|---|---|---|
| C1 | 0.936 / 0.934 / 0.937 | 0.002 | 0.940 |
| C2 | 0.966 / 0.980 / 0.972 | 0.007 | 0.976 |
| C3 | 0.654 / 0.627 / 0.628 | 0.016 | 0.635 |
| C4 | 0.872 / 0.869 / 0.861 | 0.006 | 0.867 |
| C5 | 0.980 / 0.983 / 0.979 | 0.002 | 0.981 |
| C6 | 0.682 / 0.680 / 0.673 | 0.005 | 0.677 |
| C7 | 0.892 / 0.898 / 0.894 | 0.003 | 0.895 |
| A1 | 0.850 / 0.848 / 0.847 | 0.001 | 0.850 |
| A2 | 0.926 / 0.932 / 0.919 | 0.006 | 0.930 |
| A3 | 0.927 / 0.922 / 0.919 | 0.004 | 0.924 |
| A4 | 0.953 / 0.953 / 0.950 | 0.002 | 0.957 |
| N5 | 0.976 / 0.977 / 0.978 | 0.001 | 0.980 |
| N6 | 0.691 / 0.671 / 0.709 | 0.019 | 0.692 |
| N7 | 0.894 / 0.879 / 0.899 | 0.010 | 0.890 |
| NA3 | 0.894 / 0.887 / 0.879 | 0.008 | 0.893 |
| NA4 | 0.939 / 0.940 / 0.938 | 0.001 | 0.941 |
| C2−C3 | 0.311 / 0.353 / 0.344 | 0.022 | 0.341 |
| C2−C4 | 0.094 / 0.111 / 0.111 | 0.010 | 0.109 |
| C5−C6 | 0.298 / 0.302 / 0.307 | 0.004 | 0.304 |
| C5−C7 | 0.088 / 0.085 / 0.085 | 0.002 | 0.086 |
| A1−C3 | 0.195 / 0.221 / 0.220 | 0.015 | 0.215 |
| A2−C4 | 0.053 / 0.063 / 0.058 | 0.005 | 0.062 |
| A3−C6 | 0.244 / 0.241 / 0.246 | 0.002 | 0.248 |
| A4−C7 | 0.060 / 0.055 / 0.056 | 0.003 | 0.062 |

Table 10: Training-seed variability (secondary): pooled out-of-fold macro-F1 per seed, its sample SD, and the ensemble score. Differences are computed within seed.

| Model | Protocol | f1 | f2 | m1 | m2 | Pooled |
|---|---|---|---|---|---|---|
| Speech head | joint holdout | 0.609 | 0.686 | 0.611 | 0.555 | 0.621 |
| Speech head | shared CV | 0.881 | 0.918 | 0.972 | 0.988 | 0.940 |
| Acq./nuisance LR | joint holdout | 0.207 | 0.338 | 0.323 | 0.223 | 0.333 |
| Acq./nuisance LR | shared CV | 0.711 | 0.583 | 0.352 | 0.922 | 0.669 |

Table 11: Macro-F1 per held-out speaker under the joint speaker-and-sentence holdout (secondary), next to the same models under the original shared-speaker cross-validation, which is not a leave-one-speaker-out result. For the shared-CV speech head, macro-F1 agrees with the accuracy in Table 13 to three decimals: each speaker's classes are balanced and its errors are few or nearly symmetric.

of the 300 beam-2 outputs are at least 127 tokens long.

The main run used `transformers` 5.14.1 and the model class `WhisperForConditionalGeneration` for whisper-large-v3 (revision `06f233fe`), calling `generate` with task `transcribe` and language `bn` (forced decoder ids cleared), beam 2, early stopping, 200 new tokens, no sampling, no prompt, no timestamps, repetition and length penalty 1.0 and no $n$-gram blocking (library defaults), float16, batch size 8, and 16 kHz mono audio padded to 30 s log-Mel features with an attention mask. Because every clip is at most 8.31 s, generation takes the short-form path; Whisper's long-form options `condition_on_prev_tokens`, `compression_ratio_threshold`, `logprob_threshold` and `no_speech_threshold` were not set and would be inactive. No hypothesis reaches the 200-token cap (the longest is about 172 tokens), no audio file failed to load, and no output is empty. Across the full corpus, 9.8% of Whisper hypotheses contain Latin-script characters, including 11 of the 20 with the highest WER. Only 2 of 3,996 show an immediate $n$-gram loop (an $n$-gram with $n \leq 4$ repeated at least three times in a row) or a zlib compression ratio above 2.4, and neither contains any of the insertions.

### A.17 Acquisition-order diagnostic

Table 15 reports the filesystem-timestamp reconstruction of acquisition order that supports the audit in Section 6.5.

### A.18 Orthographic canonicalization check

Table 16 reports the post-hoc orthographic canonicalization check. A canonicalization scheme differing more from our base normalizer might yield a different answer.

| ASR system | Class | $n$ | WER | CER | Sub. | Del. | Ins. |
|---|---|---|---|---|---|---|---|
| Bengali XLS-R CTC | negative | 1999 | 0.262 | 0.109 | 0.219 | 0.026 | 0.017 |
| Bengali XLS-R CTC | positive | 1997 | 0.290 | 0.114 | 0.241 | 0.031 | 0.018 |
| Whisper large-v3 | negative | 1999 | 0.699 | 0.352 | 0.586 | 0.011 | 0.103 |
| Whisper large-v3 | positive | 1997 | 0.676 | 0.367 | 0.595 | 0.012 | 0.069 |

Table 12: ASR quality by sentiment class. For Whisper, aggregate class-level WER runs opposite to the downstream F1 asymmetry, arguing against a simple class-dependent error-rate explanation (Appendix A.14).

| Spk | $n$ | WER-W | WER-B | C1 | C2 | C3 | C4 | C5 | C6 | C7 |
|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1000 | 0.713 | 0.312 | 0.881 | 0.976 | 0.662 | 0.856 | 0.975 | 0.672 | 0.863 |
| f2 | 1000 | 0.668 | 0.226 | 0.918 | 0.976 | 0.681 | 0.904 | 0.978 | 0.698 | 0.911 |
| m1 | 998 | 0.691 | 0.267 | 0.972 | 0.976 | 0.675 | 0.872 | 0.979 | 0.687 | 0.893 |
| m2 | 998 | 0.679 | 0.299 | 0.988 | 0.976 | 0.665 | 0.844 | 0.993 | 0.758 | 0.916 |

Table 13: Per-speaker results. The C1–C7 columns report accuracy. Speaker is confounded with recording condition in this corpus, so these results are descriptive only.

| Configuration | Budget | WER | CER | Latin-script output | Empty |
|---|---|---|---|---|---|
| Greedy, Bengali forced | 128 | 0.744 | 0.467 | 31.0% | 0.0% |
| Beam 5, Bengali forced | 128 | 0.679 | 0.352 | 2.3% | 0.0% |
| Greedy, autodetect | 128 | 0.750 | 0.481 | 31.0% | 0.0% |
| Beam 2, Bengali forced$^{\dagger}$ | 200 | 0.692 | 0.368 | 9.0% | 0.0% |

Table 14: Whisper decoding diagnostic on a 300-utterance speaker-stratified sample of this corpus. $^{\dagger}$Secondary row: it rescores the main run's transcripts, which every downstream condition uses, on the same utterances with the same normalizer and scoring. Budget is the maximum number of new tokens. WERs are measured on the sample and are not directly comparable to the full-corpus 0.687 in Table 1.

| Speaker | Switches | $r$(order, label) |
|---|---|---|
| m1 | 1 | $-0.866$ |
| f2 | 2 | $-0.863$ |
| m2 | 9 | $-0.784$ |
| f1 | 11 | $+0.476$ |

Table 15: Filesystem creation-timestamp acquisition-order diagnostic. Creation timestamps are used only as a proxy for acquisition order; switch counts and correlations are computed separately for each speaker.

| System | Orig. gap | Canon. gap | Changed |
|---|---|---|---|
| Whisper | 0.341 | 0.341 | 4/3996 |
| XLS-R | 0.109 | 0.109 | 0/3996 |

Table 16: Post-hoc orthographic canonicalization check. The externally specified BanglaBERT-oriented canonicalizer changes almost no hypotheses after the frozen base normalization, so this null result has narrow scope.

# MoEzhi-Guru: Multi-Perspective Misogynistic Meme Detection via Multimodal Mixture of Experts

**Sekar M**[*] [†] **Pranava Mohan**[*] **A. Santhanavijayan**
Department of Computer Science and Engineering
National Institute of Technology Tiruchirappalli, India
{406124006,106125084}@nitt.edu, vijayana@nitt.edu

## Abstract

Multimodal content moderation in online social spaces is a critical safety challenge, yet existing systems are heavily dominated by monolithic cultural biases and perform poorly on underrepresented languages. In this paper, we study culturally grounded misogynistic meme detection under a novel Multi-Perspective evaluation framework, which simultaneously assesses models across Indian, Chinese, and Western cultural lenses. We propose **MoEzhi-Guru** (drawing on the Dravidian term *mozhi* for language and the Indic term *guru* for expert/guide), an adaptable, resource-efficient Multimodal Mixture of Experts (MoE) model designed to handle complex cross-cultural and cross-lingual shifts. The system encodes input using mDeBERTa-v3 and SigLIP to extract rich multilingual text and global visual semantics, then fuses and routes these features through a custom MoE layer utilizing dense soft routing. This gating mechanism dynamically weights specialized expert subnetworks, capturing localized socio-cultural nuances before predicting labels through culture-specific classification heads. We evaluate our framework in four typologically diverse language subsets: Tamil and Malayalam (underrepresented Dravidian scripts), Chinese (containing colloquial and code-switched text), and English. Our system demonstrates robust cross-lingual generalization, yielding an overall average score of 0.8515 Macro-F1 and consistently exceeding a Gemma3-4b baseline by more than 0.42 Macro-F1 points. This work provides a scalable, inclusive foundation for culturally aware safety systems in low-resource and multilingual environments.

## 1 Introduction

Misogyny online is increasingly spread through multimodal formats like memes, which layer text over images to convey humor, sarcasm, or ideological beliefs. Detecting this kind of harmful content is difficult because the misogynistic signal is often implicit and grounded in specific cultural norms. Many current multimodal models struggle to generalize across cross-lingual and cross-cultural distribution shifts.

Cross-cultural misogynistic meme detection directly addresses this gap, benchmarking models across Indian, Chinese, and Western (English) cultural contexts. In this paper, we outline our approach to the Cross-Culture Evaluation task through **MoEzhi-Guru**. Since different cultures require different interpretive lenses, we designed a Mixture of Experts (MoE) architecture. Rather than relying on a single dense layer to process the fused multimodal features, our MoE routing mechanism lets the model dynamically allocate specialized capacity depending on the input, with the explicit goal of improving cross-cultural robustness.

## 2 Related Work

### 2.1 Multimodal Hate Speech and Misogyny Meme Detection

Early computational approaches to online abuse detection relied heavily on text-only architectures, utilizing classical machine learning or early transformer-based text encoders. However, internet memes violate text-only assumptions, as their meaning is deeply tied to multimodal interactions. To encourage the development of models capable of true multimodal reasoning, Kiela et al. introduced the *Hateful Memes Challenge* (Kiela et al., 2020). This dataset features "benign confounders", i.e., samples where either the image or the text is modified to flip the label, ensuring that unimodal shortcuts (e.g., detecting specific offensive words or visual triggers) are ineffective.

Following this, the SemEval-2022 Task 5 on *Multimedia Automatic Misogyny Identification*

[*] Equal contribution.
[†] Corresponding author.



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 55–63
November 9, 2026 ©2026 Association for Computational Linguistics

(MAMI) (Fersini et al., 2022) focused specifically on gender-targeted online abuse. The competition challenged models to perform binary misogyny detection (Sub-task A) and fine-grained categorization (Sub-task B), including shaming, stereotyping, objectification, and violence. Traditional SOTA approaches for MAMI leveraged large-scale pre-trained models such as VisualBERT, Oscar, UNITER, and ERNIE-Vil (Fersini et al., 2022). However, these frameworks exhibit significant performance gaps when evaluated on non-Western and low-resource languages, due to the lack of representative training corpora.

### 2.2 Social Media Safety in Low-Resource Dravidian Languages

Low-resource Dravidian languages such as Tamil and Malayalam present severe challenges for standard multilingual models. These languages are typologically diverse, morphologically rich, and characterized by extensive code-mixing (e.g., blending Dravidian scripts with English text) and informal phonetic spelling. To establish digital safety benchmarks in these contexts, Ponnusamy et al. introduced the *Misogyny Detection Meme Dataset* (MDMD) (Ponnusamy et al., 2024).

Recent shared tasks organized under the *DravidianLangTech* and *LT-EDI* workshops (Chakravarthi et al., 2024, 2025b) have accelerated research in Dravidian social media safety. Models evaluated on Tamil and Malayalam misogyny tasks often combine multilingual textual embeddings from IndicBERT, mBERT, or LaBSE with spatial visual features from ResNet or EfficientNet backbones (Ratul et al., 2025). More recently, Rajkumar et al. proposed a dual-stream architecture pairing XLM-RoBERTa with the Swin Transformer via cross-attention fusion (Rajkumar et al., 2026), showing consistent improvements in capturing implicit and contextual harmful signals. Furthermore, Dravidian political meme tasks highlight the necessity of capturing complex socio-political context, sarcasm, and regional visual symbolism to mitigate negative transfer across distinct domains (Rajiakodi et al., 2026).

### 2.3 Mixture of Experts and Gating Paradigms

The Mixture of Experts (MoE) paradigm, originally introduced by Shazeer et al. (Shazeer et al., 2017), scales model capacity by routing inputs dynamically to a set of specialized sub-networks (experts) via a trainable gating network. By executing only a handful of experts per forward pass (conditional computation), MoE architectures drastically scale model parameters without a proportional increase in FLOPS. Fedus et al. further simplified this framework with the *Switch Transformer* (Fedus et al., 2022), which utilizes a top-1 routing strategy to reduce communication overhead, simplify routing mechanics, and stabilize low-precision training.

While sparse MoE models are highly efficient for massive language modeling tasks, they are prone to training instabilities and expert collapse when applied to smaller, highly imbalanced multimodal safety datasets. In this paper, we address this by adopting dense soft routing. Rather than gating out experts, our router computes a soft probability distribution over all available subnetworks on every forward pass. This configuration maintains stable gradient flow, allows cooperative specialization across experts, and is intended to reduce gradient interference among divergent socio-cultural classification heads.

### 2.4 Multimodal Representation Learning

Modern multimodal architectures rely heavily on high-quality representation encoders. For multilingual natural language understanding, mDeBERTa-v3 (He et al., 2021) introduces replaced token detection (RTD) combined with gradient-disentangled embedding sharing (GDES) to decouple generator and discriminator gradients, achieving superior semantic and cross-lingual representation efficiency over XLM-RoBERTa. Its multilingual variant is pre-trained on the CC100 corpus (He et al., 2021), which includes Tamil and Malayalam, and it has been applied to abusive-comment detection in code-mixed Tamil, where it outperformed mBERT, MuRIL, and XLM-RoBERTa (Subramanian et al., 2025).

For the visual modality, contrastive vision-language pre-training (CLIP) has historically been the dominant approach. However, SigLIP (Zhai et al., 2023) proposes a simpler, pairwise sigmoid loss that replaces standard softmax normalization. SigLIP decouples batch size from the learning objective, showing exceptional zero-shot retrieval efficiency and robust multilingual visual-textual alignment on small hardware budgets. By fusing the representation capacity of mDeBERTa-v3 and SigLIP within our soft-routed MoE framework, our system captures subtle visual-textual dynamics across highly diverse cultural contexts.

## 3 Methodology

Our proposed architecture, **MoEzhi-Guru**, independently encodes text and images, fuses the two modalities, then passes the representation through an MoE layer before routing it to culture-specific classification heads. Figure 1 illustrates the high-level workflow of our system.

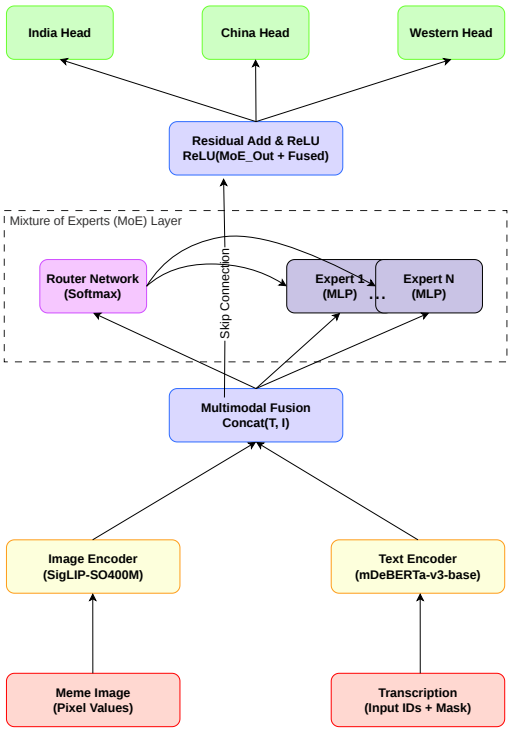



Figure 1: Overview of our MoEzhi-Guru Multimodal Mixture of Experts (MoE) architecture for cross-cultural meme detection.

### 3.1 Multimodal Encoders

**Text Encoder:** We use `microsoft/mdeberta-v3-base` (He et al., 2021) to extract rich, multilingual contextual text embeddings from meme transcriptions. Text features $T$ are taken as the CLS-token representation (the first token of the final hidden layer, `last_hidden_state[:, 0, :]`).

**Image Encoder:** For the visual modality, we use `google/siglip-so400m-patch14-384` (Zhai et al., 2023), taking advantage of its strong zero-shot image representation capabilities. Image features $I$ are obtained from the model's global pooled output (`pooler_output`). We enable gradient checkpointing on both encoders to keep memory usage manageable during training.

### 3.2 Feature Fusion and Mixture of Experts

We concatenate the text encoder's CLS-token representation $T$ and the image encoder's pooled output $I$ to form a unified multimodal representation:

$$F = \text{Concat}(T, I) \tag{1}$$

To capture divergent cultural interpretations, this fused representation $F$ is passed through an MoE layer. Our MoE consists of a routing network and $N$ expert networks. Each expert is a two-layer MLP with architecture $d \to d/2 \to d$, where $d = \dim(F)$ is the concatenated feature dimension. Specifically, the first projection maps down to $d/2$, followed by a GELU activation, a Dropout$(0.2)$ layer, and a second projection back to $d$. The router uses *dense (soft) routing*: all $N$ experts run on every forward pass, and the router produces a probability distribution over them via softmax, optimized end-to-end via the classification loss. The final output is the weighted sum of all experts' outputs:

$$M = \sum_{i=1}^{N} \text{softmax}(W_r F)_i \, E_i(F) \tag{2}$$

Here $W_r$ denotes the learned weights of the routing linear layer, and $E_i$ denotes the output of the $i$-th expert. Unlike sparse top-$k$ routing, no expert is ever fully gated out; every expert contributes to every prediction, proportionally to the router's confidence. A residual connection followed by a ReLU activation ensures stable gradient flow:

$$H = \text{ReLU}(M + F) \tag{3}$$

### 3.3 Classification Heads

The final representation $H$ feeds into three independent linear classification heads, one for each target cultural perspective: India, China, and Western. The model is trained using Binary Cross-Entropy with Logits Loss. Because the three heads are linear maps over the same representation $H$, they are equivalent to a single linear layer with three outputs and add no perspective-specific capacity; any perspective-dependent behaviour must therefore arise in the shared MoE layer.

## 4 Experimental Setup

### 4.1 Data and Preprocessing

We evaluate our approach on a consolidated benchmark combining the English MAMI dataset (Fersini et al., 2022), the Tamil

and Malayalam subsets from MDMD (Ponnusamy et al., 2024), and the Chinese CMMD dataset (Chakravarthi et al., 2025a). Images are preprocessed using the standard SigLIP image processor, and texts are tokenized via the mDeBERTa tokenizer with a maximum sequence length of 128 tokens. Missing perspective labels are masked with $-1.0$ and excluded from the loss calculation.

Throughout this paper, a cultural *perspective* (or *lens*) denotes the label assigned by an annotator group of a given national or regional background (Indian, Chinese, or Western). This is a coarse, national-level operationalization of culture that does not capture sub-national, linguistic, or individual variation. The perspective labels and their annotation protocol are those released by the CC-MMD 2026 organizers; we do not re-annotate any data.

Table 1: Dataset statistics per language and split. The percentage columns report the proportion of samples labeled as misogynistic under each cultural perspective. The perspective corresponding to a subset's native annotators is denoted with $^{\dagger}$.

| Language | Split | #Samples | Indian (%) | Chinese (%) | Western (%) |
|---|---|---|---|---|---|
| Tamil | Train | 1,137 | 24.0$^{\dagger}$ | 21.3 | 23.6 |
| | Dev | 284 | 26.4$^{\dagger}$ | 24.6 | 24.3 |
| Malayalam | Train | 640 | 40.0$^{\dagger}$ | 35.2 | 34.4 |
| | Dev | 160 | 40.0$^{\dagger}$ | 34.4 | 33.1 |
| English | Train | 9,395 | 30.2 | 52.0 | 50.0$^{\dagger}$ |
| | Dev | 600 | 29.8 | 50.7 | 49.8$^{\dagger}$ |
| Chinese | Train | 1,190 | 19.2 | 29.3$^{\dagger}$ | 25.3 |
| | Dev | 170 | 18.8 | 27.6$^{\dagger}$ | 18.2 |

As detailed in Table 1, the combined corpus comprises 13,576 samples across both splits. The distribution is dominated by the English (MAMI-derived) subset, which accounts for 9,995 samples ($\sim$74% of the corpus), while Tamil, Malayalam, and Chinese each contribute between a few hundred and just over a thousand instances per split. This substantial cross-lingual size disparity motivates the differential learning-rate strategy outlined in §4.2, enabling the randomly initialized Mixture-of-Experts (MoE) routing layers and task-specific classification heads to adapt rapidly without overfitting the smaller language subsets.

The perspective-level label distributions reveal pronounced cross-cultural divergence that directly justifies our multi-perspective task formulation. For the English subset, native Western annotators identify 50.0% of the training samples as misogynistic closely aligned with Chinese cross-cultural annotators at 52.0%. In contrast, Indian annotators label only 30.2% of the identical content as misogynistic, reflecting a notable $\sim$20 percentage-point gap. A similar divergence appears in the Chinese subset: native annotators assign misogynistic labels to 29.3% of samples, whereas Indian annotators flag only 19.2%. In contrast, the Tamil and Malayalam subsets exhibit narrower cross-cultural variance (within 2-6 percentage points), demonstrating that annotator disagreement varies based on cultural context rather than following a uniform offset. Our MoE-based architecture is explicitly designed to model this contextual variability.

Finally, the evaluation protocol accommodates differences in perspective annotations across subsets: Tamil, Malayalam, and Chinese subsets are evaluated across all three cultural perspectives (Native, Indian/Chinese/Western where applicable), whereas the English (MAMI) subset is evaluated on the two non-native cross-cultural perspectives (Indian and Chinese) since native Western perspectives are already captured at source. Consequently, overall English performance is computed as a 2-perspective Macro-F1 average.

### 4.2 Training and Hyperparameter Tuning

We train for 6 epochs using AdamW with a cosine learning rate schedule and a 10% warmup phase. We use a *differential learning rate* strategy: the pretrained backbones (mDeBERTa-v3 and SigLIP) receive base_lr $\times$ 0.1, while the randomly initialized MoE layer and classification heads receive base_lr $\times$ multiplier, letting the new components adapt more quickly than the pretrained encoders. We train with a per-device batch size of 2 (effective batch size of 4 via 2-step gradient accumulation) on one NVIDIA V100 GPU, using mixed-precision training (FP16) through PyTorch's (Paszke et al., 2019) Autocast and GradScaler. Gradient norms are clipped to 1.0 at each optimizer step via `clip_grad_norm_`.

One notable aspect of our loss formulation is that summed per-task losses are divided by the gradient accumulation step count, rather than by the number of active cultural perspectives in a batch. Thus, samples with only one valid cultural label still have their effective loss contribution halved. This behavior was retained across all trials as part of the final tuned configuration.

We performed hyperparameter optimization using Optuna (Akiba et al., 2019) over 25 trials within a 48-hour budget. The search space in-

cluded the base learning rate (log-uniform over $[10^{-6}, 5\times10^{-5}]$), the MoE learning rate multiplier (uniform over $[1.0, 10.0]$), and the number of experts ($\{4, 6, 8\}$). Underperforming trials were pruned using Optuna's `MedianPruner`, and the study was checkpointed to an SQLite database for seamless resumption. The optimal setup used 4 experts, a base learning rate of $4.2841 \times 10^{-5}$, and an MoE learning-rate multiplier of 3.2128, producing an encoder-to-MoE learning rate ratio of approximately $1:32.1$.

### 4.3 Evaluation Metrics

For binary misogyny classification, let $\mathcal{C} = \{0, 1\}$ be the label set and let $\mathrm{TP}_c$, $\mathrm{FP}_c$, $\mathrm{FN}_c$ denote the true positives, false positives, and false negatives for class $c$.

Per-class Precision ($P_c$), Recall ($R_c$), and $F1$ ($F1_c$) are defined as:

$$P_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c}, \quad R_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c}$$
$$F1_c = \frac{2 \cdot P_c \cdot R_c}{P_c + R_c} \tag{4}$$

The *Macro-F1* score averages $F1_c$ equally across classes, removing class-frequency bias:

$$\text{Macro-F1} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} F1_c. \tag{5}$$

The overall score for a language subset $\ell$ with applicable cultural perspectives $\mathcal{P}_\ell$ is the unweighted mean Macro-F1 across all perspectives:

$$\text{Score}_\ell = \frac{1}{|\mathcal{P}_\ell|} \sum_{p \in \mathcal{P}_\ell} \text{Macro-F1}_p. \tag{6}$$

For Tamil, Malayalam, and Chinese subsets $|\mathcal{P}_\ell| = 3$; for English $|\mathcal{P}_\ell| = 2$ (see §4.1). Mean accuracy across perspectives serves as a secondary tie-breaker metric.

## 5 Results

### 5.1 Comparison Against Baseline

Table 2 compares our system against a Gemma3-4b baseline. Baseline scores[1] are computed as the unweighted average of the per-culture Macro-F1 values reported for each language. Our architecture improves consistently over the baseline by 0.42 to 0.64 Macro-F1 points across all four language subsets. As these baseline figures were not reproduced under our training setup, this comparison serves as a reference point rather than a controlled comparison; the controlled architectural comparison is the ablation in §5.3.

Table 2: Comparison against a Gemma3-4b baseline (unweighted average Macro-F1). The English baseline uses the 2-perspective average (Indian + Chinese).

| Language | Baseline | Ours | Δ |
|---|---|---|---|
| Tamil | 0.3247 | 0.8427 | **+0.518** |
| Malayalam | 0.3251 | 0.9133 | **+0.588** |
| Chinese | 0.2283 | 0.8686 | **+0.640** |
| English | 0.3574 | 0.7815 | **+0.424** |

### 5.2 Per-Culture Results

Table 3 reports [2] the per-culture Macro-F1 and accuracy scores, providing the full breakdown across all cultural perspectives for each language subset.

Table 3: Per-culture Macro-F1 results.

| Language | Perspective (Macro-F1) | | | **Overall** | |
|---|---|---|---|---|---|
| | Indian | Western | Chinese | **Macro-F1** | **Accuracy** |
| Tamil | 0.820 | 0.851 | 0.857 | **0.843** | 0.783 |
| Malayalam | 0.925 | 0.925 | 0.890 | **0.913** | 0.908 |
| Chinese | 0.874 | 0.856 | 0.876 | **0.869** | 0.808 |
| English* | 0.796 | - | 0.767 | **0.782** | 0.760 |

*English evaluated on 2 perspectives only.

### 5.3 Ablation Study

To evaluate the contribution of each input modality and assess the incremental effect of the proposed Mixture of Experts (MoE) fusion architecture, we conduct a controlled ablation study. Specifically, our evaluation investigates two core research dimensions:

1. **Modality Contributions**: Dissecting the relative impact of individual modalities by comparing unimodal baselines (*Text-Only* utilizing `mDeBERTa-v3-base` and *Image-Only* utilizing `SigLIP-so400m`) against joint multimodal representations ($F = [\mathbf{e}_{\text{text}} \,\|\, \mathbf{e}_{\text{img}}]$).

2. **Fusion Architecture Efficiency**: Evaluating fusion mechanisms by comparing a conventional *Single Dense Fusion Layer* (a 2-layer

[1] Baseline results sourced from the official CC-MMD 2026 Challenge Tasks benchmark: https://sites.google.com/view/cc-mmd2026/challenge-tasks?authuser=0

[2] All reported benchmark metrics correspond to official evaluation results released and publicly available through the CC-MMD Challenge evaluation platform.

MLP without dynamic routing) against our proposed *MoE Fusion Layer* configured with soft-router gating across $N = 4$ expert sub-networks.

All ablation variants are trained under strictly identical experimental setups, optimization hyper-parameters (differential base learning rate base_lr $= 2 \times 10^{-5}$, AdamW optimizer, cosine decay schedule), and random initialization seeds ($S = 42$) to isolate architectural effects. For computational uniformity across ablation variants, we fix $N = 4$ experts. Experimental results across all target cultural contexts are reported in Table 4.

**Necessity of Multimodal Integration** As demonstrated in Table 4, unimodal models exhibit noticeable performance degradation when evaluated across diverse cultural perspectives. The *Text-Only* variant achieves an Overall Macro-F1 of $0.6984$, effectively capturing textual transcripts but remaining blind to visual meme templates and non-verbal humor. Conversely, the *Image-Only* model achieves $0.7330$ Overall Macro-F1. The image-only variant likely outperforms the text-only one because meme text is rendered inside the image and was not removed: SigLIP, pre-trained on large-scale image-text data, can read at least part of this overlaid text, so the image input carries both visual and partial textual signal. Fusing both modalities in *Dense Fusion* increases the overall Macro-F1 to $0.7472$ (+0.0488 absolute / +6.99% relative over Text-Only and +0.0142 absolute / +1.94% relative over Image-Only), demonstrating that cross-modal synergy is indispensable for cross-cultural meme analysis.

**Superiority of MoE Expert Routing over Dense MLP** A critical comparison lies between the *Dense Fusion* baseline ($0.7472$ Overall F1) and our proposed *MoE Fusion* architecture ($0.8491$ Overall F1). Conventional dense fusion forces all cultural context heads to share a rigid, single set of transformation weights, inducing negative cross-cultural transfer. In contrast, the router-gated MoE layer dynamically allocates capacity across $N = 4$ specialized expert sub-networks. This dynamic gating yields substantial performance gains across all target cultural contexts, boosting the Overall Macro-F1 by $+0.1019$ absolute points (+13.64% relative over Dense Fusion). The largest gain is observed in the China context ($0.8700$ Macro-F1, $+0.1378$ over Dense Fusion), followed by the Western context ($0.8653$ Macro-F1, $+0.1094$ over Dense Fusion) and the India context ($0.8120$ Macro-F1, $+0.0585$ over Dense Fusion). We hypothesize that this gain stems from the router allocating capacity differently across cultural contexts, mitigating negative transfer between perspective heads; directly verifying expert-level specialization via routing analysis is left to future work.

### 5.4 Discussion

Our experiments reveal strong cross-cultural performance for all four language subsets. In **Tamil**, the model is less proficient in understanding from the native Indian viewpoint (0.820) compared to the Western (0.851) and Chinese (0.857) viewpoints due to higher variance among annotators in source-culture annotations. Our model has similar performance metrics for **Malayalam** and **Chinese** (within 0.04 Macro-F1 for all viewpoints), which means MoE routing works fine even though there are different class distributions. Lower performance of **English** can be attributed to the difficulty of predicting the meaning of memes for Indian and Chinese audiences.

Although the evaluation metric is calculated per-language separately, one of the strongest features of the MoE architecture we implemented is its domain-independent consistency. In order to test this assumption, we calculated the unweighted overall average Macro-F1 for all four languages. Our architecture obtained an average overall Macro-F1 score of 0.8515, indicating that the model successfully avoids performance degradation across disparate languages, generalizing well across scripts rather than specializing narrowly in a few. Because the English subset is scored on only two perspectives (§4.1), this average is a summary statistic rather than a uniform cross-lingual measure.

As demonstrated in our ablation study (§5.3), the Mixture of Experts layer outperforms a conventional dense layer, while our hyperparameter optimization phase (detailed in §4.2) provides further empirical insight into how the Mixture of Experts layer scales with capacity. During our Optuna search space evaluation of 4, 6, and 8 expert configurations, we observed that performance remains notably stable. The top development-set Macro-F1 scores for the 4, 6, and 8 expert setups were 0.8491, 0.8383, and 0.8425, respectively. This tight spread (less than 0.011) suggests that dense soft routing is robust to the number of experts, avoiding the

Table 4: Ablation study comparing input modalities (Text-Only, Image-Only, Joint Multimodal) and fusion architectures (Single Dense MLP vs. Proposed MoE Layer) evaluated by Macro F1 score across target cultural contexts.

| Model Variant | Modalities | Fusion Mechanism | India F1 | China F1 | Western F1 | Overall F1 |
|---|---|---|---|---|---|---|
| Text-Only | Text Only (`mDeBERTa-v3`) | N/A | 0.6898 | 0.7113 | 0.6941 | 0.6984 |
| Image-Only | Image Only (`SigLIP-so400m`) | N/A | 0.7413 | 0.7400 | 0.7176 | 0.7330 |
| Dense Fusion | Text + Image | Single Dense MLP | 0.7535 | 0.7322 | 0.7559 | 0.7472 |
| **MoE Fusion (Ours)** | **Text + Image** | **MoE Layer ($N = 4$ Experts)** | **0.8120** | **0.8700** | **0.8653** | **0.8491** |

instability or expert collapse often seen in sparsely-gated models on small datasets.

**Error Analysis:** While our model demonstrates strong overall performance, per-perspective metrics reveal specific cross-cultural failure modes. Interestingly, although our system performs competitively on the English subset, its absolute Macro-F1 score (0.782) is the lowest across all languages. This suggests the English subset (MAMI) poses a universally high difficulty, likely because the task requires predicting Indian and Chinese perspectives on Western-origin memes, a highly challenging cross-cultural semantic transfer. Furthermore, on typologically distinct datasets like Tamil, the model struggles more with the native Indian perspective (0.820) compared to the Western (0.851) and Chinese (0.857) lenses. One possible explanation is that when a meme's misogynistic cue relies on native, non-Western visual context or localized pop-culture tropes, the model leans towards interpretations learned from the English-dominated training data, resulting in false negatives for the native perspective. The pretraining distributions of our backbones may also contribute, as SigLIP's global visual representations may not adequately capture deeply localized regional contexts. A qualitative, example-level analysis is needed to confirm this pattern.

### 5.5 Limitations

Despite strong overall results, several limitations of our system are worth acknowledging. First, our MoE uses dense soft routing, meaning all $N$ experts run on every forward pass (Eq. 2). While this avoids expert collapse on a small corpus, it gives up the computational savings of sparse top-$k$ routing and does not enforce strict expert specialization; we have not yet analyzed routing distributions to verify whether experts specialize by culture or language. Second, the benchmark covers only three cultural perspectives (Indian, Chinese, Western), whereas meme misogyny is a global phenomenon. Communities in South-East Asia, Latin America, and sub-Saharan Africa are entirely absent, which limits how broadly the generalization claims can be applied. Third, meme transcriptions are truncated to 128 tokens; longer memes in which the sarcastic or misogynistic cue appears late in the text may be silently cut off, degrading the text signal. Finally, dividing the loss by the gradient-accumulation step count rather than by the number of active cultural perspectives per batch halves the effective learning signal for single-perspective samples and may under-weight rare cross-cultural labels during training. In addition, all reported results come from a single training seed, so run-to-run variance is not quantified; the Gemma3-4b baseline was taken from the challenge rather than reproduced under our setup; and the training mixture is dominated by English (MAMI) data ($\sim$74%), so low-resource performance may depend on this mixture, which we did not vary.

## 6 Conclusion

We presented MOEZHI-GURU, a multimodal Mixture of Experts system for cross-cultural misogynistic meme detection. By combining mDeBERTa-v3 and SigLIP with a specialized MoE routing layer and culture-specific classification heads, the model effectively captures the implicit cultural nuances that make misogynistic memes so difficult to detect. Results show improvements exceeding 0.42 Macro-F1 points over a Gemma3-4b baseline. Our model's cross-dataset consistency yields an average Macro-F1 of 0.8515 across all four language subsets combined, confirming that the MoE approach is a strong foundation for unified, globally robust multimodal moderation systems.

## Ethics and Privacy Statement

This research addresses automated detection of harmful misogynistic content, with the broader aim of contributing to safer online environments. We do, however, acknowledge the potential for dual use: automated moderation tools can mistakenly

flag benign cultural expressions when training data encodes cultural biases. Our use of cross-cultural datasets helps mitigate monolithic cultural bias, but further evaluation is needed to ensure fairness across all marginalized sub-groups. No personally identifiable information was used beyond the publicly released datasets.

## References


Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631.

Bharathi Raja Chakravarthi, Rahul Ponnusamy, Ping Du, Xiaojian Zhuang, Saranya Rajiakodi, Paul Buitelaar, Premjith B, Bhuvaneswari Sivagnanam, Anshid Kizhakkeparambil, and Lavanya S.K. 2025a. An overview of the misogyny meme detection shared task for Chinese social media. In *Proceedings of the 5th Conference on Language, Data and Knowledge: Fifth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 200–208, Naples, Italy. Unior Press.

Bharathi Raja Chakravarthi, Rahul Ponnusamy, Saranya Rajiakodi, Shunmuga Priya Muthusamy Chinnan, Paul Buitelaar, Bhuvaneswari Sivagnanam, and Anshid Kizhakkeparambil. 2025b. Findings of the shared task on misogyny meme detection: Dravidianlangtech@ naacl 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 721–731.

Bharathi Raja Chakravarthi, Saranya Rajiakodi, Rahul Ponnusamy, Kathiravan Pannerselvam, Ramachandran Rajalakshmi, Hariharan LekshmiAmmal, Anshid Kizhakkeparambil, Susminu S Kumar, Bhuvaneswari Sivagnanam, Charmathi Rajkumar, et al. 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.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research (JMLR)*, 23(120):1–43.

Elisabetta Fersini, Francesca Gasparini, Giulia Rizzi, Aurora Saibene, Berta Chulvi, Paolo Rosso, Alyssa Lees, and Jeffrey Sorensen. 2022. SemEval-2022 task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 533–549, Seattle, United States. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

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*, volume 33, pages 2611–2624. Curran Associates, Inc.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Rahul Ponnusamy, Kathiravan Pannerselvam, Saranya Rajiakodi, Prasanna Kumar Kumaresan, Sajeetha Thavareesan, Bhuvaneswari Sivagnanam, Anshid K.A, Susminu S Kumar, Paul Buitelaar, and Bharathi Raja Chakravarthi. 2024. From laughter to inequality: Annotated dataset for misogyny detection in Tamil and Malayalam memes. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7480–7488, Torino, Italia. ELRA and ICCL.

Saranya Rajiakodi, Shunmuga Priya Muthusamy Chinnan, Subalalitha Cn, Rahul Ponnusamy, Anshid Kizhakkeparambil, Bhuvaneswari Sivagnanam, Bharathi Raja Chakravarthi, et al. 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.

Charmathi Rajkumar, Malliga Subramanian, and Bharathi Raja Chakravarthi. 2026. Multimodal transformer framework for multilingual harmful meme classification. In *Proceedings of the Sixth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 99–107.

MD Musa Kalimullah Ratul, Sabik Aftahee, Tofayel Ahmmed Babu, Jawad Hossain, and Mohammed Moshiul Hoque. 2025. Cuet_networksociety@ dravidianlangtech 2025: A multimodal framework to detect misogyny meme in dravidian languages. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 529–535.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff

Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*.

Malliga Subramanian, Sathishkumar Veerappampalayam Easwaramoorthy, and Nandhini Subbarayan. 2025. Empowering sentiment analysis in social media: a comprehensive approach to enhance the classification of abusive Tamil comments using transformer models. *Journal of Big Data*, 12:208.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11941–11952.

# Axiom@MMLow 2026: Cascaded Prompting with QLoRA-Tuned Qwen2-VL for Hierarchical Tamil Political Meme Classification

**Md. Ajwad Hossain**
Chittagong University of Engineering and Technology
md.ajwadhossain@gmail.com

## Abstract

We present our system for the MMLow@AACL-IJCNLP 2026 shared task on multi-level classification of Tamil political memes. Our approach fine-tunes Qwen2-VL-2B-Instruct with QLoRA under a cascaded prompting strategy, in which a Level 1 stance prediction (Support/Praise vs. Troll/Oppose) conditions the prompt used for Level 2 target classification (party vs. person, per stance). On the official test set, our system achieves a Level 1 macro-F1 of 0.7173 and a Level 2 macro-F1 of 0.4429, for a final score of 0.5801, placing 5th of 7 teams. Our error analysis shows that all Level 1 errors propagate downstream, accounting for the majority of Level 2 mistakes. Additionally, a preliminary qualitative review suggests that sarcasm interpreted literally drives a large share of Level 1 misclassifications, pointing to sarcasm detection and cross-modal reasoning as key bottlenecks for this task.

## 1 Introduction

Political memes compress complex messages into shareable images, shaping discourse, mobilizing supporters, and occasionally spreading misinformation (Mihailescu, 2024; AlAfnan, 2025; Munawar, 2025). Understanding them requires genuine cross-modal reasoning, as a meme's meaning typically relies on the cross-modal interaction between individually benign image and text rather than anything literally stated (Kiela et al., 2020; Pramanick et al., 2021; Dimitrov et al., 2021).

Tamil political memes compound this difficulty. They draw on code-mixed Tamil-English, a linguistic setting that remains under-resourced (Chakravarthi et al., 2021) and heavily utilize culturally specific humor and sarcasm (Suryawanshi et al., 2020). While meme classification has matured considerably for English content, code-mixed settings "remain comparatively underexplored; historical Tamil multimodal benchmarks have consistently peaked well below comparable English tasks (Suryawanshi et al., 2020; Suryawanshi and Chakravarthi, 2021; B. et al., 2022).

This paper describes our system[1] for the MMLow@AACL-IJCNLP 2026 shared task, targeting hierarchical stance and target classification for Tamil-English political memes.Our contribution is threefold: we fine-tune Qwen2-VL-2B under tight hardware constraints using QLoRA, structure inference as a two-step cascaded prompting process, and provide a detailed error analysis identifying literal interpretations of sarcasm as a primary failure mode, corroborating prior warnings in this domain. (Suryawanshi et al., 2020; Singh et al., 2024).

## 2 Related Work

**Meme Classification and VLMs.** The Hateful Memes Challenge established the standard for subtle, multimodal reasoning (Kiela et al., 2020), which later expanded into political and propagandistic domains (Pramanick et al., 2021; Dimitrov et al., 2021). Modeling approaches for English memes have rapidly evolved from early fusion baselines (Hasan et al., 2022; Abdullakutty and Naseem, 2024) to CLIP-aligned encoders (Kumar and Nandakumar, 2022; Shah et al., 2024), and most recently to pipelines that prompt large vision-language models (VLMs) directly (Hee and Lee, 2025; Tzelepi and Mezaris, 2025). Our cascaded approach adapts this latest VLM paradigm to a hierarchical task.

**Code-Mixing and Sarcasm.** Our task extends the DravidianCodeMix trajectory, spanning from offensive language identification to multimodal troll meme classification (Chakravarthi et al., 2021; Suryawanshi et al., 2020; B. et al., 2022). Across

[1] Code available on GitHub.



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 64–69
November 9, 2026 ©2026 Association for Computational Linguistics

these tasks, sarcasm remains a notorious obstacle due to cross-modal incongruity that surface-level sentiment cues miss (Yue et al., 2023; Liang et al., 2022). While prior efforts confirm the difficulty of Tamil meme analysis, none have addressed the fine-grained hierarchical labels targeted in the current shared task.

**Multilingual Grounding and PEFT.** Separately, predominantly English-trained VLMs frequently degrade on low-resource tasks due to weak multilingual instruction-following (Schmidt et al., 2025; Atuhurra et al., 2025; Qu et al., 2026; Li et al., 2025). To adapt these massive backbones to code-mixed Tamil under severe compute constraints, we employ QLoRA. This parameter-efficient fine-tuning (PEFT) technique injects trainable low-rank matrices into frozen quantized weights, drastically reducing the memory footprint while maintaining competitive downstream performance (Hu et al., 2021; Martín-Fernández et al., 2025; Elgendy et al., 2025).

## 3 Methodology

### 3.1 Data

The shared task dataset consists of Tamil and Tamil-English code-mixed political memes, each labeled at two hierarchical levels. Table 1 shows the class distribution in the working training split, before and after balancing.

The raw distribution is lopsided, memes opposing an individual outnumber every other category by a wide margin, while support-for-party memes are scarce. Training on this directly risks pushing the model toward the majority class. The balanced subset fixes this by matching counts first across the two Level 1 classes, then across the Level 2 subclasses within each, leaving 324 training examples split almost evenly at Level 1 (163 Support/Praise, 161 Troll/Oppose).

| Level 2 Class | Raw | Balanced |
|---|---|---|
| Troll/Oppose Against Person | 465 | 83 |
| Support for Person | 150 | 81 |
| Troll/Oppose Against Party | 124 | 80 |
| Support for Party | 84 | 80 |
| **Total** | 823 | 324 |

Table 1: Training data class distribution before and after hierarchical balancing.

The full training set contains 969 labeled examples, 823 of which form the working split; the rest are held out for validation. The official test set comprises 244 unlabeled examples, with gold labels released by the organizers only after the evaluation period closed.

### 3.2 Model Architecture

Qwen2-VL-2B-Instruct serves as the backbone, chosen mainly for practical reasons: native support in the transformers library, decent OCR on embedded meme text out of the box, and stable performance under SDPA attention on hardware without FlashAttention2.

### 3.3 Cascaded Prompting

Rather than predicting both labels in one pass, the task splits into two sequential generation steps. The model first judges Level 1 stance from the image and its embedded text; that prediction then feeds into a second prompt asking for the specific Level 2 category (exact prompts in Appendix A). Training conditions the second step on the ground-truth Level 1 label, but at inference the model has only its own prediction to work with. That gap between training and inference matters a great deal, as Section 6 shows. Because the VLM generates free-text, we apply a "label snapping" function to raw outputs: we first check for an exact label match, then fall back to a substring match, and finally default to a fallback class if the generation is completely invalid.

### 3.4 Parameter-Efficient Fine-Tuning

Fine-tuning uses QLoRA with 4-bit NF4 quantization and bfloat16 compute precision. LoRA adapters (rank 16, alpha 32, dropout 0.05) attach to all attention and MLP projection layers, giving roughly 18.5M trainable parameters out of 2.23B total (0.83%). Full hyperparameters are in Appendix B.

## 4 Experimental Setup

Training runs on a single Tesla T4 GPU (15.6GB VRAM) on a free-tier cloud notebook, and that hardware shapes several design decisions. Qwen2-VL's dynamic-resolution tokenizer produces a different image-token count per sample, ruling out large batch sizes; instead, examples were processed with a micro-batch size of 2 and gradient accumulation steps of 8. Gradient checkpointing is likewise not optional: turning it off causes immediate out-of-memory failures even at the smallest batch size tested. Three epochs of training run in total, with

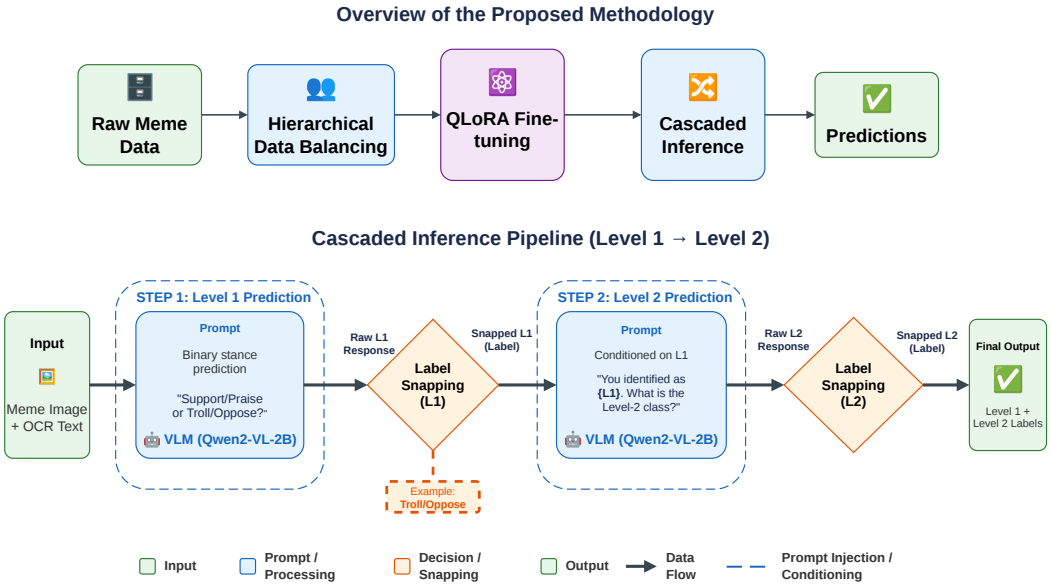



Figure 1: Cascaded inference pipeline (Level 1 → Level 2). The Level 1 binary stance prediction is inserted into the Level 2 prompt template, conditioning the detailed target classification on the model's own upstream prediction.

the final checkpoint chosen by validation accuracy on a held-out training split. These constraints also rule out a hyperparameter sweep, a LoRA rank ablation, and any comparison against larger or more explicitly multilingual backbones — open questions for future work rather than conclusions drawn without testing.

## 5 Results

| Split | L1 Acc | L2 Acc |
|---|---|---|
| Zero-shot baseline | 70.0% | 50.0% |
| Epoch 1 (val, best) | 92.5% | 62.5% |
| Epoch 2 (val) | 80.0% | 52.5% |
| Epoch 3 (val) | 87.5% | 60.0% |
| Test (official) | 75.0% | 57.0% |

Table 2: Validation accuracy across training epochs vs. final official test accuracy, using the epoch-1 checkpoint.

The epoch-1 checkpoint goes into the final submission, and the choice is fairly clear-cut: training loss sharply drops after the first epoch before plateauing slightly ($0.3275 \rightarrow 0.1138 \rightarrow 0.1181$), but validation accuracy peaks after epoch 1 and never comes back indication of classic overfitting on a small balanced subset. On the official test set, this checkpoint scores 0.7173 macro-F1 at Level 1 and 0.4429 at Level 2, for a final combined score of 0.5801, good for 5th place out of 7 teams.

The gap between the validation numbers (92.5%/62.5%) and the official test result (75.0%/57.0%) is larger than ideal, and three factors likely drive most of it. The validation split comes from the same source distribution as training, so it probably understates how different the real test set turns out to be. Validation scoring also relies on lenient substring matching rather than exact match, which almost certainly inflates the numbers relative to the official strict scorer. And epoch checks cover only a 40-sample subset of validation data to save time, adding noise to an already tight selection call.

## 6 Error Analysis

Since the Level 2 prompt conditions on the model's own Level 1 prediction at inference, not the ground truth, any Level 1 mistake gets a direct shot at derailing Level 2. That propagation turns out to be complete rather than partial: Level 2 accuracy sits at 75.96% when Level 1 is correct, but drops to exactly **0.00%** when Level 1 is wrong ($n = 61$). While all 61 Level 1 errors successfully derailed Level 2, approximately 44 additional Level 2 errors occurred even when Level 1 was correct. Thus, roughly 58% of Level 2 mistakes originate directly from upstream Level 1 failures; nothing in the current pipeline lets the model second-guess an earlier call.

The Level 1 confusion matrix shows a clear lean: genuine Troll/Oppose content gets mistaken for

Support/Praise about twice as often as the reverse (41 vs. 20 cases). Target identification alone (person vs. party, ignoring stance) fares better at 75.8% accuracy, though a bias shows up here too, "Support for Person" gets over-predicted relative to gold (81 vs. 45) while "Support for Party" gets under-predicted (10 vs. 25), pointing to a default toward naming a recognizable individual even when a meme's real target is a party or organization.

A manual review of 15 misclassified examples (categorized in Appendix C) digs into what actually drives the Level 1 errors. While this sample is too small for definitive statistical conclusions, this preliminary review suggests sarcasm read literally is the largest single pattern, at roughly 40% of the sample: memes quoting a political figure's self-congratulatory claim, undercut moments later by an ironic follow-up, get scored as genuine praise. A smaller, opposite pattern shows up too (about 27%): intensely worded but genuinely supportive fan rhetoric gets mistaken for hostility, likely because surface tone outweighs underlying intent in the model's judgment. The rest split between cases needing specific cultural or political background knowledge and a handful that are genuinely ambiguous even under close manual review.

## 7 Discussion

Set against the top-performing system's 0.9027/0.6694 L1/L2 macro-F1, this result points to a real multilinguality gap. Qwen2-VL-2B is pretrained mainly on English and Mandarin text, and while OCR on embedded Tamil script works well enough, its grip on Tamil-English code-mixed political discourse seems noticeably looser, especially around sarcasm, which leans on culturally specific cues a model without strong grounding in the language and its politics would likely miss. That lines up with the top-ranked teams' apparent choice of more broadly multilingual backbones, though this hypothesis hasn't been tested directly against an alternative backbone given the compute budget here, so it stands as a plausible explanation rather than a demonstrated one.

## 8 Conclusion

This paper presents a cascaded system for hierarchical Tamil political meme classification, fine-tuning Qwen2-VL-2B-Instruct with QLoRA under fairly tight hardware constraints. The final system reaches a combined score of 0.5801 (L1 macro-F1 0.7173, L2 macro-F1 0.4429), placing 5th among 7 teams. Two failure modes stand out from the error analysis: the strict downstream propagation of Level 1 errors under the cascade, and preliminary qualitative evidence pointing to literal interpretations of sarcasm as a primary contributor to Level 1 mistakes. Both point fairly directly to where the work should go next: architectures that let the model revisit an early misstep instead of committing to it, backbones with stronger multilingual and code-mixed grounding, and training data curated with sarcasm and irony specifically in mind.

## Limitations

The validation setup relies on lenient substring matching and checks only a 40-sample subset of validation data per epoch, likely overstating real performance relative to the official strict-match scorer and contributing to potential instability in model selection; exact-match scoring from the start would serve future work better. Additionally, discarding over half of the available training data (823 down to 324 examples) for hierarchical balancing likely harmed generalization in an already low-resource setting. Future iterations should explore class-weighted losses or oversampling instead of strict undersampling.

Crucially, because the Level 2 labels inherently combine stance and target, a direct four-way classification model might implicitly recover both levels without requiring an upstream prediction. A limitation of this work is that cascaded prompting was not explicitly ablated against a joint or direct four-way prediction baseline, meaning the isolated benefit of the cascade remains an open question.

Finally, a non-native Tamil speaker carried out the qualitative error review. While the broad categories assigned seem reasonably solid, subtler sarcasm markers may have gone unnoticed; native-speaker annotation would strengthen this analysis. Compute constraints also rule out hyperparameter ablations, LoRA rank sweeps, and head-to-head comparisons against larger or more explicitly multilingual backbones, which remain the most promising next steps rather than settled questions.

## References


F. Abdullakutty and Usman Naseem. 2024. Decoding memes: A comprehensive analysis of late and early fusion models for explainable meme analysis. *Companion Proceedings of the ACM Web Conference 2024*.

M. AlAfnan. 2025. The role of memes in shaping political discourse on social media. *Studies in Media and Communication*.

Jesse Atuhurra, Iqra Ali, Tomoya Iwakura, Hidetaka Kamigaito, and Tatsuya Hiraoka. 2025. Vlures: Benchmarking vlm visual and linguistic understanding in low-resource languages. *Preprint*, arXiv:2510.12845.

P. B., Bharathi Raja Chakravarthi, Malliga Subramanian, Bharathi B., Soman Kp, Kishore D., Sreelakshmi K., Arunaggiri Pandian, and Prasanna Kumar Kumaresan. 2022. Findings of the shared task on multimodal sentiment analysis and troll meme classification in dravidian languages. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*.

Bharathi Raja Chakravarthi, R. Priyadharshini, Vigneshwaran Muralidaran, Navya Jose, Shardul Suryawanshi, E. Sherly, and John P. McCrae. 2021. Dravidiancodemix: sentiment analysis and offensive language identification dataset for dravidian languages in code-mixed text. *Language Resources and Evaluation*, 56:765–806.

Dimitar I. Dimitrov, Bishr Bin Ali, S. Shaar, Firoj Alam, F. Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. 2021. Detecting propaganda techniques in memes. pages 6603–6617.

Hosam Elgendy, Ahmed Sharshar, Ahmed Aboeitta, Yasser Ashraf, and Mohsen Guizani. 2025. Geollava: Efficient fine-tuned vision-language models for temporal change detection in remote sensing. *Preprint*, arXiv:2410.19552.

M. Hasan, Nusratul Jannat, Eftekhar Hossain, Omar Sharif, and M. M. Hoque. 2022. Cuet-nlp@dravidianlangtech-acl2022: Investigating deep learning techniques to detect multimodal troll memes. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*.

Ming Shan Hee and Roy Ka-Wei Lee. 2025. Demystifying hateful content: Leveraging large multimodal models for hateful meme detection with explainable decisions. *Preprint*, arXiv:2502.11073.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

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 *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Gokul Karthik Kumar and Karthik Nandakumar. 2022. Hate-CLIPper: Multimodal hateful meme classification based on cross-modal interaction of CLIP features. In *Proceedings of the Second Workshop on NLP for Positive Impact (NLP4PI)*, pages 171–183, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Juncheng Li, Qing Yang, Bojian Jiang, Shaolin Zhu, and Qingxuan Sun. 2025. Lrm-llava: Overcoming the modality gap of multilingual large language-vision model for low-resource languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 24449–24457.

Bin Liang, Chenwei Lou, Xiang Li, Min Yang, Lin Gui, Yulan He, Wenjie Pei, and Ruifeng Xu. 2022. Multi-modal sarcasm detection via cross-modal graph convolutional network. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1767–1777, Dublin, Ireland. Association for Computational Linguistics.

Iván Martín-Fernández, Sergio Esteban-Romero, Fernando Fernández-Martínez, and Manuel Gil-Martín. 2025. Parameter-efficient adaptation of large vision—language models for video memorability prediction. *Sensors*, 25(6).

Mihaela-Georgiana Mihailescu. 2024. Never mess with the "memers": How meme creators are redefining contemporary politics. *Social Media + Society*, 10.

Zunaira Munawar. 2025. The role of political social media memes on shaping public opinion: A survey analysis. *Lead Sci Journal of Management, Innovation and Social Sciences*.

Shraman Pramanick, Shivam Sharma, Dimitar Dimitrov, Md. Shad Akhtar, Preslav Nakov, and Tanmoy Chakraborty. 2021. MOMENTA: A multimodal framework for detecting harmful memes and their targets. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4439–4455, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiaoye Qu, Mingyang Song, Wei Wei, Daizong Liu, Jianfeng Dong, and Yu Cheng. 2026. Mitigating multilingual hallucination in large vision-language models. *ACM Transactions on Multimedia Computing, Communications and Applications*, 22(5).

Fabian David Schmidt, Florian Schneider, Chris Biemann, and Goran Glavaš. 2025. MVL-SIB: A massively multilingual vision-language benchmark for cross-modal topical matching. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16285–16312, Vienna, Austria. Association for Computational Linguistics.

Siddhant Bikram Shah, Shuvam Shiwakoti, Maheep Chaudhary, and Haohan Wang. 2024. MemeCLIP: Leveraging CLIP representations for multimodal meme classification. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17320–17332, Miami, Florida, USA. Association for Computational Linguistics.

Aakash Singh, Deepawali Sharma, and V. K. Singh. 2024. Mimic: Misogyny identification in multimodal internet content in hindi-english code-mixed language. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

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, Kyiv. Association for Computational Linguistics.

Shardul Suryawanshi, Bharathi Raja Chakravarthi, Pranav Verma, Mihael Arcan, John P. 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).

Maria Tzelepi and Vasileios Mezaris. 2025. Improving multimodal hateful meme detection exploiting lmm-generated knowledge. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 202–211.

Tan Yue, Rui Mao, Heng Wang, Zonghai Hu, and Erik Cambria. 2023. Knowlenet: Knowledge fusion network for multimodal sarcasm detection. *Information Fusion*, 100:101921.

# A Cascade Prompts

**Level 1 prompt:**

```
Read the text in this meme
and look at the image.  Is
this 'Support/Praise' or
'Troll/Oppose'?  Answer
with just the label.
```

**Level 2 prompt (cascaded on predicted/gold L1 label):**

```
You identified this meme
as {level1_label}.  Based
on the image and text,
what is the specific Level
2 category?  Answer with
just the label.
```

# B Hyperparameters

| Parameter | Value |
|---|---|
| Backbone | Qwen2-VL-2B-Instruct |
| Quantization | 4-bit NF4 |
| Compute dtype | bfloat16 |
| LoRA rank / alpha / dropout | 16 / 32 / 0.05 |
| Target modules | q,k,v,o,gate,up,down_proj |
| Trainable params | 18.5M / 2.23B (0.83%) |
| Batch size / grad accum | 2 / 8 (eff. 16) |
| Max new tokens | 8 |
| Epochs | 3 |
| Training samples | 324 (balanced) |
| Attention implementation | SDPA |
| Hardware | 1× T4, 15.6GB VRAM |
| Training time | ∼3.7h/epoch (∼11h total) |

Table 3: Training hyperparameters.

# C Qualitative Error Categorization

We manually reviewed a sample of 15 Level 1 misclassifications and identified four recurring failure categories:

- **Sarcasm/irony read literally (6/15, 40%):** Memes quoting a political figure's self-congratulatory claim, undercut by an ironic follow-up line, scored as literal endorsement.
- **Reverse bias intense phrasing mistaken for hostility (4/15, 27%):** Exclamatory or combative language in genuinely supportive fan rhetoric misread as opposition.
- **Cultural/political knowledge gaps (2–3/15, ∼15–20%):** Cases requiring specific historical or intra-party knowledge (e.g. references to deceased political figures, party succession politics) underrepresented in general-purpose VLM pretraining.
- **Genuinely ambiguous (∼2/15, ∼13%):** Borderline cases even under manual, non-native-speaker review.

# 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.

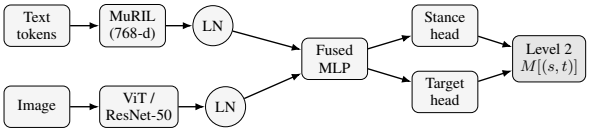



Figure 1: Factorized multimodal architecture. Text (MuRIL) and image (ViT or ResNet-50) are encoded separately, normalised per modality (LN), fused by an MLP, and read out by two binary heads. Level 2 is reconstructed from the (stance, target) pair via Eq. 1.

## 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—$2 \times 10^{-5}$ for the pretrained encoders and $1 \times 10^{-3}$ 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.

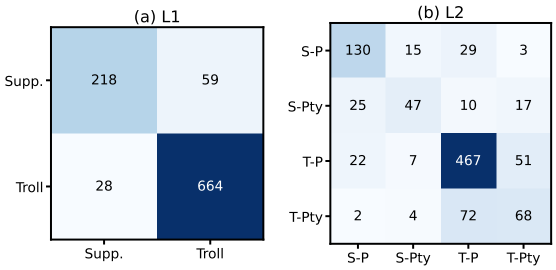



Figure 2: OOF confusion matrices: (a) Level 1, (b) Level 2 (S-P/S-Pty = Support person/party; T-P/T-Pty = Troll/Oppose Against person/party). Off-diagonal mass sits mainly between the two Troll/Oppose classes, confirming target-, not stance-axis, confusion drives residual error (§6).

## 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) | $2 \times 10^{-5}$ / $2 \times 10^{-5}$ |
| LR (fusion / heads) | $1 \times 10^{-3}$ |
| 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.

# Projukti Kavalan@MMLow 2026: Weighted Visual Frame and Cross-modal Fusion for Misogynistic Video Detection in Dravidian Language

**Palaniraj I***
BestWay Solution
Tamil Nadu, India
pazhaniraj@bestwaysolution.in

**Dwaipayan Dutta***
Dept. of Mathematics
Indian Institute of Space Science and Technology
Kerala, India
dwaipayandutta.25@res.iist.ac.in

## Abstract

Misogyny and hate content have become a serious issue in the current digital era. Such content predominantly manifests as short video clips, making automated detection mechanisms imperative for social media platforms. To effectively classify such videos, a multimodal setup is required, as video data contains image frames, audio clips, and text captions. The multimodal misogyny video classification task aims to combine all audio, visual, and extracted text information. Conventional methods typically aim to process this extracted information in parallel to reach a conclusion. Our proposed framework utilizes the pre-trained models DINOv2, wav2vec 2.0, and MuRiL-large to embed visual, audio, and text features, respectively. To mitigate overfitting on datasets of different sizes, we propose a dual-architecture approach: we utilize a regularized Gated Early-Fusion Network (EF-Net) for the highly constrained Tamil dataset, and a deeper Cross-Modal Temporal Attention Network (TA-Net) with an ensemble strategy for the Malayalam dataset. Ultimately, our framework achieved a 68.8% macro-F1 score for the Malayalam dataset and an 80.7% macro-F1 score for the Tamil dataset, securing Rank 1 in misogyny video classification for both languages.

## 1 Introduction

Social media platforms have amplified gender-biased hate, increasingly distributed via multimodal formats like short videos. While robust detection architectures exist for high-resource languages, models for Dravidian languages like Tamil and Malayalam—which have massive digital footprints—lag significantly behind, necessitating specialized multimodal approaches capable of identifying misogyny across speech, text, and visual streams.

## 2 Related Work

Early research in misogyny detection concentrated on unimodal text models (Cao et al., 2020). As harmful content migrated to multimedia, the community adopted multimodal strategies, such as bimodal fusion for images and captions (Jindal et al., 2024) or chain-of-thought reasoning for memes (Kumari et al., 2024).

In video-based detection, filtering noisy temporal streams is paramount. Wang et al. (2023) introduced a frame-weighting mechanism utilizing cross-modal attention to isolate informative visual frames against on-screen text, which Céspedes-Sarrias et al. (2025) expanded using audio. While Céspedes-Sarrias et al. (2025) relies on standard cross-modal attention across parallel streams, our framework directly regulates the visual sequence. We adapt the frame-weighting concept from Wang et al. (2023) but significantly extend it: instead of their additive fusion for a single semantic stream, we introduce an independent audio weight vector ($w_{\text{audio}}$) and employ a strict multiplicative consensus mechanism ($\odot$). This ensures video frames are emphasized only upon simultaneous audio, text, and visual agreement, drastically reducing noise in constrained datasets.

Crucially, while these models excel on massive datasets, they suffer catastrophic overfitting on low-resource languages. Recent shared tasks (Chakravarthi et al., 2021; G et al., 2025; Rajiakodi et al., 2025) emphasize that multimodal hate speech detection in Dravidian languages remains a critical challenge. Our framework addresses this gap by strictly constraining architecture capacity based on dataset size for the MMLow 2026 shared task.

## 3 Methodology

Given the extreme resource differences between the Tamil and Malayalam datasets, a "one-size-fits-all" model is sub-optimal. Instead, we extract robust tri-



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 76–85
November 9, 2026 ©2026 Association for Computational Linguistics

modal features and route them to specialized architectures: a highly-regularized Gated Early-Fusion Network (**Architecture A**) for the extremely constrained Tamil dataset, and a deeper Cross-Modal Temporal Attention Network (**Architecture B**) for the larger, balanced Malayalam dataset.

### 3.1 Data Preprocessing and Feature Extraction

Two different video datasets are being used for this work: 146 total videos for Tamil (120 train, 26 test) and 343 for Malayalam (303 train, 40 test). Raw videos were segmented into visual frames (Culjak et al., 2012), audio was separated into vocal tracks (Rouard et al., 2023), and transcripts were generated via Gemini Flash 3.5 (Team et al., 2023). We extracted 1024-dimensional embeddings using DINOv2 (Oquab et al., 2023) for visuals, MuRiL-Large (Khanuja et al., 2021) for text, and wav2vec 2.0 (Baevski et al., 2020) for audio.

### 3.2 Architecture A: Gated Early-Fusion Network

To prevent memorization on the highly constrained Tamil dataset, we designed a Gated Early-Fusion Network operating on globally pooled features $(F_v, F_a, F_t)$. Collapsing temporal sequences creates a strong informational bottleneck (regularizer). To suppress noise, we apply a learnable gate: $f_m = \mathrm{ReLU}(W_f F_m + b_f)$, $g_m = \sigma(W_g F_m + b_g)$, $\hat{F}_m = \mathrm{Dropout}(f_m \odot g_m)$, where $W_f \in \mathbb{R}^{64\times1024}$ compresses raw embeddings to 64 dimensions. Surviving features are concatenated $F_{fused} = [\hat{F}_v; \hat{F}_t; \hat{F}_a]$ and passed through a Residual MLP.

**High-Capacity Variant:** `EF-Net (High-Cap)` replaces the 64-dim bottleneck with a 512-dim projection (1536-dim concatenated) and a deeper MLP (0.4 dropout) to serve as a capacity-matched baseline counterpart.

### 3.3 Architecture B: Cross-Modal Temporal Attention Network

For the more balanced and contextually complex Malayalam dataset, we implemented a temporal attention architecture consisting of a Multimodal Information Encoding Module (MIE), a Visual Frame Weighting Module (VFW), and a Cross-Modal Transformer Fusion network (CMTF).

#### 3.3.1 Problem Formulation

Given dataset $\mathcal{D} = \{(\mathcal{V}_i, y_i)\}$, with $y_i \in \{0, 1\}$, each video contains three modalities: visual $V \in \mathbb{R}^{n\times d_v}$, audio $A \in \mathbb{R}^{8\times d_a}$, and text $T \in \mathbb{R}^{1\times d_t}$. To maintain strict temporal alignment, visual streams are uniformly interpolated to 30 frames, and audio to 8 steps. Transcripts are represented as a globally pooled vector.

#### 3.3.2 Multimodal Information Encoding Module (MIE)

Each representation is projected to a short dimension $d(= 64)$. The visual and audio representations are passed through a bidirectional GRU (Gated Recurrent Unit) model to encode information in both directions. Therefore visual encoder produces $F_v \in \mathbb{R}^{n\times d}$, audio encoder produces $F_a \in \mathbb{R}^{8\times d}$. For each video, we select the reference training clip with the same label whose visual representation has the highest cosine similarity with the video:

$$\mathcal{V}^+ = \{\mathcal{V}_{ref} \mid \arg\max_{V_{ref}} \mathrm{sim}(V, V_{ref}),\ y_V = y_{V_{ref}}\}$$

where $V$ is the visual representation of the video clip $\mathcal{V}$, and $V_{ref}$ is the visual representation of the reference video clip $\mathcal{V}_{ref}$. These positive visual references are encoded by the visual encoder and produce $F_v^+ \in \mathbb{R}^{n\times d}$. Moreover, the transcript representations pass through a linear projection and layer normalization layer, and so, the transcript encoder produces $F_t \in \mathbb{R}^{1\times d}$. Final encodings are: $F_v = \mathrm{Vis}(V)$, $F_v^+ = \mathrm{Vis}(V_{ref})$, $F_a = \mathrm{Aud}(A)$, and $F_t = \mathrm{Txt}(T)$.

#### 3.3.3 Visual Frame Weighting Module (VFW)

The core of this architecture adapts a multimodal frame-weighting mechanism originally proposed by (Wang et al., 2023). From their foundational framework, we reuse the visual self-weight ($w_{\text{self}}$), the text-based cross-modal weight ($w_{\text{text}}$), and the contrastive reference-context weight ($w_{\text{ctx}}$) derived from positive sample alignment. To elevate this to a robust trimodal architecture, we introduce a novel cross-modal video-audio weight vector ($w_{\text{audio}}$). Furthermore, rather than using an additive fusion as in (Wang et al., 2023), we propose a multiplicative consensus fusion via the Hadamard product ($\odot$). Prior to this operation, any multi-step temporal sequences (such as audio $F_a \in \mathbb{R}^{8\times d}$ or visual $F_v \in \mathbb{R}^{n\times d}$) serving as the context $c$ are first reduced to a single vector $c \in \mathbb{R}^d$ via an attention-based pooling layer. This pooling operation produces a unified context representation for subse-

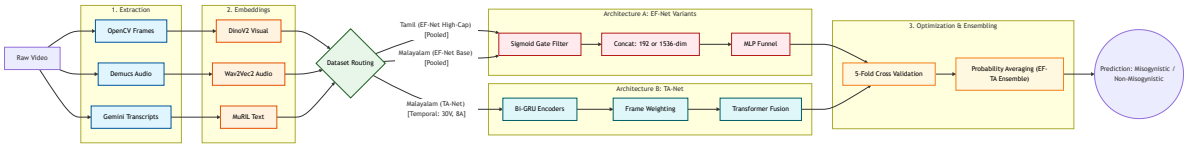



Figure 1: Overview of the proposed pipeline: text, visual, and audio features are encoded, four attention signals $(w_t, w_{self}, w_{ctx}, w_a)$ jointly weight the video frames, the top-k frames are selected, and a cross-modal transformer fuses them with text and audio to predict misogynistic vs. non-misogynistic content.

quent weight computation:

$$\mathrm{FW}(c, F_v) = \mathrm{softmax}\left(\frac{(W_q\, c)\,(W_k\, F_v)^\top}{\sqrt{d}}\right) \in \mathbb{R}^n. \tag{1}$$

The final temporal weights are derived via multiplicative fusion to extract the top-$K$ ($K = 4$) frames most relevant to the misogynistic context. Note that unlike the original MCA-WF implementation, we rely on standard scaled dot-product attention without a temperature hyperparameter ($\tau$), instead opting for a deterministic top-$K$ extraction:

$$w = \frac{w_{\text{self}} \odot w_{\text{text}} \odot w_{\text{audio}} \odot w_{\text{ctx}}}{\sum(w_{\text{self}} \odot w_{\text{text}} \odot w_{\text{audio}} \odot w_{\text{ctx}})}, \tag{2}$$

Finally, to fuse modalities, we add modality and positional embeddings to the top-$K$ frames ($F_k$), text ($F_t$), and audio ($F_a$): $Z_0 = [F_k + E^{(v)}_{mod}; F_t + E^{(t)}_{mod}; F_a + E^{(a)}_{mod}] + E_{pos}$. The combined sequence passes through $L = 1$ Transformer block, is globally averaged, and classified.

### 3.4 Training Objective

We optimize a composite loss: $\mathcal{L} = \mathcal{L}_{\text{focal}} + \lambda_{\text{ml}} \left\|F_v - F_v^+\right\|_F^2$, with $\lambda_{\text{ml}} = 0.1$. The focal loss addresses class imbalance dynamically per language. The metric-learning term pulls visual representations toward same-class positives to encourage intra-class compactness.

### 3.5 Training Regime and Probability Ensembling

To explicitly address severe class imbalances, we implemented a Dynamic Focal Loss that penalizes majority-class bias. All models are trained under a 5-fold cross-validation setup using AdamW (learning rate $10^{-4}$, weight decay $10^{-2}$) and batch size 8 for up to 100 epochs, with early stopping (patience 10) on validation macro-F1. A fixed random seed (42) controls for initialization variance. During inference, we average the softmax probabilities across the five folds. Ultimately, our final Tamil submissions relied heavily on Gated Early-Fusion variants, while Malayalam submissions utilized a mathematically averaged ensemble of Gated Early-Fusion and Temporal Attention to capture nuanced behaviors.

## 4 Results and Evaluation

Table 1: Performance on the Malayalam and Tamil test sets.

| Lang. | Method | Acc | M-Pre | M-Rec | M-F1 |
|---|---|---|---|---|---|
| Mal. | Baseline-MCA-WF | 0.600 | 0.639 | 0.612 | 0.583 |
| | Baseline-MMHSD | 0.675 | 0.720 | 0.685 | 0.665 |
| | TA-Net | 0.675 | 0.720 | 0.685 | 0.665 |
| | **EF-TA Ensemble** | **0.700** | **0.765** | **0.712** | **0.688** |
| Tam. | Baseline-MCA-WF | 0.808 | 0.810 | 0.808 | 0.807 |
| | Baseline-MMHSD | 0.808 | 0.810 | 0.808 | 0.807 |
| | EF-Net | 0.769 | 0.776 | 0.769 | 0.768 |
| | **EF-Net (High-Cap)** | **0.808** | **0.810** | **0.808** | **0.807** |

On the Malayalam test set, our EF-TA Ensemble significantly outperforms the MMHSD (Céspedes-Sarrias et al., 2025) and MCA-WF (Wang et al., 2023) baselines by 2.3 and 10.5 percentage points (Table 1), confirming the effectiveness of blending early multimodal fusion with temporal attention. Conversely, on the Tamil test set, all top models hit a ceiling Macro-F1 of 0.807. However, class-wise performance reveals these models make distinct precision-recall trade-offs despite identical macro scores.

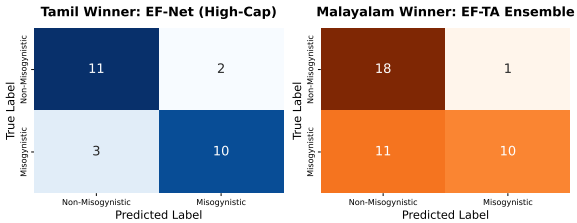



Figure 2: Confusion matrices for the winning architectures on Tamil and Malayalam test sets.

### 4.1 Cross-Lingual Performance Analysis

A distinct performance difference can be observed in Table 1 between the two languages. The best Macro-F1 obtained for Tamil data is 0.807, whereas the best Macro-F1 for Malayalam data is 0.688, representing a difference of 11.9 percentage points. A similar difference is observed for accuracy: the best Tamil result is 0.808, whereas for Malayalam data it is 0.700. This indicates that the proposed multimodal framework is substantially more effective on the Tamil test set. The observed language-dependent variation may be attributed to differences in dataset characteristics, linguistic properties, modality quality, class distributions, and the effectiveness of the extracted representations for each language. The lower Malayalam performance suggests that further investigation into language-specific representation learning and modality alignment is beneficial.

### 4.2 Discussion

On the Malayalam set ($n = 40$), the EF-TA Ensemble maximized precision (reducing false positives) at the expense of target-class recall (0.476). This trade-off naturally suppresses noise in constrained datasets, though improving recall remains a priority. On the Tamil set ($n = 26$), the baseline and HighCap variant tied on macro-F1 (0.807) but exhibited diverging operating points (Table 2); the baseline favored recall (84.6% vs. 76.9%) while EF-Net favored precision (83.3% vs. 78.6%).

Table 2: Class-wise Precision (P) and Recall (R) comparing the baselines with the proposed winners.

| Language / Model | Class 0 (P / R) | Class 1 (P / R) |
|---|---|---|
| **Tamil (Constrained)** | | |
| - Baseline (MMHSD) | 0.833 / 0.769 | 0.786 / **0.846** |
| - EF-Net (High-Cap) | 0.786 / **0.846** | **0.833** / 0.769 |
| **Malayalam (Balanced)** | | |
| - Baseline (MMHSD) | 0.607 / 0.895 | 0.833 / 0.476 |
| - **EF-TA Ensemble** | **0.621 / 0.947** | **0.909 / 0.476** |

Analyzing local 5-fold validation stability, our Malayalam Temporal Attention network achieved a mean Macro-F1 of $0.917 \pm 0.030$, demonstrating tighter variance than the MCA-WF baseline ($0.887 \pm 0.050$). The Tamil dataset exhibited predictably high variance ($\approx \pm 0.087$) across all approaches. Ultimately, evaluating on such constrained sets ($n = 26$, $n = 40$) is challenging, as percentage gains resolve to single-video shifts and aggressive regularization forces architectures toward similar decision boundaries. Nevertheless, our framework successfully extracted the incremental advantages necessary to secure Rank 1, indicating strong potential for deployment on larger, unconstrained datasets.

## 5 Conclusion

We introduced a dual-architecture framework for detecting multimodal misogyny in Dravidian languages. To mitigate catastrophic overfitting on highly constrained datasets, we matched architecture capacity to dataset size. A heavily constrained Gated Early-Fusion Network generalized on the tiny Tamil dataset, while the larger Malayalam dataset supported a deeper Cross-Modal Temporal Attention Network. This empirically driven framework secured Rank 1 across both languages. Because our models were optimized for strict constraints, they favored precision over recall. Future work will expand training data and explore end-to-end joint training to improve recall.

## Ethical Considerations

Automated content moderation systems carry inherent risks. Given severe data constraints and the subjectivity of hate speech, our models are susceptible to misclassifications—risking unjust censorship (false positives) and the unchecked spread of harmful content (false negatives). Because our models heavily favor precision, some misogynistic content may evade detection. Furthermore, the limited dataset scale prevents us from guaranteeing the absence of latent socio-cultural biases. We strongly advocate that these models serve strictly as assistive tools for human moderators.

## Acknowledgements

Both authors contributed equally to this work.

## Code Availability

Our complete code is publicly available at: https://github.com/Projukti-Kavalan/MMLow-2026

## References


Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Rui Cao, Roy Ka-Wei Lee, and Tuan-Anh Hoang. 2020. Deephate: Hate speech detection via multi-faceted text representations. In *Proceedings of the 12th ACM Conference on Web Science*, WebSci '20, page 11–20, New York, NY, USA. Association for Computing Machinery.

Berta Céspedes-Sarrias, Carlos Collado-Capell, Pablo Rodenas-Ruiz, Olena Hrynenko, and Andrea Cavallaro. 2025. Mm-hsd: Multi-modal hate speech detection in videos. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, page 2546–2555, New York, NY, USA. Association for Computing Machinery.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan R L, John Philip McCrae, and Elizabeth Sherly. 2021. Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 133–145, Kyiv. Association for Computational Linguistics.

Ivan Culjak, David Abram, Tomislav Pribanic, Hrvoje Dzapo, and Mario Cifrek. 2012. A brief introduction to OpenCV. In *2012 Proceedings of the 35th International Convention MIPRO*, pages 1725–1730.

Jyothish Lal G, Premjith B, Bharathi Raja Chakravarthi, Saranya Rajiakodi, Bharathi B, Rajeswari Natarajan, and Ratnavel Rajalakshmi. 2025. Overview of the shared task on multimodal hate speech detection in Dravidian languages: DravidianLangTech@NAACL 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 114–122, Acoma, The Albuquerque Convention Center, Albuquerque, New Mexico. Association for Computational Linguistics.

Nitesh Jindal, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Sajeetha Thavareesan, Saranya Rajiakodi, and Bharathi Raja Chakravarthi. 2024. Mistra: Misogyny detection through text–image fusion and representation analysis. *Natural Language Processing Journal*, 7:100073.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: multilingual representations for indian languages (2021). *arXiv preprint arXiv:2103.10730*.

Gitanjali Kumari, Kirtan Jain, and Asif Ekbal. 2024. M3Hop-CoT: Misogynous meme identification with multimodal multi-hop chain-of-thought. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22105–22138, Miami, Florida, USA. Association for Computational Linguistics.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Saranya Rajiakodi, Bharathi Raja Chakravarthi, Shunmuga Priya Muthusamy Chinnan, Ruba Priyadharshini, Rajameenakshi J, Kathiravan P, Rahul Ponnusamy, Bhuvaneswari Sivagnanam, Paul Buitelaar, Bhavanimeena K, Jananayagam V, and Kishore Kumar Ponnusamy. 2025. Findings of the shared task on abusive Tamil and Malayalam text targeting women on social media: DravidianLangTech@NAACL 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 671–681, Acoma, The Albuquerque Convention Center, Albuquerque, New Mexico. Association for Computational Linguistics.

Simon Rouard, Francisco Massa, and Alexandre Défossez. 2023. Hybrid transformers for music source separation. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Yuqing Wang, Zhuang Qi, Xiangxian Li, Jinxing Liu, Xiangxu Meng, and Lei Meng. 2023. Multi-channel attentive weighting of visual frames for multimodal video classification. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.


## A Appendix: Training Dynamics

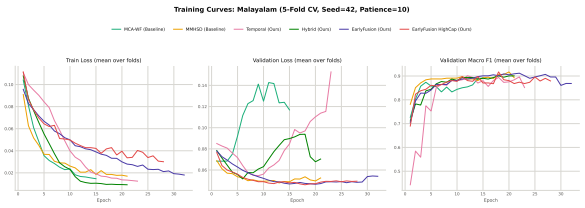



Figure 3: Training/validation loss and validation macro-F1 across epochs for Malayalam (5-fold CV, mean over folds).

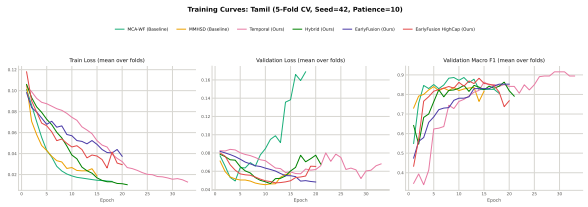



Figure 4: Training/validation loss and validation macro-F1 across epochs for Tamil (5-fold CV, mean over folds).

# B Appendix B: Architectural Decision-Making and Ablations

To empirically justify the Empirical Architecture Selection framework described in Section 3, extensive hyperparameter grid searches were conducted on the local validation splits created from the released training dataset. Furthermore, to evaluate the distinct contributions of the data streams, this validation process inherently serves as our modality ablation study. Whereas the existing baseline architectures—Multi-Modal Hate Speech Detection (MMHSD) and Multi-channel Attentive Weighting of Visual Frames (MCA-WF)—utilize restricted modality combinations, our 'Proposed 3-Way' models fully integrate the trimodal features (Audio, Visual, and Text). While a strict unimodal Text-Only baseline was not evaluated due to the shared task's submission limits, the performance gap between the restricted-modality baselines and our trimodal architectures demonstrates the added value of the visual and audio streams, as detailed in Table 3.

## B.1 Text Encoder Selection

We systematically evaluated two state-of-the-art multilingual text encoders: E5-Large and MuRiL-Large (optimized for Indian languages). Across both datasets and irrespective of whether a Pooled (Early Fusion) or Temporal (Cross-Modal Attention) architecture was utilized, the MuRiL-Large embeddings consistently outperformed E5-Large. Notably, on the Tamil dataset, the MuRiL-Large Temporal setup with our proposed 3-way routing achieved a validation macro-F1 of 1.000 (zero false positives/negatives on the local fold), whereas E5-Large capped at 0.956. For Malayalam, the MuRiL-Large Pooled configuration achieved a local validation macro-F1 of 0.917, significantly surpassing E5-Large's 0.852. Consequently, MuRiL-Large was selected as the definitive text extractor for all final models. Note that the perfect validation score of 1.000 for the Temporal model on Tamil is a small-sample validation artifact representing catastrophic overfitting to a single 24-video fold, which justified our shift to the constrained EF-Net.

## B.2 Hyperparameter Grid Search and Regularization

The severe disparity in dataset sizes required careful tuning of our regularization strategies to prevent catastrophic memorization, particularly on the Tamil dataset.

**Standard Architectures (EF-Net & TA-Net):** For our core architectures—the Gated Early-Fusion Network (EF-Net) and the Cross-Modal Temporal Attention Network (TA-Net)—an extensive grid search revealed that aggressive compression and regularization were universally necessary across both languages to maintain out-of-sample robustness. The optimal configuration for both Tamil and Malayalam utilized a unified latent dimension of $DIM = 64$, an extreme dropout rate of 0.7, and a high weight decay of 0.01. This massive informational bottleneck proved highly effective at suppressing dataset-specific noise.

**High-Capacity Variant:** To explicitly test if the Tamil dataset could support deeper representations without overfitting, we evaluated the `EF-Net (High-Cap)` variant. This model was configured with a much wider latent dimension of $DIM = 512$ and a more forgiving dropout of 0.4. While this variant successfully reached parity with the baselines on the Tamil test set, the standard heavily-regularized architectures proved more stable across both languages on average, as shown by the variance metrics in Table 4.

# C Appendix C: Final Competition Submissions

During the official final phase of the competition, we submitted only our top 3 models per language based on their validation performance. However, to provide a comprehensive scientific analysis for this paper, Table 5 reports the test set results for an expanded suite of 9 runs per language, representing a systematic progression from established baselines to our specialized multimodal ensembles. This clarifies that our Rank 1 placement was achieved within a strict 3-submission limit, and the expanded 9-run post-evaluation confirms that the EF-Net (High-Cap) variant and the EF-TA Ensemble provide the most robust generalization.

Table 3: Comprehensive modality and architecture ablation results evaluated on the local validation split. This demonstrates modality ablation: while baselines rely on limited modalities, the 'Proposed 3-Way' achieves performance gains via full trimodal integration (Audio, Visual, Text). Note that the 1.000 score on Tamil is a localized small-sample validation artifact.

| Language | Text Encoder | Input Type | Model Architecture | Val Accuracy | Val Macro-F1 |
|---|---|---|---|---|---|
| Tamil | E5-Large | Pooled | MMHSD | 0.833 | 0.829 |
| Tamil | E5-Large | Pooled | MCA-WF | 0.958 | 0.955 |
| Tamil | E5-Large | Pooled | Proposed 3-Way | 0.917 | 0.911 |
| Tamil | E5-Large | Temporal | MMHSD | 0.875 | 0.863 |
| Tamil | E5-Large | Temporal | MCA-WF | 0.875 | 0.873 |
| Tamil | E5-Large | Temporal | Proposed 3-Way | 0.958 | 0.956 |
| Tamil | MuRiL-Large | Pooled | MMHSD | 0.917 | 0.914 |
| Tamil | MuRiL-Large | Pooled | MCA-WF | 0.958 | 0.956 |
| Tamil | MuRiL-Large | Pooled | Proposed 3-Way | 0.917 | 0.914 |
| Tamil | MuRiL-Large | Temporal | MMHSD | 0.875 | 0.863 |
| Tamil | MuRiL-Large | Temporal | MCA-WF | 0.875 | 0.869 |
| Tamil | MuRiL-Large | Temporal | Proposed 3-Way | 1.000 | 1.000 |
| Malayalam | E5-Large | Pooled | MMHSD | 0.853 | 0.853 |
| Malayalam | E5-Large | Pooled | MCA-WF | 0.853 | 0.852 |
| Malayalam | E5-Large | Pooled | Proposed 3-Way | 0.853 | 0.852 |
| Malayalam | E5-Large | Temporal | MMHSD | 0.836 | 0.836 |
| Malayalam | E5-Large | Temporal | MCA-WF | 0.787 | 0.787 |
| Malayalam | E5-Large | Temporal | Proposed 3-Way | 0.869 | 0.869 |
| Malayalam | MuRiL-Large | Pooled | MMHSD | 0.902 | 0.901 |
| Malayalam | MuRiL-Large | Pooled | MCA-WF | 0.918 | 0.918 |
| Malayalam | MuRiL-Large | Pooled | Proposed 3-Way | 0.918 | 0.918 |
| Malayalam | MuRiL-Large | Temporal | MMHSD | 0.836 | 0.836 |
| Malayalam | MuRiL-Large | Temporal | MCA-WF | 0.869 | 0.869 |
| Malayalam | MuRiL-Large | Temporal | Proposed 3-Way | 0.869 | 0.869 |

Table 4: Mean local validation Macro-F1 scores and standard deviations across 5 folds for our proposed architectures and baselines (utilizing MuRiL-Large embeddings). The variance highlights the stability of our models on Malayalam and the extreme volatility inherent to the tiny Tamil dataset.

| Language | Model | Macro-F1 (Mean $\pm$ SD) |
|---|---|---|
| Malayalam | Baseline: MCA-WF | $0.887 \pm 0.050$ |
| Malayalam | Baseline: MMHSD | $0.930 \pm 0.026$ |
| Malayalam | Proposed: Early Fusion | $0.917 \pm 0.060$ |
| Malayalam | Proposed: Temporal (TA) | $0.917 \pm 0.030$ |
| Tamil | Baseline: MCA-WF | $0.901 \pm 0.085$ |
| Tamil | Baseline: MMHSD | $0.868 \pm 0.086$ |
| Tamil | Proposed: Temporal (TA) | $0.900 \pm 0.085$ |
| Tamil | Proposed: EF-Net (High-Cap) | $0.887 \pm 0.087$ |

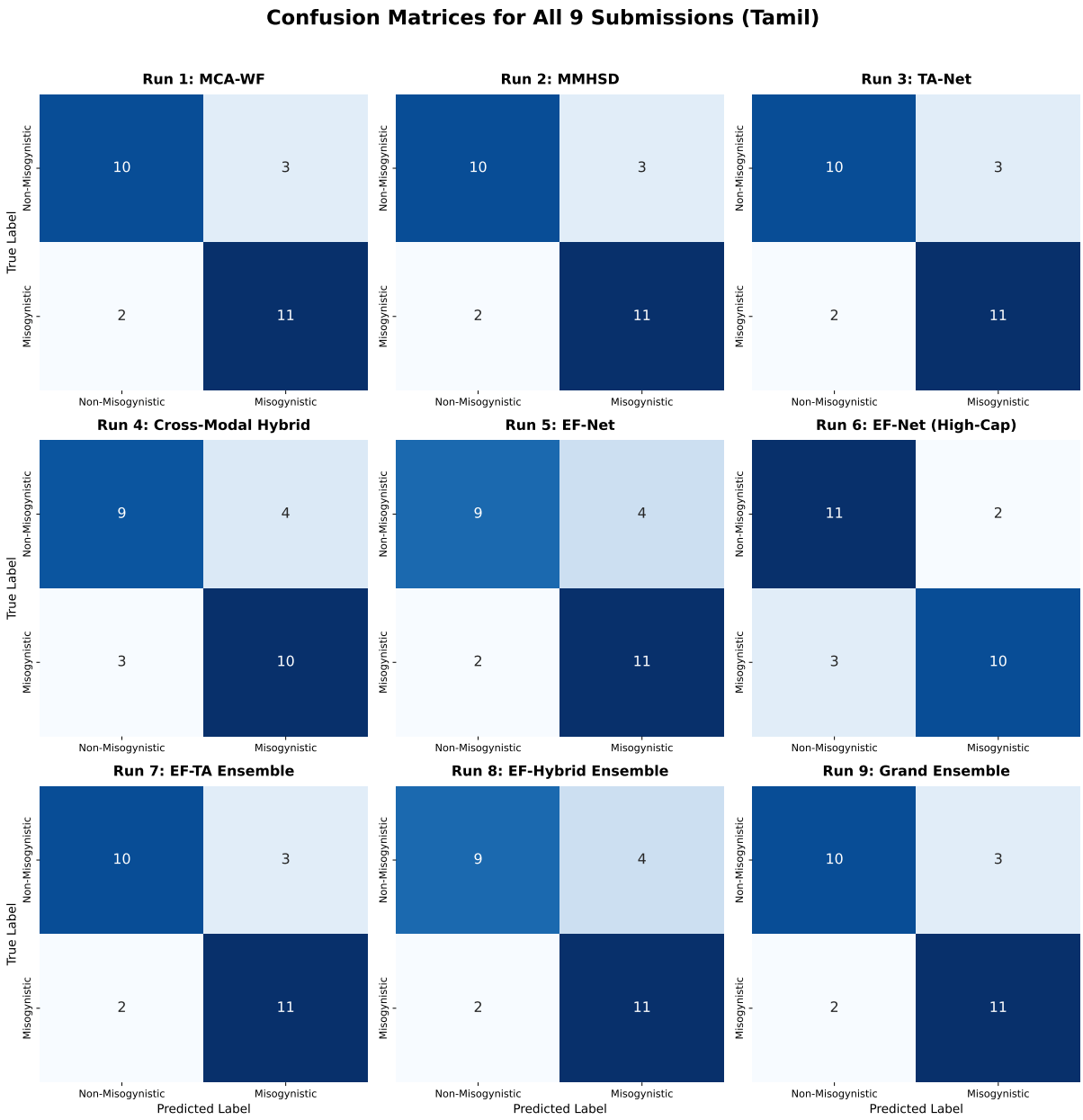



Figure 5: Detailed confusion matrices for all 9 models submitted for the Tamil dataset, demonstrating the evolution of predictive behavior.

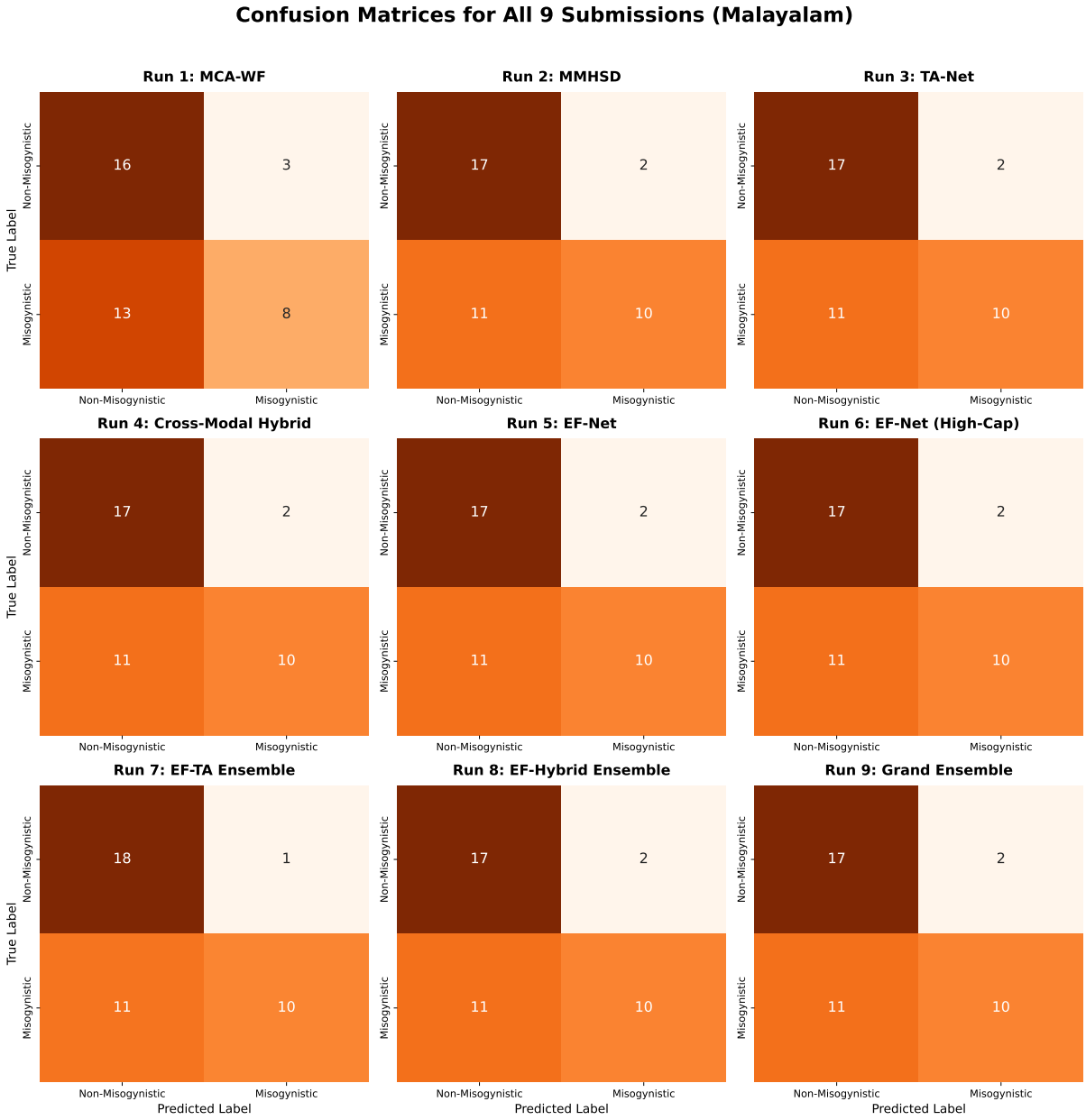



Figure 6: Detailed confusion matrices for all 9 models submitted for the Malayalam dataset, demonstrating the evolution of predictive behavior.

Table 5: Results of all 9 experimental runs submitted to the final evaluation phase for both datasets.

| **Language** | **Model Architecture** | **Accuracy** | **Macro-Precision** | **Macro-Recall** | **Macro-F1** |
|---|---|---|---|---|---|
| Tamil | MCA-WF | 0.808 | 0.810 | 0.808 | 0.807 |
| Tamil | MMHSD | 0.808 | 0.810 | 0.808 | 0.807 |
| Tamil | TA-Net | 0.808 | 0.810 | 0.808 | 0.807 |
| Tamil | Cross-Modal Hybrid | 0.731 | 0.732 | 0.731 | 0.730 |
| Tamil | EF-Net | 0.769 | 0.776 | 0.769 | 0.768 |
| Tamil | EF-Net (High-Cap) | 0.808 | 0.810 | 0.808 | 0.807 |
| Tamil | EF-TA Ensemble | 0.808 | 0.810 | 0.808 | 0.807 |
| Tamil | EF-Hybrid Ensemble | 0.769 | 0.776 | 0.769 | 0.768 |
| Tamil | Grand Ensemble | 0.808 | 0.810 | 0.808 | 0.807 |
| Malayalam | MCA-WF | 0.600 | 0.639 | 0.612 | 0.583 |
| Malayalam | MMHSD | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | TA-Net | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | Cross-Modal Hybrid | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | EF-Net | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | EF-Net (High-Cap) | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | EF-TA Ensemble | 0.700 | 0.765 | 0.712 | 0.688 |
| Malayalam | EF-Hybrid Ensemble | 0.675 | 0.720 | 0.685 | 0.665 |
| Malayalam | Grand Ensemble | 0.675 | 0.720 | 0.685 | 0.665 |

# KEC@MMLow 2026: Multimodal Fusion for Hierarchical Tamil Political Meme Classification

**Asvitha RM and Amirthamozhi VS**
Department of Computer Science and Engineering
Kongu Engineering College
Erode, Tamil Nadu, India

## Abstract

Political memes are inherently multimodal, combining visual cues, embedded text, and code-mixed language to express political stance and target specific political entities. This makes automated analysis particularly challenging in low-resource settings such as Tamil. We present the KEC system for the MMLow 2026 shared task at AACL-IJCNLP 2026, which requires hierarchical classification of Tamil political memes at two levels: stance identification and political target identification. Our approach combines XLM-RoBERTa and Vision Transformer representations with OCR-derived textual information, multimodal fusion, and bidirectional cross-modal attention. We further introduce hierarchy-aware prediction in which Level-2 target identification is conditioned on the Level-1 stance distribution, together with class-weighted objectives, adaptive task weighting, and hierarchical consistency regularization. A controlled six-stage ablation study and three-seed stability analysis are conducted on the validation split. The final hierarchical model obtains a Level-1 macro-F1 of $78.77 \pm 1.21\%$ and Level-2 macro-F1 of $46.07 \pm 3.23\%$, with a joint exact-match accuracy of $55.40 \pm 2.34\%$ across seeds 42, 123, and 2024. Our analysis indicates that visual information is particularly beneficial for stance classification, whereas multimodal fusion plays an important role in target identification, while OCR augmentation and hierarchical conditioning introduce performance trade-offs at Level 2.

## 1 Introduction

Political memes have become an important medium for expressing opinions, satire, support, and opposition on social media. Unlike conventional text-based content, memes communicate through interactions among visual elements, written text, symbols, and context-dependent cues. This multimodal nature makes interpretation challenging, particularly in low-resource and code-mixed languages such as Tamil, where memes frequently combine Tamil, English, Romanized Tamil, informal spellings, and text embedded within images.

The MMLow 2026 shared task at AACL-IJCNLP 2026 (Rajiakodi et al., 2026) addresses this challenge through hierarchical classification of Tamil political memes. The task requires systems to first determine the stance expressed by a meme at **Level 1** (*Support/Praise* or *Troll/Oppose*) and then identify its political target at **Level 2**. These decisions are closely coupled, as target interpretation can depend on stance polarity, while textual and visual evidence may contribute differently to each level.

In this study, team **KEC** presents a multimodal framework combining XLM-RoBERTa for code-mixed text, a Vision Transformer for visual information, and OCR-derived text extracted from meme images. We investigate both feature concatenation and bidirectional cross-modal attention, adapting the pretrained encoders using **Low-Rank Adaptation (LoRA)**. The final model incorporates hierarchical conditioning by using the Level-1 stance distribution for Level-2 prediction and jointly optimizes both objectives with class balancing, **homoscedastic uncertainty weighting**, and hierarchical consistency regularization. Our controlled experiments further examine the effects of visual information, OCR augmentation, multimodal fusion, cross-modal attention, and hierarchical conditioning through a six-stage ablation and three-seed stability analysis. [1]

## 2 Task and Dataset

The MMLow 2026 shared task performs hierarchical classification of Tamil political memes. Level 1 classifies stance as *Support/Praise* or *Troll/Oppose*, while Level 2 identifies the political target. The official Tamil dataset (Rajiakodi et al., 2026) con-

[1] Code is available on GitHub.



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 86–90
November 9, 2026 ©2026 Association for Computational Linguistics

tains 969 training samples and 244 unlabeled test samples.

### 2.1 Data Preparation

A data-quality audit excluded five hierarchy-inconsistent samples, leaving 964 instances; six caption-missing samples were retained using OCR fallback. The cleaned data were split into 819 training and 145 validation samples using an 85:15 stratified split on joint Level-1/Level-2 labels, with text normalized using Unicode NFC.

### 2.2 Evaluation

The shared task evaluates classification performance using macro-averaged precision, recall, and F1 score across the classes. Accordingly, we use macro-F1 as the primary metric for evaluating the two classification levels, while also reporting the joint exact-match accuracy for the hierarchical prediction.

## 3 Methodology

Our system is a configurable multimodal architecture that combines textual, visual, and OCR-derived information for hierarchical meme classification. The same architecture is used throughout a controlled six-stage ablation, with components introduced progressively to isolate their contributions.

### 3.1 Text and OCR Representation

We use XLM-RoBERTa-base (Conneau et al., 2020) to encode the Tamil–English code-mixed textual content. Input sequences are truncated or padded to a maximum length of 160 tokens. Captions are Unicode-normalized using NFC normalization before tokenization.

Since important information may be embedded in meme images rather than captions, we use OCR as an auxiliary textual source, particularly for names, slogans, and target-related text. We employ EasyOCR (JaidedAI, 2023) with Tamil and English models, discarding detections below 0.4 confidence and removing consecutive duplicates. OCR text is combined with the normalized caption using a `[SEP]` marker, while caption-missing samples use OCR alone. Although OCR can introduce noise from stylized, low-resolution, and code-mixed text, retaining it allows us to quantify this trade-off explicitly in our ablation study.

To adapt the pretrained language encoder efficiently to the small training set, we apply Low-Rank Adaptation (LoRA) (Hu et al., 2022) to the query and value projections of its attention layers, using rank $r = 8$, scaling factor $\alpha = 16$, and dropout 0.1. Across the two pretrained encoders, LoRA leaves approximately 1.09% of the total model parameters trainable, reducing the number of updated parameters while retaining task-specific adaptation.

### 3.2 Visual Representation

Visual features are extracted using a pretrained ViT-Base/16 encoder (Dosovitskiy et al., 2021). Images are converted to RGB and resized to $224 \times 224$ using aspect-ratio-preserving resize-with-padding, thereby avoiding distortion of faces, logos, political symbols, and embedded text. During training, light brightness and contrast perturbations and a small random crop are applied; validation uses deterministic resizing and normalization.

The ViT encoder is also adapted using LoRA with $r = 8$, $\alpha = 16$, and dropout 0.1, targeting its transformer $qkv$ projections.

### 3.3 Multimodal Fusion

We evaluate direct feature concatenation and bidirectional cross-modal attention based on multi-head attention (Vaswani et al., 2017). For cross-attention, text and image tokens are projected into a shared 512-dimensional space and processed by separate 8-head attention modules in both directions. Each path uses residual connections and layer normalization, followed by mean pooling and concatenation.

### 3.4 Hierarchical Prediction

The fused representation is passed to separate prediction heads for the two task levels. The Level-1 head uses a 256-dimensional hidden layer with GELU activation and dropout of 0.2 to predict stance. In the final hierarchical configuration, the resulting Level-1 probability distribution is concatenated with the fused representation and supplied to the Level-2 head:

$$p_1 = softmax(W_1 h + b_1), \qquad h_2 = [h; p_1]. \tag{1}$$

The Level-2 head uses the same 256-dimensional hidden layer, GELU activation, and dropout to predict the political target.

### 3.5 Hierarchical Training Objective

Both levels use class-weighted cross-entropy, with class weights computed from the training partition. For the final hierarchical model, the two task losses are adaptively weighted using homoscedastic uncertainty (Kendall et al., 2018). Let $\mathcal{L}_1$ and $\mathcal{L}_2$

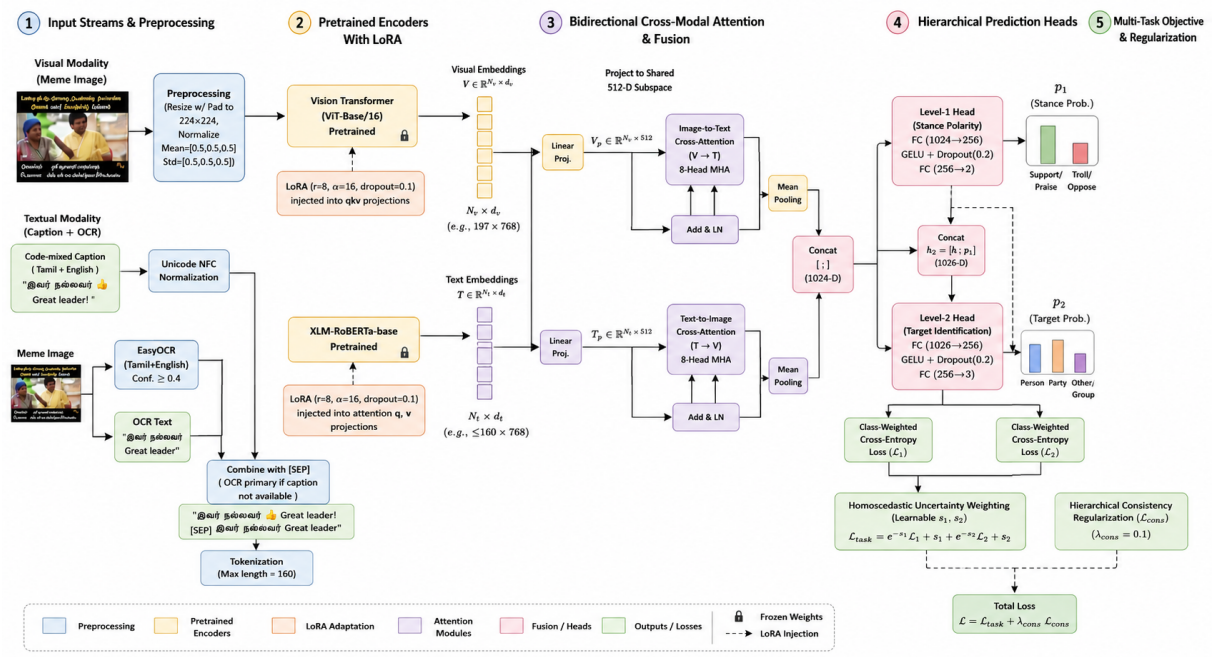



Figure 1: Overview of the proposed hierarchy-aware multimodal framework, combining LoRA-adapted XLM-RoBERTa and ViT-Base/16, bidirectional cross-modal attention, hierarchical conditioning, and uncertainty-weighted multi-task learning.

denote the Level-1 and Level-2 losses, and $s_1, s_2$ their learnable log-variance parameters. The task objective is

$$\mathcal{L}_{\text{task}} = e^{-s_1}\mathcal{L}_1 + s_1 + e^{-s_2}\mathcal{L}_2 + s_2. \quad (2)$$

We additionally apply a hierarchical consistency penalty to discourage Level-2 probability mass from being assigned to targets incompatible with the corresponding Level-1 class. The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}}, \quad (3)$$

where $\lambda_{\text{cons}} = 0.1$.

### 3.6 Optimization

All experiments use AdamW (Loshchilov and Hutter, 2019) ($2 \times 10^{-5}$ learning rate, 0.01 weight decay) for up to 12 epochs, with 10% warm-up, cosine decay, gradient clipping at 1.0, and mixed precision. Early stopping monitors validation Level-2 macro-F1 with patience 4; the best checkpoint is restored before evaluation. Because selection is based on Level-2 performance, it does not necessarily maximize Level-1 performance, which is reported separately.

## 4 Experiments and Results

### 4.1 Experimental Setup

Experiments use the cleaned 819/145 training–validation split with seed 42. We evaluate six progressively modified configurations from unimodal models through multimodal fusion, OCR, cross-modal attention, and hierarchical prediction. Macro-F1 is the primary metric at both levels; joint exact-match accuracy is also reported.

### 4.2 Ablation Study

| Configuration | L1 Macro-F1 | L2 Macro-F1 |
|---|---|---|
| A – Text-only | 0.2968 | 0.3015 |
| B – Image-only | 0.7654 | 0.3325 |
| C – Multimodal Concatenation | 0.7384 | **0.5181** |
| D – Multimodal + OCR | 0.7486 | 0.4458 |
| E – Cross-attention + OCR | 0.7851 | 0.3889 |
| F – Final Hierarchical | **0.7860** | 0.4448 |

Table 1: Single-run ablation results on the validation set using seed 42. Macro-F1 is reported separately for Level 1 and Level 2.

We conducted a six-stage ablation on the same validation split with seed 42, progressively adding visual information, multimodal concatenation, OCR, bidirectional cross-modal attention, and hierarchical conditioning with uncertainty-weighted training. Configuration F represents the complete hierarchy-aware system, rather than the configuration with the highest Level-2 score: C achieves the best single-run Level-2 macro-F1 (0.5181), whereas F achieves the best Level-1 macro-F1 (0.7860).

Visual information benefits Level 1, with macro-F1 increasing from 0.2968 (text-only) to 0.7654

| Evaluation | Level-1 Macro-F1 | Level-2 Macro-F1 | Joint/Final |
|---|---|---|---|
| Our 3-seed validation | $78.77 \pm 1.21\%$ | $46.07 \pm 3.23\%$ | $55.40 \pm 2.34\%$ |
| Official shared-task | — | 0.4329 | 0.4329 |

Table 2: Internal three-seed validation results and official shared-task evaluation for KEC Run 1.

(image-only) and 0.7860 (F). For Level 2, concatenation achieves 0.5181, while OCR and cross-modal attention reduce performance to 0.4458 and 0.3889; hierarchical conditioning reaches 0.4448. The cleaned dataset contains 690 Troll/Oppose and 274 Support/Praise samples at Level 1, and 544 Troll/Oppose Against Person, 176 Support for person, 146 Troll/Oppose Against Party, and 98 Support for party at Level 2. Six samples lack captions; the text-only configuration uses the original text without OCR. Its macro-F1 of 0.3015 exceeds the Level-2 majority-class reference of 0.1806.

We also evaluated the consistency penalty with the same architecture, split, and seed. With $\lambda_{\text{cons}} = 0.1$, Level-1 macro-F1, Level-2 macro-F1, and joint exact-match accuracy were 0.7860, 0.4448, and 0.5793; without the penalty, they were 0.7823, 0.4176, and 0.5379, respectively. Paired permutation tests (10,000 permutations) and paired-bootstrap 95% confidence intervals on the 145-sample split gave a Level-1 difference of 0.0037 (CI: [-0.0204, 0.0326], $p = 1.0000$) and a Level-2 difference of 0.0272 (CI: [-0.0410, 0.0955], $p = 0.4396$). An exact McNemar test for joint exact-match accuracy gave $p = 0.3269$. These results do not establish statistically significant improvements from the consistency penalty on this validation split.

### 4.3 Multi-Seed Stability

We evaluated the final hierarchical model with seeds 42, 123, and 2024. Table 2 reports the mean and standard deviation alongside the official KEC Run 1 evaluation, which reports Level-2 score 0.4329 but no Level-1 score. The submission ranked 7th.

## 5 Discussion and Error Analysis

### 5.1 Main Findings

Stance classification benefits primarily from visual cues, whereas target identification requires complementary textual and visual evidence. The weaker Level-2 results with OCR and cross-modal attention suggest that additional multimodal processing can introduce noise in this low-resource setting, while greater seed variability indicates higher sensitivity to initialization.

### 5.2 Error Analysis

Fine-grained target identification remains challenging due to class imbalance and ambiguous textual–visual evidence. Person–party ambiguity is recurring: a meme may display a politician while targeting the associated party. OCR also adds noise for stylized, low-resolution, and code-mixed text, reducing Level-2 macro-F1 from 0.5181 to 0.4458.

We analyze the final model on the 145-sample validation split; Table 3 reports class-wise Level-2 precision, recall, and F1.

| Level-2 Class | Precision | Recall | F1 |
|---|---|---|---|
| Support for party | 0.500 | 0.133 | 0.211 |
| Support for person | 0.484 | 0.577 | 0.526 |
| Troll/Oppose Against Party | 0.292 | 0.318 | 0.304 |
| Troll/Oppose Against Person | 0.721 | 0.756 | 0.738 |

Table 3: Class-wise Level-2 precision, recall, and F1 on the validation split.

Troll/Oppose Against Person has the highest F1 (0.738), while Support for party is most difficult (0.211). Table 4 shows confusion among support classes and between the two Troll/Oppose targets.

| True / Pred. | Support party | Support person | Troll party | Troll person |
|---|---|---|---|---|
| Support for party | 2 | 8 | 2 | 3 |
| Support for person | 2 | 15 | 2 | 7 |
| Troll/Oppose Against Party | 0 | 1 | 7 | 14 |
| Troll/Oppose Against Person | 0 | 7 | 13 | 62 |

Table 4: Level-2 confusion matrix on the 145-sample validation split.

## 6 Limitations and Conclusion

### 6.1 Limitations

Our study is constrained by the small, imbalanced training set and the noise introduced by OCR on stylized, low-resolution and code-mixed text. Validation results may also not generalize to memes from different periods, platforms or linguistic settings.

### 6.2 Conclusion

We presented the KEC system for the MMLow 2026 shared task, combining multilingual text, visual features, OCR, multimodal fusion, and hierarchy-aware learning for Tamil political meme classification. Visual information was most effective for Level-1 stance prediction, while direct multimodal concatenation gave the strongest Level-2 target identification. The multi-seed results showed stable stance performance but greater variability for target identification, highlighting the challenges of multimodal learning in low-resource, code-mixed political meme data.

## References


Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451. 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*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491.

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

Saranya Rajiakodi, Shunmuga Priya Muthusamy Chinnan, B Premjith, CN Subalalitha, Rahul Ponnusamy, Anshid K A, Bhuvaneswari Sivagnanam, V Jananayagan, N Ragavan, P Santhini, 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*. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.


## A Additional Implementation Details

The experiments use XLM-RoBERTa-base and ViT-Base/16 with LoRA adaptation. For both encoders, LoRA uses rank $r = 8$, scaling factor $\alpha = 16$, and dropout $0.1$. Text sequences are limited to 160 tokens, while meme images are resized to $224 \times 224$ using aspect-ratio-preserving padding with bilinear interpolation.

OCR is performed using EasyOCR with Tamil and English recognition models, discarding detections with confidence below 0.4. Cross-modal attention uses 512-dimensional projections and eight attention heads. Both prediction heads use a 256-dimensional hidden layer with GELU activation and dropout 0.2.

Training uses AdamW with learning rate $2 \times 10^{-5}$ and weight decay 0.01, for up to 12 epochs with 10% warm-up and cosine decay. Training and validation use a batch size of 16, with 145 samples in the validation split. Gradient norms are clipped to 1.0, mixed-precision training is enabled, and early stopping monitors validation Level-2 macro-F1 with patience four. The controlled ablations use random seed 42, while the stability analysis uses seeds 42, 123, and 2024. Experiments were run on an NVIDIA GeForce RTX 3050 Laptop GPU; the six ablation runs required approximately 61 minutes in total.

# CPSLab2@MMLow 2026: A Hierarchical Multimodal Framework with Stratified K-Fold Training for Political Meme Classification

**Anika Srivastava[1,*], Vishal Kumar[1,†], Harshita Sharma[1,‡], Nitisha Aggarwal[1,§], Geetika Jain Saxena[2,¶], Amit Pundir[2,‖], Sanjeev Singh[1,**]**

[1]Institute of Informatics and Communication, University of Delhi
[2]Department of Electronics, Maharaja Agrasen College, University of Delhi

[*]anikasrivastava151@gmail.com, [†]researchwithvishal@gmail.com, [‡]harshitasharma@south.du.ac.in, [§]nitisha@south.du.ac.in, [¶]gsaxena@mac.du.ac.in, [‖]amitpundir@mac.du.ac.in, [**]sanjeev@south.du.ac.in

## Abstract

Political memes communicate opinions using the interplay between visual and textual information, posing a challenging multimodal classification task. We present a hierarchical multimodal framework for political meme classification, which decomposes the task into stance and target entity prediction. Our framework was evaluated on the data from the Multi-Level Political Meme Classification Shared Task at MMLow @ AACL-IJCNLP 2026. The task comprised two levels of classification: (i) predicting the political stance of a meme as either Support/Praise or Troll/Oppose, and (ii) identifying the target of the meme combined with the stance. We approached the task by first predicting the Level-1 labels followed by predicting the target entity (Person/Party) and then combining them in a deterministic manner to predict the final Level-2 labels. Our method used a SigLIP vision transformer for stance classification and a hybrid SigLIP-XLM-RoBERTa architecture for target classification with late fusion. We trained our models using 5-fold stratified cross-validation with checkpoint ensembling. We also performed ablation experiments to analyze the contribution of individual modalities, observing that visual features are more informative for stance prediction while textual features contain more information for target identification, which supports the multimodal task at hand. Our system achieved a macro F1 score of 0.9027 and 0.6694 for Level-1 and Level-2 classification, respectively, achieving a final score of 0.7860, ranking first in the shared task.

## 1 Introduction

Political memes have proved to be one of the most prominent online phenomena that shape political opinions through their multimodality, including visual, textual, and symbolic content, as well as their ability to convey complex semantic messages through humor and satire (Anderau and Barbarrusa, 2024). These memes pose unique challenges to computational methods due to their multimodal nature, which requires joint analysis of multiple signals. Political memes, especially those found in low-resource languages such as Tamil, Malayalam, Nepali (Thapa et al., 2025) etc. also pose additional challenges due to code-switching between languages, as well as the need to infer specific political context.

The MMLow Shared Task on Multi-Level Political Meme Classification approached the problem of classifying political memes through a multi-level classification paradigm. In particular, the task is split into two hierarchical levels, with Level 1 asking to classify the input meme into either Support/Praise or Troll/Oppose classes. Level 2 then uses the output of the first classifier to determine the finer classification of the meme, which is one of Support for Person, Support for Party, Troll/Oppose against Person, or Troll/Oppose against Party (Rajiakodi et al., 2026).

Rather than predicting each of the four Level-2 classes directly, we decomposed the problem into two binary classification subtasks: stance prediction (Support/Praise vs. Troll/Oppose) and target entity prediction (Party vs. Person). The two predicted labels were then combined using a deterministic function to obtain the final Level-2 label.

The first stage of our pipeline is based on the Vision Transformer model SigLIP that classifies the image part of the meme into either Support or Troll category. The second pipeline classified the meme into Party or Person by using a combination of multimodal representations created by SigLIP and textual representations obtained with the help of the XLM-RoBERTa language model. Furthermore, we performed an ablation study to measure the contribution of visual and textual modalities to the stance, target, and Level-2 classification tasks. Our proposed method achieved the best overall score (0.7860) for the shared task, ranking first.



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 91–96
November 9, 2026 ©2026 Association for Computational Linguistics

## 2 Related Work

Advances in vision-language pretraining have enabled better understanding of the associations between images and text through the joint learning of representations in a large-scale dataset. Specifically, CLIP (Contrastive Language–Image Pretraining) learns image and text encoders in a joint manner using natural language supervision. This allowed for achieving strong zero-shot performance across a variety of vision tasks (Radford et al., 2021). SigLIP, an updated version of CLIP, uses a sigmoid-based pairwise loss, allowing for optimizing individual image-text pairs, as opposed to large batches of pairs in the original CLIP. This modification allowed a significant reduction in the amount of calculation needed to obtain results comparable to CLIP in terms of image-text matching accuracy and achieving higher performance on downstream vision tasks (Zhai et al., 2023). XLM-RoBERTa (XLM-R) is a Transformer-based model which was pretrained using masked language modeling on one hundred different languages (Conneau et al., 2020). It has shown strong performance on various downstream tasks in different languages, especially for tasks that may contain code-switching between two or more languages.

Low-resource languages pose additional challenges in achieving multimodal understanding due to a lack of annotated data, code-mixing, transliteration, and culturally specific references that are not present in high-resource languages. For Dravidian languages, recent works have aimed to construct benchmark data sets and initiate shared tasks for automatic social media content understanding. Such works include the tasks of detecting offensive language, identifying hate speech, and classifying memes (Chakravarthi et al., 2021); (B et al., 2022). These works showed that pretrained multilingual language models can benefit from the joint modeling of vision and language with limited parallel data. However, understanding political memes requires common sense, political, and regional cultural knowledge, necessitating a tailored multimodal approach.

Recent works have addressed the task of multimodal political meme analysis in the low-resource Dravidian languages context in the framework of the DravidianLangTech–ACL 2026 Shared Task on Multi-Level Political Meme Classification (Rajiakodi et al., 2026). The shared task, similar to the task at hand, proposed a hierarchical approach to classifying political memes in Tamil and Malayalam languages requiring first to detect the stance of a meme and then to identify the political target. The participants attempted to address the challenge by employing various multimodal approaches including transformer-based text encoders, CLIP-based vision encoders, OCR-assisted pipelines, cross-modal fusion techniques, and Vision Language Models. The results revealed that fusion of modalities and hierarchical modeling contribute significantly to improving the performance of stance detection while the identification of targets remains a challenging task due to sarcasm, cultural contexts, OCR errors, and label imbalance. The shared task provided a benchmark for exploring effective multimodal approaches to classifying political memes in the low-resource language setup.

## 3 Methodology

Figure 1 shows the overall architecture of our proposed framework. Our approach broke the problem into two hierarchical subtasks. First, a visual classifier was used to predict the coarse-grained stance (Support/Praise or Troll/Oppose). Second, a multimodal classifier was utilized to classify if the target of the meme is a political person or a political party by combining visual and textual information. Finally, a deterministic mapping was applied to the outputs of the first two classifiers to achieve the four fine-grained Level-2 categories (Support for Person, Support for Party, Troll/Oppose Against Person, and Troll/Oppose Against Party).

### 3.1 Data Preprocessing

The meme dataset consisted of images and their corresponding text i.e. the text present in the meme. The images were processed using the SigLIP AutoImageProcessor. Then the images were resized to $512 \times 512$ pixels. Normalization of the images was done according to the pretrained SigLIP Vision Transformer requirements (Zhai et al., 2023). To ensure robustness and reduce overfitting risks, each image underwent data augmentation, including RandomResizedCrop, 5-degree rotation, and ColorJitter. The meme text was tokenized using the XLM-RoBERTa tokenizer with a maximum sequence length of 128. Notably, any caption exceeding the limit was truncated, while short texts were padded. Moreover, to avoid blank captions, any missing text was replaced with “no caption”.

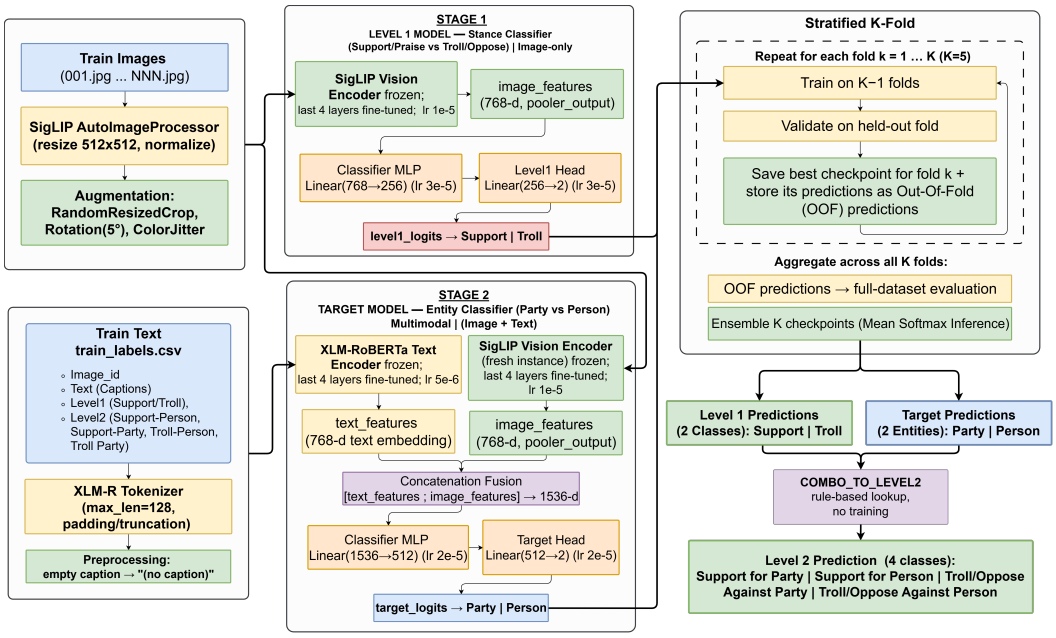



Figure 1: Overview of the training pipeline for the proposed hierarchical multimodal framework.

### 3.2 Stage 1: Visual Stance Classification for Level 1

The first stage was an image-only classifier to differentiate between the Support/Praise and Troll/Oppose stances. A pretrained SigLIP Vision Transformer was utilized as the image encoder. Given an input image (I), the encoder (E) produced a pooled visual representation

$$v = E_v(I), \quad \text{where } v \in R^{768} \tag{1}$$

To retain the pretrained visual knowledge while adapting to the downstream task, we partially fine-tuned the encoder by freezing all transformer layers except the last four. The resulting visual embedding was then fed into a lightweight multilayer perceptron, described in Table A.4. The model was trained with a class-weighted cross-entropy loss to address the imbalance between the two classes.

### 3.3 Stage 2: Multimodal Target Classification

The second level used both visual and textual information to predict whether the target of the meme was a party or a person. The input image was encoded using the pretrained SigLIP Vision Transformer from Section 3.2 (Eq. 1) resulting in a 768-dimensional visual embedding. The text from the accompanying meme was encoded using XLM-RoBERTa-base to obtain a 768-dimensional text embedding using attention-mask-aware mean pooling,

$$t = E_t(T), \quad \text{where } t \in R^{768} \tag{2}$$

where $E_t$ denotes the text encoder and T represents the input meme text. For both models, only the last four transformer layers were fine-tuned, while the rest were frozen. Finally, visual and textual embeddings were concatenated: h=[v;t] to get a 1536-dimensional fused representation. This feature-level concatenation is termed as "late fusion" which allows the classifier to exploit complementary information from the two modalities for target prediction. The fused embedding, h, was then used as input to the classification network, which is described in Table A.4. The classifier then predicted whether the meme targets a political party or a political person.

### 3.4 Five-Fold Stratified Cross-Validation

Both the Stage-1 and Stage-2 models were trained using 5-fold stratified cross-validation (Kohavi, 1995) to ensure maximum utilization of the data, due to the low resource nature of the dataset, while retaining the distribution of the different classes in each fold. For each fold, one of them was taken as a hold-out validation set, and the other four were used for training. For each fold, a new model was trained from the pretrained encoder weights and early-stopped based on the performance of the validation set. The best model checkpoint and OOF predictions on the hold-out set were saved for ensemble and performance estimation purposes.

Once the five folds had been processed, the OOF predictions of all folds were pooled together, giv-

ing an unbiased estimate of performance on the training set. This allowed the models to generalize properly on unseen data later. When making predictions on the test set, all five of the best performing checkpoints were used to make the final prediction using mean softmax ensembling.

### 3.5 Level-2 Prediction

For a meme, the Stage 1 classifier predicts the stance label, i.e., the Level 1 for the task, while the target classifier predicts the target entity. The final Level-2 label is then obtained through a deterministic Cartesian-product mapping of the predicted stance and target labels, as shown in Table A.3, which produced the output of one of the four official shared-task labels for Level 2 (Support for Person, Support for Party, Troll Against Person, and Troll Against Party).

## 4 Results

Table 1 presents the official evaluation results of our submitted system on the test set which achieved Rank 1. The system is evaluated using the macro F1 Score, as described in Appendix A.2.

| Category | Level 1 | Level 2 | Final |
|---|---|---|---|
| Score | 0.9027 | 0.6694 | 0.7860 |

Table 1: Evaluation results as reported by the organizers.

## 5 Discussion

### 5.1 Stratified K-Fold Cross Validation

To obtain an unbiased assessment of model performance while using the training data most effectively, we performed a 5-fold stratified cross-validation. The validation was stratified on the Level-2 labels as it preserved the labels of Level 1. We performed stratified sampling to ensure that each class is represented with the same distribution in all folds. Training and validating on different splits reduces the model's dependence on a particular train-validation split and gives a better estimation of its generalization ability.

### 5.2 Ablation Study

The results of the ablation study comparing modalities, as shown in Table A.1, demonstrated that the majority modality depended on the task. The image-only model achieved a comparatively high Macro F1 of $0.8472 \pm 0.0520$ on the Level-1 stance classification, suggesting the effectiveness of visual cues such as political symbols, facial expressions, and meme templates, in differentiating the Support/Praise from Troll/Oppose stances. Seeing these results, we independently fine-tuned a Level-1 image-only model which attained an even better result of $0.8952 \pm 0.0179$, as compared to the Text+Image model at Level 1 ($0.8838 \pm 0.0263$). Thus, an image-only classifier was ultimately used for the Level-1 prediction. We concluded that the addition of text modality did not consistently improve the performance and may have introduced additional noise, possibly due to the non-uniform and highly variable nature of meme accompanying texts.

Target prediction benefited more from textual information, as the text-only model ($0.6618 \pm 0.0562$) significantly outperformed its image-only counterpart ($0.2732 \pm 0.0872$). The multimodal model achieved a relatively higher performance than the unimodal models of $0.7288 \pm 0.0433$, implying the complementarity of vision and language cues for target identification. These insights directly informed the design of our hierarchical framework.

## 6 Conclusion

Our paper proposed a hierarchical classification framework for the Shared Task on Multi-Level Political Meme Classification. It decomposed the Level-2 classification task into two separate tasks: stance prediction and target entity discovery. The method predicted the stance label using a SigLIP-based image classifier while detecting target entities relies on a multimodal SigLIP–XLM-RoBERTa classifier with late fusion. The results of the two classifiers are merged into the final solution using a lookup table. Our method demonstrated its effectiveness as the validation score achieved using the 5-fold stratified cross-validation procedure almost equals the official test result, which suggests that the model is not overfitting. Our final shared-task score was 0.7860, which placed us in the first position. Some limitations faced were the dataset used for training is small and imbalanced, which impacts the quality of representation. Secondly, the target entity classifier used only learned representations for classification; incorporating additional knowledge, like a list of political parties and people, would allow improving the classification, especially for the Party/Person targets distinction.

## References


Glenn Anderau and Daniel Barbarrusa. 2024. The function of memes in political discourse. *Topoi*, 43(5):1529–1546.

Premjith B, Bharathi Raja Chakravarthi, Malliga Subramanian, Bharathi B, Soman KP, Dhanalakshmi Vadivel, Sreelakshmi K, Arunaggiri Pandian, and Prasanna Kumar Kumaresan. 2022. Findings of the shared task on multimodal sentiment analysis and troll meme classification in Dravidian languages. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 254–260. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan R L, John Philip McCrae, and Elizabeth Sherly. 2021. Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 133–145. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. *Preprint*, arXiv:1911.02116.

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. *Preprint*, arXiv:2103.00020.

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.

Marina Sokolova and Guy Lapalme. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing Management*, 45(4):427–437.

Surendrabikram Thapa, Hariram Veeramani, Liang Hu, Qi Zhang, Wei Wang, and Usman Naseem. 2025. A multimodal prompt-based framework for analyzing Code-Mixed and Low-Resource memes. *Proceedings of the International AAAI Conference on Web and Social Media*, 19:1913–1923.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.


# A Appendix

## A.1 Ablation Study Results

| Model | Level 1 | Target | Level 2 |
|---|---|---|---|
| Text-only | $0.2618 \pm 0.0726$ | $0.6618 \pm 0.0562$ | $0.2154 \pm 0.0147$ |
| Image-only | $0.8472 \pm 0.0520$ | $0.2732 \pm 0.0872$ | $0.3510 \pm 0.0350$ |
| Text + Image | $\mathbf{0.8838 \pm 0.0263}$ | $\mathbf{0.7288 \pm 0.0433}$ | $\mathbf{0.6668 \pm 0.0357}$ |

Table A.1: Macro F1 score results from the ablation study.

## A.2 Evaluation Metrics

Following the shared task evaluation protocol, the model's performance was measured in terms of Precision, Recall, and F1-score. Precision represents the ratio of instances of a given class predicted by the model as positive that are actually positive. In turn, the Recall is the ratio of instances of a given class predicted as positive by the model to the total number of instances of this class. Finally, with F1-score being the harmonic mean of Recall and Precision, it summarizes the model's performance in predicting positive instances.

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP}, \\ \text{Recall} &= \frac{TP}{TP + FN}, \\ \text{F1-score} &= 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \end{aligned} \tag{3}$$

where TP, FP, and FN denote the number of true positives, false positives, and false negatives, respectively.

The overall performance can be measured in two ways – macro and weighted averages. The macro-average is the simple average of the different classes, calculated by the average between the classes regardless of their frequency. The weighted-average applies the number of instances in each class as weights, making the results affected by the class distribution and not an average value anymore. The macro F1 is useful in cases when we have a high imbalance between the classes and want to consider all the classes equally. On the other hand, the weighted F1 is a better estimate of the overall performance of the model than the macro F1 when the classes are imbalanced (Sokolova and Lapalme, 2009).

## A.3 Dataset Distribution

| Class | Before | After |
|---|---|---|
| Support/Praise | 277 | 276 |
| Troll/Oppose | 692 | 693 |
| Support for person | 177 | 177 |
| Support for party | 99 | 99 |
| Troll/Oppose Against Person | 547 | 547 |
| Troll/Oppose Against Party | 146 | 146 |
| **Total** | **969** | **969** |

Table A.2: Class distribution before and after Level 1 label correction.

Table A.2 presents the distribution of samples across the Support/Praise and Troll/Oppose categories at Level 1, along with their corresponding Level 2 classes. During data preprocessing, few inconsistencies between the Level 1 and Level 2 annotations were discovered, where the Level 1 label did not match the category implied by the Level 2 label. These inconsistent labels were overridden using the corresponding Level 2-derived Level 1 label. The modified Level 1 labels were used in the experiments.

## A.4 Level-2 Classification Mapping

| Level-1 Class | Target | Level-2 Class |
|---|---|---|
| Support/Praise | Person | Support for Person |
| Support/Praise | Party | Support for Party |
| Troll/Oppose | Person | Troll/Oppose Against Person |
| Troll/Oppose | Party | Troll/Oppose Against Party |

Table A.3: Stance and target combinations for the Level-2 classification.

## A.5 Confusion Matrix and Error Analysis

Out of 969 instances, 710 (73.27%) had both stance and target correctly predicted. 177 (18.27%) instances had the correct stance but an incorrect target, while 56 (5.78%) had the correct target but an incorrect stance. The remaining 26 (2.68%) instances contained errors in both predictions. Therefore, it can be concluded that the majority of misclassified instances had issues with target identification, which suggests that the prediction of targets was more challenging than stance detection in this hierarchical pipeline.

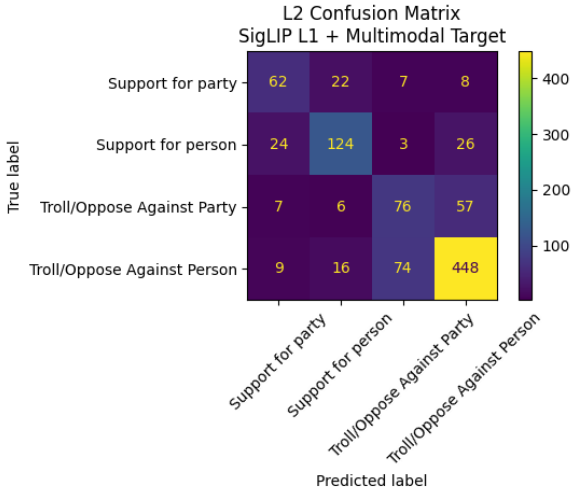



Figure A.1: Confusion matrix for the Level-2 classification, obtained by combining the SigLIP-based Level-1 stance predictions with the multimodal target predictions.

## A.6 Classifier Architecture

| Component | Level-1 Stance Classifier Configuration | Target Entity Classifier Configuration |
|---|---|---|
| Input | 768-dimensional SigLIP image embedding | 1536 (768+768)-dimensional fused image-text embedding |
| Dropout | 0.3 | 0.3 |
| Linear Layer | $768 \rightarrow 256$ | $1536 \rightarrow 512$ |
| Normalization | Layer Normalization | Layer Normalization |
| Activation | GELU | GELU |
| Dropout | 0.2 | 0.2 |
| Output Layer | Linear ($256 \rightarrow 2$) | Linear ($512 \rightarrow 2$) |
| Prediction | Support/Praise or Troll/Oppose | Party or Person |

Table A.4: Architecture of the proposed Level-1 stance classifier and the multimodal target entity classifier.

## A.7 Experimental Setup

The proposed approach was implemented in Python 3.12.13 using PyTorch library (version 2.11.0+cu128) and Hugging Face Transformers (version 5.13.1). In particular, the vision encoder was implemented using the pretrained SigLIP Vision Transformer (google/siglip-base-patch16-512), while the text encoder used was the pretrained XLM-RoBERTa-base (xlm-roberta-base). Finally, the evaluation scores including Precision, Recall, and F1 were calculated using scikit-learn library, while all computations were performed on the NVIDIA GPU with CUDA acceleration.

## A.8 Code Availability

The implementation of our proposed framework is available at: https://github.com/anika-py/CPSLab2-MMLow-Political-Meme-Classification/tree/main

# CPS Lab 1@MMLow 2026: Hierarchical Multimodal Classification of Tamil Political Memes via Logit-Conditioned Stance-Target Prediction

**Vishal Kumar[1,*], Anika Srivastava[1,†], Harshita Sharma[1,‡], Nitisha Aggarwal[1,§], Geetika Jain Saxena[2,¶], Amit Pundir[2,‖], Sanjeev Singh[1,**]**

[1]Institute of Informatics and Communication, University of Delhi
[2]Department of Electronics, Maharaja Agrasen College, University of Delhi

[*]researchwithvishal@gmail.com, [†]anikasrivastava151@gmail.com, [‡]harshitasharma@south.du.ac.in, [§]nitisha@south.du.ac.in, [¶]gsaxena@mac.du.ac.in, [‖]amitpundir@mac.du.ac.in, [**]sanjeev@south.du.ac.in

## Abstract

Political memes have become an effective tool for online political communication and engagement. Their interpretation is particularly challenging in low-resource and code-mixed languages such as Tamil, where memes often include Tamil-English, use culturally relevant lexicon, incorporate political entities, and utilize references to prominent personalities and parties. This paper presents our submission to the MMLow shared task at AACL-IJCNLP 2026 on multi-level classification of Tamil political memes, which requires jointly predicting (i) a coarse-grained stance label (Support vs. Troll) and (ii) a fine-grained target label identifying whether the meme concerns a political party or a person. We proposed a lightweight hierarchical classifier built on frozen, pretrained multimodal encoders: images are represented using CLIP (ViT-B/32) and text using the multilingual BGE-M3 embedding model, with the two modalities combined via early, non-parametric concatenation fusion. A shared encoder feeds two prediction heads, where the target-identification head is explicitly conditioned on the stance head's output logits via a gradient-detachment mechanism, allowing the model to respect the task's label hierarchy without corrupting the coarse-level classifier's gradients. To address pronounced class imbalance in both label levels, we applied balanced loss weighting at each level. On the test set of unlabeled data, our system achieved 3rd place, with macro-F1 scores of 0.856 for stance classification and 0.532 for target identification, yielding a final score of 0.694, indicating promising performance on the Tamil political meme classification problem.

## 1 Introduction

Social media has become a battleground for opinions, and memes have emerged as one of the most powerful tools for expressing political views on them. By combining images, text, symbols, and contextual information, memes allow for communicating ideas effectively to a broad audience that may not fully understand the underlying message's linguistic meaning. The control on these memes becomes increasingly challenging in low-resource and code-mixed language scenarios. In social media meme posts where both Tamil and English are used together. This creates a barrier for monolingual or English-only NLP pipelines to perform well.

The Shared Task on Multi-Level Political Meme Classification, organized as part of MMLow@AACL-IJCNLP 2026 (Rajiakodi et al., 2026), addresses this challenge by proposing a benchmark for automated culturally aware analysis of code-mixed Tamil-English political memes. A sample of the dataset is shown in Figure 1. This shared task is structured as a multi-level classification problem:

- **Level 1 - Stance Identification:** Identify whether a meme expresses support or trolling.
- **Level 2 - Target Identification:** Determine whether the meme concerns a political party or a person, conditioned on Level 1 (e.g., support for a party under a "support" label or troll against a person under a "troll" label).




Figure 1: Sample of the original dataset representing different labels at each stance and target level



*Proceedings of the Fifth Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 97–102
November 9, 2026 ©2026 Association for Computational Linguistics

This hierarchical structure reflects the natural organization of political meme discourse, in which the target of a meme is dependent on its stance. Accurate predictions require this dependency to be modeled, rather than approached as two independent classification tasks, in order to obtain labels that are internally consistent with the hierarchical structure defined by the shared task.

Our system combines frozen pretrained multimodal encoders: images are represented using CLIP (ViT-B/32) (Radford et al., 2021), and text using the multilingual BAAI/bge-m3 embedding model (Chen et al., 2024), with a jointly trained hierarchical classifier. In this classifier, predictions at Level 2 (target) depend directly on those at Level 1 (stance) through a gradient-detached logit-conditioning mechanism. This mechanism captures the label hierarchy structurally while maintaining separate optimization paths for the two output layers.

## 2 Related Work

Multimodal meme analysis has attracted increasing interest due to memes' ability to encode information through the combination of text, images, symbols, and context. (Suryawanshi et al., 2020) proposed the Tamil Memes dataset for troll classification, which emphasizes the importance of addressing the challenges of multimodal Tamil meme classification.

Prior work by (Das et al., 2022) investigated multimodal and ensemble methods for classifying Tamil troll memes, finding that representation learning benefits significantly from combining text and vision. More recently, the task of multimodal meme analysis has been extended to the detection of misogynistic Tamil and Malayalam memes, with promising results obtained with pretrained language and vision models (Ratul et al., 2025).

The MMLow 2026 shared task (Rajiakodi et al., 2026) extends previous Tamil meme research by formulating political meme understanding as a multi-level classification problem. The first level identifies whether a meme expresses support or trolling, while the second level identifies the corresponding political person or party. Thus, MMLow 2026 moves beyond binary troll classification toward fine-grained political stance and target identification, providing an opportunity to develop culturally aware multimodal models for code-mixed Tamil political discourse.

## 3 Methodology

We formulate the MMLow 2026 political meme classification shared task as a hierarchical multi-level classification problem, as shown in Figure 2.

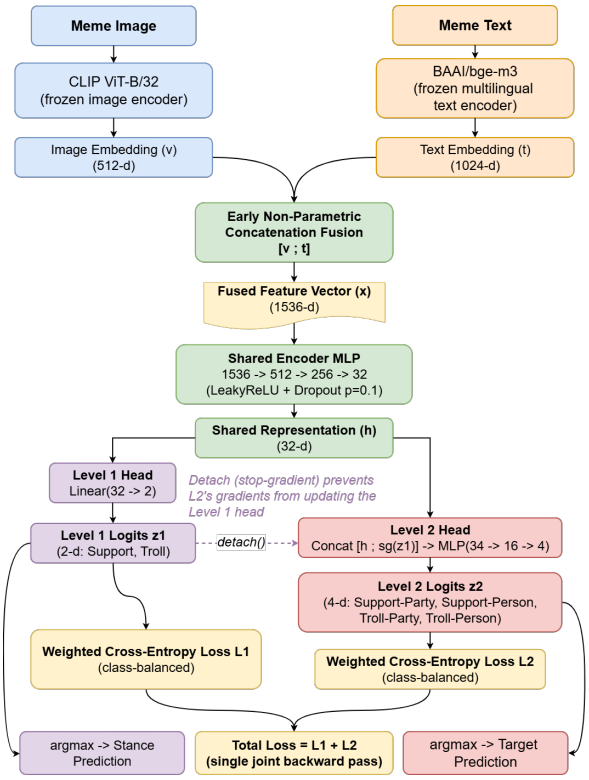



Figure 2: Flow diagram explaining the methodology of our hierarchical approach

Given a meme instance consists of an image $I$ and its associated textual content in Tamil, $T$, our objective is to jointly predict:

- **Coarse-grained Level 1 label:** $y_1 \in \mathcal{Y}_1$, where $\mathcal{Y}_1 = \{\text{Support}, \text{Troll}\}$.
- **Fine-grained Level 2 label:** $y_2 \in \mathcal{Y}_2$, where $\mathcal{Y}_2 = \{\text{Support for Person}, \text{Support for Party}, \text{Troll against Person}, \text{Troll against Party}\}$.

where $y_2$ is semantically dependent on $y_1$. Level 2 prediction is explicitly conditioned on the Level 1 prediction, reflecting this label hierarchy structurally.

### 3.1 Dataset Description and Preprocessing

The dataset contained 969 labeled memes, where each data instance included a meme image along with its corresponding text present in the image. The labelled original dataset D (Rajiakodi et al., 2026) is partitioned into training and validation subsets, using a stratified 90/10 split on the Level

1 label. This strategy preserves class proportions across both subsets and mitigates the effect of the class imbalance (see Figure 4). Categorical labels at both levels are transformed to integer class indices using label encoding.

### 3.2 Multimodal feature extraction

To get semantically rich, modality-specific representations, we employ two frozen pretrained encoders:

- **Textual Encoder:** We experimented with two multilingual text encoders to represent textual information: BGE-M3 (Chen et al., 2024) and MuRIL (Khanuja et al., 2021). We found that BGE-M3 consistently outperformed the other model thus became our preferred text encoder, while MuRIL was used as a baseline for comparison. A detailed comparison of the two encoders is provided in Appendix Table 3. Specifically, the textual content $t_i$ is encoded using the BGE-M3 (BAAI/bge-m3) multilingual encoder.
- **Visual Encoder:** Meme image $v_i$ is encoded using a CLIP ViT-B/32 image encoder (Radford et al., 2021).

### 3.3 Multimodal feature fusion

Visual and textual embeddings are combined via early (Barnum et al., 2020), non-parametric concatenation fusion:

$$x_i = v_i \parallel t_i \tag{1}$$

where $v_i$ is the input meme instance, $t_i$ is the textual content present in the meme, and $\parallel$ is the concatenation operation producing a single fused feature vector $x_i$ per instance.

### 3.4 Hierarchical Classification Architecture

We introduce a hierarchical classifier, composed of a shared trunk and two prediction heads with explicit inter-level conditioning. The complete model description can be seen in the Appendix Table 4

- **Shared encoder:** $x_i$ was passed through a shared multilayer perceptron:

$$h_i = \mathrm{MLP}_{\mathrm{shared}}(x_i) \tag{2}$$

- **Level 1 head:** A linear classification layer maps the shared latent representations to Level 1 logits:

$$z_i^{(1)} = W_1 h_i + b_1 \tag{3}$$

- **Level 2 head (hierarchically conditioned):** It receives the concatenation of the shared latent representation and Level 1 logits, enabling Level 2 prediction to be conditioned on the model's Level 1 prediction:

$$z_i^{(2)} = \mathrm{MLP}_{L2}\left(h_i \parallel \mathrm{sg}\left(z_i^{(1)}\right)\right) \tag{4}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient (detach) operator. By detaching $z_i^{(1)}$ before it enters the Level 2 head, we permit the Level 2 head to exploit the coarse-grained prediction as an informative conditioning signal without allowing Level 2's gradients to backpropagate into, and potentially corrupt, the Level 1 head's optimization. This technique ensures decoupling Level 1 head learns only from its own loss, while the Level 2 head learns from its own loss.

### 3.5 Class Imbalance Mitigation

To confirm the effectiveness of class weighting, we compared the results of using a weighted and an unweighted cross-entropy loss function. As shown in Table 1 the application of a weighted loss improved the average of the macro-F1 scores for both classification levels: from 0.7652 to 0.7738. Specifically, for Level-1, the macro-F1 score increased from 0.7710 to 0.7869. As for Level-2, the performance was comparable, with the weighted loss achieving a slightly higher macro-F1 score of 0.7608 compared to 0.7593 for the unweighted loss. Overall, the use of class weighting led to a better macro-F1 performance for both levels, which suggests that it can be beneficial for addressing the class imbalance problem and improving the minority class recognition.

Table 1: Comparison of weighted and unweighted loss functions for multimodal training using the BGE-M3 text encoder.

| Split | Loss | Avg. Macro-F1 | Std. Dev. |
|---|---|---|---|
| Level-1 | Unweighted | 0.7710 | 0.0198 |
| Level-1 | Weighted | **0.7869** | 0.0508 |
| Level-2 | Unweighted | 0.7593 | 0.0329 |
| Level-2 | Weighted | **0.7608** | 0.0382 |
| Average | Unweighted | 0.7652 | 0.0264 |
| Average | Weighted | **0.7738** | 0.0445 |

### 3.6 Training Objective and Optimization

Our model is trained end-to-end via a joint multitask objective:

$$\mathcal{L} = \mathcal{L}_1\left(z_i^{(1)}, y_i^{(1)}\right) + \mathcal{L}_2\left(z_i^{(2)}, y_i^{(2)}\right) \tag{5}$$

computed and backpropagated in a single optimization step per batch, such that the shared encoder receives gradient signal from both levels simultaneously.

The model's learnable parameters are updated using AdamW optimizer with an initial learning rate of $1 \times 10^{-3}$. A ReduceLROnPlateau scheduler is employed, which monitors the average of Level 1 and Level 2 validation macro-F1 scores and slows down learning by reducing the learning rate by a factor of $0.5$ after 2 epochs without improvement. The model is trained for up to 60 epochs with a batch size of 32. The checkpoint that achieves the highest average validation macro-F1 across both levels is retained for final evaluation and inference.

### 3.7 Evaluation

Model performance is assessed on a held-out validation split using:

- **Macro-averaged F1-score:** Computed independently for Level 1 and Level 2, serving as the primary, imbalance-robust evaluation metric and model selection criterion (see Figure 3).
- **Accuracy:** Reported as a secondary metric for both levels.

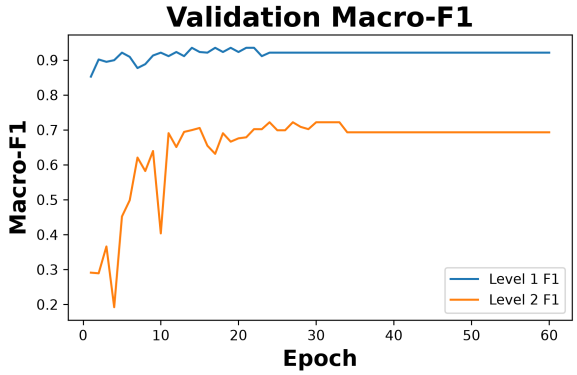



Figure 3: Macro-F1 score for both coarse-grained level 1 and fine-grained level 2 on each epoch

### 3.8 Inference

At test time, the identical feature-extraction and fusion pipeline is applied to unlabeled test instances. The best checkpointed model was used to produce Level 1 and Level 2 logits, from which the argmax class is taken at each level:

$$\hat{y}_1^{(i)} = \arg\max z_i^{(1)}, \qquad \hat{y}_2^{(i)} = \arg\max z_i^{(2)} \tag{6}$$

Predicted class indices are decoded back to their original categorical labels by using label encoders fitted on the training data, and final predictions are exported for submission.

## 4 Results

Table 2 presents the official evaluation results of our submitted system on the unlabeled test set. The system is evaluated using macro F1-score metrics described further in Appendix A.1.

Table 2: Evaluation results as per the organizers.

| | Level 1 | Level 2 | Final (average) |
|---|---|---|---|
| **Score** | 0.856 | 0.5321 | 0.694 |

## 5 Conclusion

Our results demonstrate that a hierarchical classifier built on frozen multimodal embeddings can effectively capture the coarse-to-fine label structure of the MMLow task. All reported metrics are averaged across five random seeds [7, 10, 42, 56, 100], providing a more robust estimate of model performance and reducing the impact of training variance. The strong Level 1 validation macro-F1 (0.898) and accuracy (0.918) confirm the comparative easiness to separate stance (*Support* vs. *Troll*) from fused CLIP + BGE-M3 features. Whereas the lower Level 2 validation macro-F1 (0.676) and accuracy (0.736) reflect increased difficulty of identifying the specific *target* of a meme.

Several limitations should be noted. First, the dataset is small (only 969 labeled memes). Second, by relying on only frozen pretrained encoders with simple concatenation fusion, we essentially miss out on any cross-modal interactions that could be learned by fine-tuning. Third, no named entity recognition is explicitly performed, which arguably could be crucial for target identification; perhaps adding an entity recognition/grounding module would be a useful direction for improving Level 2 performance in the future.

## References

George Barnum, Shirin Talukder, and Yisong Yue. 2020. On the benefits of early fusion in multimodal representation learning. *arXiv preprint arXiv:2011.07191*.

Jia Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multilinguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Mithun Das, Somnath Banerjee, and Animesh Mukherjee. 2022. hate-alert@dravidianlangtech-acl2022: Ensembling multi-modalities for tamil trollmeme classification. *arXiv preprint arXiv:2204.12587*.

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. *Preprint*, arXiv:2103.10730.

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. *arXiv preprint arXiv:2103.00020*.

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.

MD Musa Kalimullah Ratul, Sabik Aftahee, Tofayel Ahmmed Babu, Jawad Hossain, and Mohammed Moshiul Hoque. 2025. Cuet_networksociety@dravidianlangtech 2025: A multimodal framework to detect misogyny memes in dravidian languages. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 529–535, Albuquerque, New Mexico. Association for Computational Linguistics.

Marina Sokolova and Guy Lapalme. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437.

Shivali Suryawanshi, Bharathi Raja Chakravarthi, Pranav Verma, Mihael Arcan, John P. McCrae, and Paul Buitelaar. 2020. A dataset for troll classification of tamilmemes. In *Proceedings of the Second Workshop on Abusive Language Online*.

# A Appendix

## A.1 Evaluation Metric: Macro F1 Score

The macro F1 score (Sokolova and Lapalme, 2009) is a metric used to evaluate the performance of multi-class classification models. It is a harmonic mean of the F1 scores of all classes, where each class is weighted equally. It is suitable for imbalanced data sets and cases that require equal importance to be assigned to each class.

The calculation of this metric consists of two steps:

1. For each class, calculate an F1 score. The F1 score is the harmonic mean of precision and recall:
$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

2. Take the arithmetic average of these per-class F1 scores:
$$\text{Macro F1} = \frac{1}{N} \sum_{i=1}^{N} F1_i \quad (8)$$

where $N$ is the total number of classes, and $F1_i$ is the F1 score for the $i$-th class.

## A.2 Comparison of Text Encoders

Table 3 compares the performance of BGE-M3 and MuRIL in the weighted multimodal scenario. BGE-M3 achieved higher average macro-F1 scores than MuRIL at both classification levels. At Level-1, BGE-M3 achieved a macro-F1 score of 0.7518 compared to 0.7158 for MuRIL, an improvement of 3.6 percentage points. At Level-2, BGE-M3 achieved 0.7346, while MuRIL achieved 0.7218, corresponding to an improvement of 1.3 percentage points. Although the outperformance margin reduces at the finer-grained Level-2 classification task, BGE-M3 remains superior as it facilitates better text representations for multimodal fusion.

Table 3: Comparison of weighted multimodal training using BGE-M3 and MuRIL text encoders

| Split | Text Encoder | Avg. Macro-F1 | Std. Dev. |
|---|---|---|---|
| Level-1 | BGE-M3 | **0.7869** | 0.0508 |
| Level-1 | MuRIL | 0.7272 | 0.0579 |
| Level-2 | BGE-M3 | **0.7608** | 0.0382 |
| Level-2 | MuRIL | 0.7282 | 0.0317 |
| **Average (BGE-M3)** | | **0.7738** | 0.0445 |
| **Average (MuRIL)** | | 0.7277 | 0.0448 |

## A.3 Complete model architecture details

Table 4 presents the complete architecture of the proposed hierarchical model.

## A.4 Code Availability

The implementation of our proposed framework is publicly available at: https://github.com/vishalprasad1912/CPS_Lab_1_MMLow_2026

Table 4: Complete description of the model architecture

| Component | Type | Configuration | Parameters |
|---|---|---|---|
| shared | Sequential | Sequential(<br>(0): Linear(in=1536, out=512, bias=True)<br>(1): LeakyReLU(negative_slope=0.01)<br>(2): Dropout(p=0.1, inplace=False)<br>(3): Linear(in=512, out=256, bias=True)<br>(4): LeakyReLU(negative_slope=0.01)<br>(5): Dropout(p=0.1, inplace=False)<br>(6): Linear(in=256, out=32, bias=True)<br>(7): LeakyReLU(negative_slope=0.01) | 926,496 |
| l1_head | Sequential | Sequential(<br>(0): Linear(in=32, out=2, bias=True) | 66 |
| l2_head | Sequential | Sequential(<br>(0): Linear(in=34, out=16, bias=True)<br>(1): ReLU()<br>(2): Linear(in=16, out=4, bias=True) | 628 |
| **Total trainable** | – | – | **927,190** |

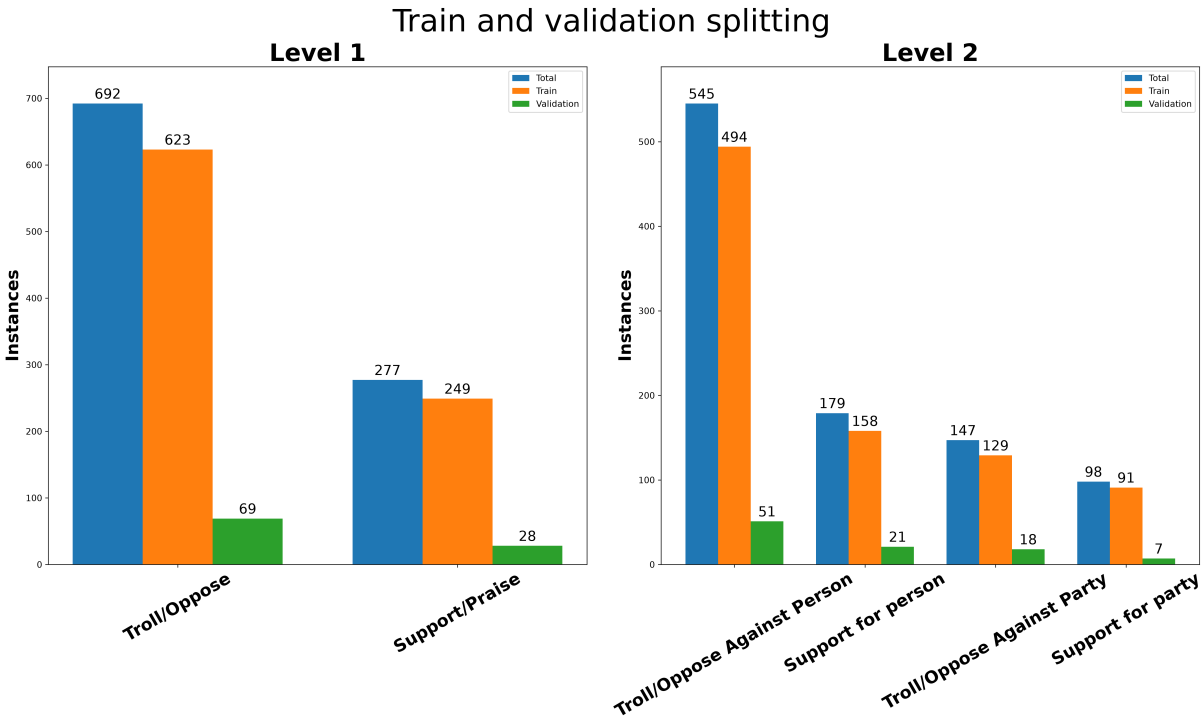



Figure 4: Distribution of the dataset for training and validation using the stratified sampling technique.

# Author Index


A S, Adithyakiran, 18
Aggarwal, Nitisha, 91, 97

Bhowmik, Shimanto, 1
Biju, Ashbi, 18

Dhorali, Sathyanarayanan, 40
Dipto, Tawsif Tashwar, 1
Dutta, Dwaipayan, 76

Ghosh, Kuntal, 40

Hossain, Md. Ajwad, 64
Hsu, Sheryl, 1

I, Palaniraj, 76
Islam, Md Sazzad, 1

K, Kevin selva, 70
Kumar, Vishal, 91, 97

M, Sekar, 55
Mohan, Pranava, 55

N, Ramkumar, 28
Nazeem, Meharuniza, 18

Pundir, Amit, 91, 97

R R, Rajeev, 18
R, Thilagavathy, 70
Raihan, Md, 40
Rajak, Nilam, 40
Reasat, Tahsin, 1
RM, Asvitha, 86

S, Navaneeth, 18
S, Rajendran, 28
Santhanavijayan, A., 55
Saxena, Geetika Jain, 91, 97
Sharma, Harshita, 91, 97
Singh, Sanjeev, 91, 97
Srivastava, Anika, 91, 97

VS, Amirthamozhi, 86