Multi-LLL 2026

Workshop on Multi-modality and Multi-lingual Language Learning

Proceedings of the Workshop

November 9, 2026

©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 979-8-89176-449-1

Preface

Welcome to the Workshop on Multi-modality and Multi-lingual Language Learning (Multi-LLL 2026), held on November 9, 2026, in Hengqin, Zhuhai, China, in conjunction with AACL-IJCNLP 2026.

Language learning draws on speech, writing, visual information, and interaction. This workshop brings together researchers working across natural language processing, computer vision, speech, artificial intelligence, and education to explore how these resources can support learners with different linguistic backgrounds and needs. Its scope includes learning technologies, assessment, feedback, and applications for English, Chinese, and low-resource languages.

Alongside research presentations and invited talks, the workshop features the VisCGEC-v2.0 shared task on visual Chinese grammatical error correction. The task asks systems to locate and correct errors in images of handwritten Chinese text, with sentence-level and paragraph-level tracks. It connects visual understanding with language analysis in an educational setting.

We thank the authors, reviewers, invited speakers, shared-task participants, and the AACL-IJCNLP 2026 organizing team for their contributions. We hope the workshop encourages continued exchange and collaboration on language learning across modalities and languages.

Xiaoman Wang, Qianyu Wang, Zhengyuan Liu, Nancy F. Chen,
Zheng Yuan, Yunshi Lan, and Jing Jiang
Multi-LLL 2026 Workshop Organizers

Organizing Committee

Workshop Organizers

Xiaoman Wang, East China Normal University
Qianyu Wang, East China Normal University
Zhengyuan Liu, A*STAR, Singapore
Nancy F. Chen, A*STAR, Singapore
Zheng Yuan, University of Sheffield
Yunshi Lan, East China Normal University
Jing Jiang, Australian National University

Program Committee

Program Committee

Ahmed Karim, Hassana Labs
Andrew Caines, University of Cambridge
Bashar Alhafni, MBZUAI
Chris J. Bryant, Writer Inc.
Chunyang Wang, East China Normal University
David Strohmaier, University of Cambridge
Diana Galvan-Sosa, University of Cambridge
Gabrielle Gaudeau, University of Cambridge
Jiazheng Li, King's College London
Jing Xu, Cambridge University Press & Assessment
Judy Wang, Hosei University
Lei Shi, Newcastle University
Luca Benedetto, Institut polytechnique de Paris & Telecom SudParis
Mariano Felice, British Council
Michael Hofmeyr, Tokyo University of Science
Michael Zock, Aix-Marseille University
Michail Korakakis, University of Cambridge
Mingrui Ye, King's College London
Nevena Cekovic, University of Belgrade
Nguyen Li, Nanyang Technological University
Ranka Stanković, University of Belgrade
Robert Swier, Kindai University
Roman Yangarber, University of Helsinki
Siya Qi, King's College London
Stella Xin Yin, Nanyang Technological University
Suchir Salhan, University of Cambridge
Thierry Geoffre, University of Luxembourg
Xuesong Lu, East China Normal University
Yan Wang, East China Normal University
Yuan Gao, University of Cambridge

Invited Speakers

Zitao Liu, Jinan University
Guanliang Chen, Beijing Normal University
Zhongqing Wang, Soochow University

Keynote Talk

Multimodal Language Technologies for Personalized Learning: From Knowledge Tracing to a Daily AI Study Companion

Zitao Liu
Jinan University
Monday, November 9, 2026 – Time: **09:05 – 09:45** – Room: **MR8M**

Abstract: In this talk I will present our work on multimodal language technologies for personalized learning, connecting foundational research with a deployed product. I will show how we model a learner's knowledge state from text, images, handwriting, and speech using interpretable knowledge-tracing models, together with a standardized evaluation platform and a large open dataset XES3G5M. I will then describe our multimodal educational resources and models for OCR, speech, and handwritten-formula recognition, and how we improve mathematical reasoning in small models. Finally, I will introduce Xiao Jing Long, a student-facing AI study companion on learning devices: it reads photos of homework and handwritten answers, identifies each day's most important weak point, and guides a personalized 15–20-minute learning session.

Bio: Zitao Liu is the Dean of Guangdong Institute of Smart Education, Jinan University, Guangzhou, China. His research is in the area of machine learning, including artificial intelligence in education and educational data mining. He has published over 100 papers in highly ranked conference proceedings, such as NeurIPS, CVPR, AAAI, WWW, etc. and his applied research has resulted in more than 50 technology transfer and patents. Zitao serves as the Executive Committee of the International AI in Education Society, CCF Senior member and top tier AI conference/workshop organizers/program committees. He was the champions and runners-up of many competitions at top AI venues, including NeurIPS, EMNLP and Ubicomp.
He is a recipient of ACM and CCF Distinguished Speaker and Beijing Nova Program 2020. Before joining TAL, Zitao was a senior research scientist at Pinterest and received his Ph.D degree in Computer Science from University of Pittsburgh.

Keynote Talk

Beyond Automated Feedback: Advancing Educational Feedback through Human–AI Collaboration

Guanliang Chen
Beijing Normal University
Monday, November 9, 2026 – Time: **09:50 – 10:30** – Room: **MR8M**

Abstract: Generative AI is reshaping educational feedback, offering new opportunities to provide personalised, timely, and scalable support for teaching and learning. However, its potential extends beyond automating feedback generation to enhancing feedback quality and effectiveness through meaningful human–AI collaboration. Drawing on a systematic review and empirical studies, this talk examines how AI can support educational feedback practices, from generating personalised feedback to analysing feedback quality and facilitating human–AI collaborative feedback processes. Particular attention is given to how AI can complement educators' pedagogical expertise, support informed decision-making, and enable the provision of actionable and contextually relevant feedback. The talk concludes by highlighting key opportunities and challenges in developing human–AI collaborative approaches that enhance feedback practices and improve student learning experiences.

Bio: Guanliang Chen is a Professor at Beijing Normal University. His research lies at the intersection of Artificial Intelligence (AI), Learning Analytics, and Education, with a particular focus on developing responsible AI technologies to support large-scale, personalised learning, assessment, and feedback. He has published over 100 peer-reviewed journal and conference papers in leading international venues across AI, educational technology, and learning sciences, accumulating more than 10,000 citations on Google Scholar with an h-index of 40.

Keynote Talk

Large Language Models for Figurative Language: From Text to Multi-modality and Cross-lingual Inspirations

Zhongqing Wang
Soochow University
Monday, November 9, 2026 – Time: **11:00 – 11:40** – Room: **MR8M**

Abstract: Figurative language such as metaphor and synaesthesia poses great challenges for natural language understanding and second language acquisition, due to its non-literal and cognition-driven nature. This talk presents our research on figurative-language understanding from textual to multimodal and Chinese cross-lingual perspectives. We revisit textual metaphor detection with traditional neural and BERT-based models, and introduce our Chain-of-Thought empowered multimodal metaphor detection framework leveraging both image and text signals. We further explore Chinese synaesthesia: we build a human-annotated dataset, and address synaesthesia detection via joint extraction and tree-structured generation models. Our findings provide insights for cognition-aware NLP and figurative language processing across modalities and languages.

Bio: Zhongqing Wang is an Associate Professor at the School of Computer Science and Technology, Soochow University. His primary research interests span natural language processing, sentiment analysis, and information extraction. He has published over 50 papers in top-tier conferences and journals in AI and NLP. He currently leads and participates in multiple projects funded by the National Natural Science Foundation of China. He serves as a Senior Program Committee Member or Area Chair for prestigious international conferences including ACL, IJCAI, AAAI, EMNLP, and COLING.

Table of Contents

WrAFT: A Modularized Automated Writing Evaluation System for L2 Argumentative Essays
Adnan Labib, Yixuan Huang, Jiahui Wu, John Maurice Gayed, Zheng Yuan and Qiao Wang . . . 1

Beyond Accuracy: What Language-Subspace Interventions Capture, Remove, and Improve
Shaanil Punglia, Dadhichi Telwadkar, Vaibhav Rathore, Siddharth Sureka and Neha Bhargava 19

Ground Before Correcting: Evidence-Guided Visual Chinese Grammatical Error Correction at Multi-LLL 2026
Xiaochuan Xie and Keran Huang 33

Cross-Lingual TTS for Indian Languages: A Benchmark and Inference-Time Duration Estimation Strategies
Adarsh Arigala, Arjun Gangwar and Srinivasan Umesh 47

Rewrites Are Not Lessons: Teachable Edits as an Output Specification for Multimodal Language Learning
Wajdi Zaghouani 56

A Multi-Modal Interactive Narrative Game for French-Speaking Primary School Learners
Qiao Wang, Adnan Labib, Thierry Geoffre, Yixuan Huang and Zheng Yuan 65

A Training-Free Single-Pass Vision–Language Pipeline for Sentence-Level Visual Chinese Grammatical Error Correction
Lexuan Zhang 77

Outlining an Experiment Design of Gamifying Language Learning in Low-Resource Contexts: Kurdish Learning in University, Language Courses, and on Web-Based Applications
Quánzī Mǎ, Christian Schuler, Xueqing Zhang, Meiyi Chen, Mahîr Dogan, Raman Ahmad, Tomohiro Nagashima, Timo Baumann and Josef Van Genabith 84

Clean Evaluations on Contaminated Visual Language Models
Hongyuan Lu, Shujie Miao and Wai Lam 94

Audio Representation Learning and Downstream Task Modeling: A Survey
Kaihong Wang and Liang He 100

Program

Monday, November 9, 2026

09:00 - 09:05 *Opening Remarks*

09:05 - 09:45 *Zitao Liu: Multimodal Language Technologies for Personalized Learning: From Knowledge Tracing to a Daily AI Study Companion*

09:45 - 09:50 *Transition*

09:50 - 10:30 *Guanliang Chen: Beyond Automated Feedback: Advancing Educational Feedback through Human–AI Collaboration*

10:30 - 11:00 *Coffee Break*

11:00 - 11:40 *Zhongqing Wang: Large Language Models for Figurative Language: From Text to Multi-modality and Cross-lingual Inspirations*

11:40 - 11:45 *Transition*

11:45 - 11:50 *Shared Task Overview*

11:50 - 12:05 *Visual Chinese Grammatical Error Correction*

Ground Before Correcting: Evidence-Guided Visual Chinese Grammatical Error Correction at Multi-LLL 2026
Xiaochuan Xie and Keran Huang

12:10 - 12:25 *Visual Chinese Grammatical Error Correction (Recorded Video)*

A Training-Free Single-Pass Vision–Language Pipeline for Sentence-Level Visual Chinese Grammatical Error Correction
Lexuan Zhang

12:25 - 12:30 *Shared Task Discussion*

12:30 - 14:00 *Lunch Break*

14:00 - 14:16 *Writing, Feedback and Interactive Learning*

Monday, November 9, 2026 (continued)

WrAFT: A Modularized Automated Writing Evaluation System for L2 Argumentative Essays
Adnan Labib, Yixuan Huang, Jiahui Wu, John Maurice Gayed, Zheng Yuan and Qiao Wang

14:17 - 14:33 *Development of an Intelligent Essay-Correction System for International Chinese Language Education (DAN YUAN; non-archival)*

14:34 - 14:50 *PEELing Argumentation: Scaffolding Elementary Argumentative Writing with Decomposed LLM Feedback (Stella Xin Yin, Geyu Lin, Zhengyuan Liu, Brian Y. Lim, Nancy F. Chen; non-archival)*

14:51 - 15:24 *Writing, Feedback and Interactive Learning*

Rewrites Are Not Lessons: Teachable Edits as an Output Specification for Multimodal Language Learning
Wajdi Zaghouani

A Multi-Modal Interactive Narrative Game for French-Speaking Primary School Learners
Qiao Wang, Adnan Labib, Thierry Geoffre, Yixuan Huang and Zheng Yuan

15:25 - 15:30 *Session Discussion*

15:30 - 16:00 *Coffee Break*

16:00 - 17:25 *Multilingual, Multimodal and Speech Technologies*

Outlining an Experiment Design of Gamifying Language Learning in Low-Resource Contexts: Kurdish Learning in University, Language Courses, and on Web-Based Applications
Quánzī Mǎ, Christian Schuler, Xueqing Zhang, Meiyi Chen, Mahîr Dogan, Raman Ahmad, Tomohiro Nagashima, Timo Baumann and Josef Van Genabith

Beyond Accuracy: What Language-Subspace Interventions Capture, Remove, and Improve
Shaanil Punglia, Dadhichi Telwadkar, Vaibhav Rathore, Siddharth Sureka and Neha Bhargava

Cross-Lingual TTS for Indian Languages: A Benchmark and Inference-Time Duration Estimation Strategies
Adarsh Arigala, Arjun Gangwar and Srinivasan Umesh

Audio Representation Learning and Downstream Task Modeling: A Survey
Kaihong Wang and Liang He

Clean Evaluations on Contaminated Visual Language Models
Hongyuan Lu, Shujie Miao and Wai Lam

17:25 - 17:30 *Closing Remarks*

WrAFT: A Modularized Automated Writing Evaluation System for L2 Argumentative Essays

Adnan Labib[1], Yixuan Huang[2], Jiahui Wu[3], John Maurice Gayed[2], Zheng Yuan[4], Qiao Wang[5]
[1]Department of Informatics, King's College London, UK
[2]Waseda University, Tokyo, Japan
[3]Beijing University of Posts and Telecommunications, Beijing, China
[4]University of Sheffield, Sheffield, UK
[5]Hosei University, Tokyo, Japan
adnan.1.labib@kcl.ac.uk, yixuan.huang@moegi.waseda.jp, amy_wjh@bupt.edu.cn
gayed@waseda.jp, zheng.yuan1@sheffield.ac.uk, judy.wang@hosei.ac.jp

Abstract

Second language writers rarely receive the detailed feedback they need to improve, as comprehensive feedback takes teachers days per class. To solve this, we present *WrAFT*, a **Wr**iting **A**ssessment and **F**eedback **T**ool, that delivers reliable scores along with effective comprehensive feedback for argumentative essays. *WrAFT* adopts a modular design by dividing the automated writing evaluation (AWE) into essay scoring, surface-level feedback and deep-level feedback. In building the system, various Large Language Models (LLMs) have been evaluated, including LLaMA-3.3-70B-Instruct, GPT-4o, and Claude 3.7, through both supervised fine-tuning and direct prompting. Evaluation shows that *WrAFT* achieves strong agreement with human raters, with a Quadratic Weighted Kappa (QWK) of 0.84 on a proprietary dataset of 480 TOEFL Independent Writing essays. More importantly, human evaluation of system-generated feedback also reveals high approval ratings (96.14% for surface-level, 93.03% for deep-level macro feedback, and 94.69% for deep-level micro feedback). Publicly available interactive user interface has also been developed for the system.

1 Introduction

Second language writers need comprehensive feedback to improve, but rarely receive it as teachers may take days to provide feedback, and what learners get is often generic (Liu and Kunnan, 2015; Ranalli and Yamashita, 2021). Automated Writing Evaluation (AWE) systems have thus emerged as a promising solution to score and provide feedback on student essays at scale. Traditional AWE systems have mostly focused on scoring (Li and Ng, 2024), either through linguistic features, shallow text similarity measures (Li and Wu, 2023) or through neural network approaches, giving grades based on a single vector representation of the essay (Dong et al., 2017; Jin et al., 2018). Though some of the systems demonstrated satisfactory scoring agreements with human raters, such as *e-rater*[1], the scoring engine of *Criterion*, Education Testing Service's (ETS) AWE tool for the GRE and TOEFL writing exams (Attali and Burstein, 2006), they often failed to capture higher-order thinking in writing such as coherence or argumentation.

With the advancement of Large Language Models (LLMs), LLM-based AWE systems can now predict a score along with giving rich explanatory feedback. However, Li and Ng (2024) shows that in recent years, this area of research seems to be narrowly focused on developing a sophisticated model that can beat competing models in scoring a standard evaluation dataset, while there is a lack of feedback generation and validation. Li and Ng (2024) proposes that there should be different layers to AWE systems, including holistic or trait-specific scores, written feedback, and essays revised by experts. This proposal coincides with the need for comprehensive feedback stressed by second language writing research, which typically includes both corrective edits to surface-level errors in grammar, wording and mechanics, and deep-level feedback comments on issues such as organization, coherence and argumentation (Bitchener and Knoch, 2008). While some researchers

[1]https://www.ets.org/erater/about.html

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 1–18
November 9, 2026 ©2026 Association for Computational Linguistics

caution against the potential cognitive overload that comprehensive feedback might impose on learners (Truscott, 1996), it still remains a prevalent pedagogical practice in language education, as learners may selectively attend to aspects they find most relevant. However, though there have been many grammatical error correction (GEC) systems that can be used for surface-level feedback (Yuan and Briscoe, 2016; Imamura et al., 2012; Rozovskaya and Roth, 2019; Davis et al., 2024; Bryant et al., 2023; Yuan et al., 2021), deep-level feedback remains rare and is usually in a generic manner. In addition, feedback comments are frequently consolidated in standalone paragraphs separate from the essay text, as observed in (Stahl et al., 2024), which makes it inconvenient for users to associate the feedback with specific elements of their writing.

One notable AWE system, AiAWE (Gayed, 2025), provides highly reliable and accurate TOEFL-benchmarked scoring and holistic feedback to argumentative essays through LLMs fine-tuned on a proprietary TOEFL independent writing dataset provided by Education Testing Service (ETS). Inspired by the AiAWE system and drawing on the first two layers of feedback proposed by Li and Ng (2024), this research aims to address the current research gap by developing an LLM-based AWE system for argumentative essays that provides accurate scoring and layered, comprehensive feedback. Internally, a modular architecture is adopted with separate modules for scoring, surface-level feedback and deep-level feedback. This system is made available through an interactive user interface which transforms the system from a research model into a practical and pedagogically effective educational tool. The contributions are as follows:

- The scoring module achieves strong agreement with human raters (QWK of 0.84 on 1-5 score scale), allowing learners to understand their essay scores on the TOEFL rubric level.
- Human validation of the comprehensive feedback generated by the AWE system suggests highly satisfactory performance in correcting surface-level grammatical and mechanical errors, as well as in commenting on macro structure and micro aspects including context-dependent grammar, clarity, coherence, argumentation, and formality.
- For feedback generation, which is a task with non-deterministic output, supervised fine-tuning was less effective than directly prompting SOTA LLMs to elicit feedback on specific traits/aspects in writing.
- The research results in a deployable code package[2] and a web-based application to visualize comprehensive feedback through in-line corrective edits and anchored comments[3].

2 Background

2.1 Early automated writing evaluation

AWE is defined as "the process of evaluating and scoring written prose via computer programs" (Shermis and Wilson, 2024), which includes the dual task of scoring and feedback. Page (1967) first proposed computer based essay grading and developed Project Essay Grade (Page, 2003). Early AWE systems adopted machine learning approaches, for instance, *e-rater* (Attali and Burstein, 2006) from ETS utilizes a suite of handcrafted linguistic features, such as grammar errors, vocabulary sophistication, organization indicators, etc. to evaluate writing. Another AWE system, Intelligent Essay Assessor (IEA) (Landauer et al., 2003), employs latent semantic analysis to measure the semantic similarity between an essay and high-scoring responses. More recent AWE systems have adopted deep learning techniques (Li and Ng, 2024), such as the work of Taghipour and Ng (2016), who employed a convolutional neural network (CNN) to extract n-gram-level features for capturing local dependencies, followed by a long short-term memory network (LSTM) to model global, long-distance dependencies for essay scoring. While they emphasized the operational potential of these systems, later critiques (see Perelman, 2014) show that many of these scoring engines disproportionately reward surface-level features, especially essay length (word count), rather than substantive content, raising questions about the depth of construct representation.

Further investigation shows that while some of the above systems have proven effective for scoring, they lack true understanding of content, particularly higher-order thinking such as logic and argumentation. Thus, feedback in early AWE systems, if any, tends to be formulaic (e.g., pointing out grammatical and mechanical errors) rather than giving insight into argument strength or coherence (Li and

[2] https://github.com/judywq/wraft
[3] https://wraft.net/

Ng, 2024). In addition, while *e-rater* has been used extensively in high-stakes environments, Powers et al. (2002) demonstrates that writing experts and computational linguists were able to deliberately create essays that exploit the *e-rater* algorithm. Their findings highlight that automated systems could reward superficial linguistic features rather than the deeper qualities of coherent and original writing. Deane (2013) extends on this by arguing that while AWE systems can capture certain linguistic and structural features, they risk underrepresenting higher-order writing dimensions such as idea development and rhetorical effectiveness.

2.2 LLM-based approaches

Recently, LLMs, with their strong text comprehension and generation capabilities, can read a student essay and produce a detailed critique in natural language, often well beyond the templated feedback of older AWE systems. Mizumoto and Eguchi (2023) empirically evaluated the use of LLMs for automated essay scoring by using GPT-3 to assess essays from ETS's TOEFL11 dataset. The researchers reported the model achieving good levels of agreement and reliability with the human ratings by professional ETS raters. In addition, they noted improvements when combining LLM evaluation with linguistic feature-based approaches. The study demonstrated zero-shot LLM scoring is viable, but still needing methodological refinements to ensure consistency and construct validity.

Efforts to utilize LLMs for AWE tasks include both direct prompting and fine-tuning approaches. In the realm of direct prompting, a study by Mansour et al. (2024) evaluated the capabilities of ChatGPT (`https://chatgpt.com/`), and LLaMA-2 (`https://www.llama.com/llama2/`) in scoring written essays. Through various prompt-engineering tactics, they found that both models exhibited comparable performance, with ChatGPT having a slight advantage. Another study by Stahl et al. (2024) explored zero-shot and few-shot approaches inspired by Chain-of-Thought prompting (Wei et al., 2022) to generate both scores and feedback. Their study found that addressing both tasks simultaneously, rather than independently, enhances the quality of generated feedback and improves scoring performance, although the impact on feedback quality remains limited. Building on this, Pack et al. (2024) highlights while LLMs such as GPT models can achieve human-comparable scoring accuracy, performance varied across time and task type, raising questions about reliability and generalizability. Importantly, they highlight that while LLMs offer a significant advancement over earlier rule/feature-based AWE, systemic validity checks remain necessary to ensure LLMs capture more than superficial patterns in student writing. Additionally, Li and Liu (2024) also demonstrated that LLMs can match or exceed the performance of traditional AWE systems, while also providing more flexible and context-sensitive evaluations of student writing.

Alternatively, fine-tuning LLMs, even older or smaller models, has been proven more effective than directly prompting more advanced LLMs (Li and Ng, 2024). Wang and Gayed (2024) fine-tuned the GPT-3.5 model on a corpus of TOEFL argumentative essays with human benchmark scores and demonstrated that the fine-tuned models achieved higher scoring accuracy than zero-shot prompting of GPT-3.5 or GPT-4, while also being robust when scoring essays from unseen prompts. Gayed (2025) builds on this, by developing a freely accessible web-based application called AiAWE which uses an open-source LLM (Gemma-3-27B-IT) with Low-Rank Adaption (LoRA) adapters and rubric-guided system prompting to score argumentative essays. Through this system, the author was able to show both strong agreement with gold-standard ETS raters across standard metrics in addition to providing actionable deep-level rubric-aligned feedback to users. Another study by Cai et al. (2025) introduced the Rank-Then-Score (RTS) framework, which first ranks essays using a fine-tuned LLM, then scores them based on the rankings which outperformed traditional supervised fine-tuning techniques, particularly in Chinese datasets. Furthermore, Chu et al. (2024) proposed the Rationale-based Multiple Trait Scoring (RMTS) model, which combines prompt-engineering-based LLMs with a fine-tuning-based essay scoring model to provide trait-specific rationales for scores. Their approach enhances the reliability of multi-trait scoring by generating fine-grained explanations aligned with the rubric.

Despite the potential of LLMs to generate rich, formative feedback, the majority of recent AWE research has continued the traditional focus on scoring accuracy (Li and Ng, 2024). This persistent emphasis is likely because feedback evaluation is significantly more time-consuming. Unlike scores, which can be validated against quantitative benchmarks, the quality of feedback must be assessed

through qualitative, pedagogical judgment. As a result, the use of LLMs for feedback remains largely anecdotal rather than proven through rigorous research. The limited number of studies that do cover feedback generation, such as those exploring Chain-of-Thought prompting to produce comments alongside scores (Stahl et al., 2024) or generating trait-specific rationales (Chu et al., 2024), represent important but initial steps. As such, this research aims to develop and validate an AWE system designed to provide both reliable scores and pedagogically valuable, fine-grained feedback.

3 Methods

3.1 LLM choices and procedures

Scoring LLMs Both open-source and commercial models were evaluated for scoring to ensure broad applicability. For the open-source option, LLaMA-3.3-70B-Instruct [4] was selected, the top model in instruction following[5] at the time this research was conducted. For the commercial model, following suggestions in Wang and Gayed (2024), GPT-4o[6] was selected, as it was the latest model from the GPT series that allows supervised fine-tuning at the time this research took place. The fine-tuning prompt can be found at Appendix C.

Surface-level LLMs For the surface-level feedback module based on direct prompting, it is divided into two stages: (1) corrective edits and (2) the generation of explanations for those edits. In a separate study (Labib et al., 2027), DeepSeek-V3-671B[7], GPT-4o and LLaMA-3.3-70B-Instruct were compared in their GEC performance through zero-shot prompting using the W&I+LOCNESS dataset from the BEA 2019 Shared Task (Bryant et al., 2019). Findings (see Appendix G) suggest that the three models performed similarly across three commonly used evaluation metrics in GEC, including ERRANT $F_{0.5}$ (Bryant et al., 2017), GLEU (Napoles et al., 2016), and PT-ERRANT (Gong et al., 2022). Drawing on these findings, GPT-4o was chosen for the corrective edits as it performed better overall compared to LLaMA 3.3, while the small performance margin did not justify the significant infrastructure costs required to host the 671B DeepSeek model. To avoid overcorrection, the prompt to GPT-4o specifically asked it not to introduce stylistic corrections (prompt in Appendix D).

To generate explanations for the corrective edits, qualitative evaluation of three models, Claude 3.7 Sonnet (https://www.anthropic.com/news/claude-3-7-sonnet), GPT-4o, and Deepseek-R1 (https://huggingface.co/deepseek-ai/DeepSeek-R1), found that the explanations from all three models were of a similarly high quality (see Appendix B for example explanations). However, despite its strong performance, Deepseek-R1 was eliminated from consideration due to its significant latency in reasoning, which would negatively impact the user experience in an interactive system. Between the remaining two models, GPT-4o was selected to streamline the technical architecture, as using the same model for both corrections and explanations avoids the overhead of managing a second API call within the surface-level feedback module.

Deep-level LLMs For the deep-level feedback module, supervised fine-tuning with the curated dataset of 90 essays was tested on LLaMA-3.3-70B and GPT-4o. For LLaMA, the model was fine-tuned using an entropy-based loss function over 20 epochs (see Appendix A). The fine-tuning parameters of GPT-4o were unknown as it is a closed commercial model. Alongside fine-tuning we experimented with direct prompting for deep level feedback. Inspired by Chu et al. (2024), this research first conducted a thorough thematic analysis of the teacher feedback in the training subset (resulting multiple traits are shown in Table 1). Based on the two primary categories of macro (paragraph and essay level) and micro (word, phrase and sentence level) feedback, the deep-level feedback was further divided into two pipelines and few-shot prompts were engineered to instruct Claude 3.7, GPT-4o and LLaMA 3.3 to deliver feedback focusing on these subcategories in both pipelines (prompts in Appendix E).

3.2 Validation scheme

3.2.1 Score validation

For the scoring module, the fine-tuned LLaMA-3.3 and GPT-4o models were tested using the test subset of 360 essays and the same prompt as the fine-tuning prompt. To evaluate scoring accuracy, three metrics were employed: Root Mean Square Error (RMSE), Quadratic Weighted Kappa (QWK), and

[4] (https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct)

[5] https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard#/

[6] https://platform.openai.com/docs/models/gpt-4o

[7] https://github.com/deepseek-ai/DeepSeek-V3

Category	Subcategories
Macro feedback	Proper paragraphing; Text length appropriateness; Structure (introduction, body, and conclusion sections)
Micro feedback	Context-dependent grammar issues; Clarity of expression; Coherence between ideas; Argumentation quality and logical flow; Formality and academic register

Table 1: Categories for deep-level feedback

percentage agreement following Wang and Gayed (2024), which is the proportion of essays of exact agreement (same score) and adjacent agreement (a difference of 0.5) with benchmark scores.

3.2.2 Feedback validation

For surface-level feedback, each original essay and its corresponding corrected version were aligned using ERRANT v3.0.0[8] (ERRor ANnotation Toolkit; Bryant et al., 2017), a grammar error annotation tool that automatically detects and categorizes all edit operations. Two expert raters evaluated each identified edit for:

- **Necessity**: Whether the identified text element genuinely requires correction, given the reported tendency of LLMs to over-correct (Katinskaia and Yangarber, 2024).
- **Effectiveness**: Whether the suggested corrective edit effectively addresses the error.

The two raters discussed each edit and provided additional comments for cases where edits were deemed unnecessary or ineffective.

For deep-level feedback, a preliminary check on data integrity and sufficiency of feedback instances was first performed on output from the models. The prompt in the macro feedback pipeline asked the LLMs to specifically give feedback to each paragraph, thus the two raters only assessed whether each macro feedback comment was *effective*. For micro feedback, each was assessed on the dimensions of *necessity* and *effectiveness*, similar to surface-level feedback evaluation. Following the independent evaluations, inter-rater reliability was calculated. For any cases where the two primary raters disagreed, a third expert rater was consulted to deliver the final judgment.

3.3 Dataset and subsets

As this study required a dataset of argumentative essays with benchmark scores and comprehensive feedback, a proprietary dataset of 480 TOEFL Independent Writing test-taker samples with official scores from ETS[9] was obtained. TOEFL Independent Writing is a typical argumentation writing task where test-takers write in response to an essay prompt, such as "*Do you agree or disagree the following statement:...*" The samples in the dataset are evenly distributed under two essays prompts and two ETS raters delivered integer scores from 0-5 based on specific rubrics[10]. Where the discrepancy of the two rater scores was no more than 1, the final score was the average of the two. Otherwise, a third rater was engaged to deliver the final score (Blanchard et al., 2013). As essays scored 0 were excluded, the scores of the essays in the dataset range from 1-5 with 0.5 increments. The score distribution is shown in Table 2.

Score	Prompt 1	Prompt 2	Subtotal
1	1	2	3
1.5	3	2	5
2	18	20	38
2.5	32	25	57
3	65	55	120
3.5	40	45	85
4	32	38	70
4.5	35	28	63
5	14	25	39
Total	240	240	480

Table 2: Score distribution in the dataset

For the scoring module, 120 essays were selected through score-based stratified sampling across the two essay prompts as the fine-tuning subset. And the remaining 360 essays comprised the test subset.

For the feedback module, as no feedback was available in the dataset, this research curated its own datasets for comprehensive feedback. For surface-level feedback in the form of corrective

[8] https://github.com/chrisjbryant/errant

[9] Access to this dataset is restricted to researchers approved by ETS under a limited, non-exclusive, revocable, and non-transferable license.

[10] https://www.nafsa.org/sites/default/files/ektron/files/underscore/regiii/conference/2009/pruner%20indepwrihd-t2.pdf

edits to grammar and mechanics, previous research on GEC has shown that directly prompting LLMs generates satisfactory results (Davis et al., 2024; Labib et al., 2027). As such, supervised fine-tuning and the creation of corresponding datasets were not necessary. For deep-level feedback that requires higher-order thinking, 90 essays from the 480 essays were selected using score-based stratified sampling. Eight experienced university teachers who teach academic writing courses were recruited to provide feedback using Microsoft (MS) Word's comment feature (see Figure 1 for an example of the comments). Before annotation, the teachers received training on the scoring rubrics and were given nine benchmark essays scored from 1 to 5 and annotated by the researchers as reference exemplars. The 90 essays had already been processed for surface-level corrections and the scores had been removed to ensure the annotations would focus solely on deep-level features. After the teachers submitted their annotated essays, another independent teacher reviewed all feedback comments to ensure they were accurate and free from surface-level errors. Their feedback data was then processed to create the fine-tuning subset for deep-level feedback.

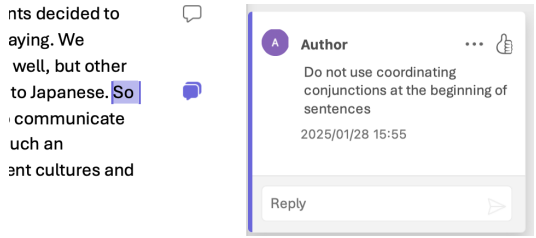


Figure 1: Example of a teacher comment in MS Word

For testing of both surface-level and deep-level feedback, due to the extensive human labor in evaluating such feedback, only 40 essays were selected (through stratified sampling) from the 390 non-annotated essays (dataset summary in Appendix F).

4 Results

4.1 Scoring module

The RMSE, QWK and percentage agreement between the essay scores generated by the two fine-tuned models and benchmark scores are shown in Table 3. The baseline model is the best performing fine-tuned model in Wang and Gayed (2024), a prior study that used the same dataset.

Results show our two fine-tuned models improve on the baseline (Wang and Gayed, 2024), with GPT-4o consistently performing the best across all three evaluation metrics. Specifically, both our fine-tuned models achieved QWK scores above 0.8, surpassing the commonly accepted threshold for near-perfect agreement (Sim and Wright, 2005). For context, ETS considers a QWK of 0.7 to be sufficient for reliable scoring of TOEFL Independent Writing tasks by its *e-rater* system (Williamson et al., 2012). Although ETS does not specify a formal threshold for RMSE, the observed discrepancies of 0.44 (GPT-4o) and 0.53 (LLaMA) from benchmark scores are reasonable, given that ETS permits up to a one-point difference between two human raters (Blanchard et al., 2013), which results in a final averaged score deviating by 0.5 from each individual score.

4.2 Surface-level feedback module

After the 40 essays in the test subset were corrected in the surface-level module, ERRANT tagged 2049 edit operations. Human evaluation deemed 1985 edits necessary (96.88%), out of which 1970 were deemed both necessary and effective (96.14%). Among the unnecessary edits, two salient patterns have been identified from rater comments. First, GPT-4o seems to prefer British use than American use (N=7). For example, it changed *favorite* to *favourite* and moved a period or a comma outside a closing quotation mark (e.g., *"...store."* to *"...store"*.) to match the British style. The second pattern is related to comma additions (N=15). In particular, GPT-4o preferred adding the Oxford comma before the last item in a list of nouns (N=10; e.g., *A, B and C* into *A, B, and C*), which was not deemed an error by raters.

4.3 Deep-level feedback module

4.3.1 Preliminary check

Preliminary check showed that neither fine-tuned models generated appropriate output data. The fine-tuned GPT-4o model produced incomplete output for every inference as it generated so many comments that the output token exceeded the context window of 8,000 tokens. Meanwhile, the fine-tuned LLaMA-3.3-70B model produced formatting errors, that prevented its output from being parsed as valid JSON. The most severe issues were structure errors, such as missing key-value delimiters and unclosed braces. As such, feedback from fine-tuned models were excluded from human evaluation.

For directly prompted models, preliminary check on feedback instances revealed that the micro feed-

Fine-tuned Model	RMSE	QWK	Percent (absolute)	Percent (adjacent)	Percent (total)
GPT-4o	0.44	0.84	0.45	0.48	0.93
LLaMA	0.53	0.81	0.41	0.45	0.86
Baseline (Wang and Gayed, 2024)	0.57	0.78	0.33	0.52	0.85

Table 3: Performance metrics of fine-tuned models on the test subset

back comments from LLaMA-3.3-70B and GPT-4o were insufficient. Of the 40 test essays, LLaMA-3.3-70B generated 336 comments and GPT-4o generated 368 comments, the averages of both were below 10. Claude 3.7, however, generated 630 comments. To verify that this was a difference in quality and not just verbosity, qualitative review of three randomly selected essays was conducted, which confirmed that the additional comments were indeed more fine-grained and pedagogically necessary. Given the intensive human labor required, human validation efforts were focused on the feedback from the most promising model, Claude 3.7.

4.3.2 Macro feedback

The 40 essays in the test subset contained 201 paragraphs and thus Claude 3.7 generated 201 macro feedback comments. The contingency table between the two raters on the effectiveness of the comments is shown in Table 4.

Rater A / Rater B	Effective	Ineffective
Effective	179	9
Ineffective	13	0

Table 4: Macro comment evaluation by two raters

To assess inter-rater reliability, Gwet's $AC1$ coefficient (Gwet, 2008) was employed, which ranges from -1 to $+1$ similar to Cohen's Kappa. This measure was selected over Cohen's Kappa due to its robustness against the "Kappa paradox", where high observed agreement coincides with low kappa values in cases of skewed marginal distributions (Wongpakaran et al., 2013). The $AC1$ coefficient was calculated to be 0.89, with a standard error (SE) of 0.03 and a 95% confidence interval (CI) ranging from 0.82 to 0.93. As there is no commonly accepted interpretation of Gwet's $AC1$ coefficient, we relied on the interpretation of Cohen's Kappa to determine the degree of agreement (Landis and Koch, 1977), under which the observed value indicates almost perfect agreement ($p < .001$).

After a third rater resolved the disagreement of the two raters, the number of effective comments was finalized as 187 (93.03%) and that of ineffective comments, 14 (6.97%). For the 14 ineffective comments, remarks from raters revealed that Claude 3.7 was not flexible enough to handle unanticipated input variations. For example, one test-taker included a title in their essay, a component not required in the writing task, and Claude 3.7 misinterpreted it as the first paragraph. In another case, some test-takers improperly formatted each sentence as a separate paragraph, which led to repeated comments that a paragraph was too short and thus should be combined with the previous one (N=8). In addition, some test-takers were unable to complete the task within the allotted time. Claude 3.7 failed to recognize the time constraint of such unfinished responses and instead provided detailed guidance on composing conclusions (N=2). Also very interesting is that Claude 3.7 was not aware of the test-taker situation where they didn't have access to external resources and recommended incorporating research evidence and citations to strengthen arguments (N=1).

4.3.3 Micro feedback

For the same 40 essays, Claude 3.7 generated 630 micro feedback comments. The contingency table between the two raters on the necessity of the comments is shown in Table 5, while that on the effectiveness of necessary comments mutually agreed by two raters is shown in Table 6.

Rater A / Rater B	Necessary	Unnecessary
Necessary	579	28
Unnecessary	22	1

Table 5: Necessity of micro feedback by two raters

Rater A / Rater B	Effective	Ineffective
Effective	577	1
Ineffective	1	0

Table 6: Effectiveness of judgment for micro feedback

Using Gwet's $AC1$ (Gwet, 2008), it is shown that the agreement on necessity was statistically

significant ($AC1$= 0.91; SE=0.01; 95% CI, [0.89, 0.94]; $p < .001$), as well as the agreement on effectiveness ($AC1$= 1.00; SE=0.00; 95% CI, [0.99, 1.00]; $p < .001$). After a third rater resolved the disagreements, the number of necessary comments was finalized as 600 (95.24%). Among the necessary comments, 596 were deemed effective while 4 were ineffective, and thus the proportion of both necessary and effective comments is 94.69%.

Based on rater comments, one salient pattern was observed: some micro comments overlapped with macro ones. Claude 3.7 at times expands the comment for a sentence to cover the whole paragraph (N=18). For instance, when the first sentence in an introduction paragraph did not properly set the background of the topic, Claude 3.7 would comment that there was a lack of proper background and then went on to give more feedback on how to write a good introduction paragraph.

5 Discussion

As shown in Figure 2, the architecture was finalized by selecting LLMs tailored to each module: fine-tuned GPT-4o model for the scoring module; direct prompting with GPT-4o for the surface-level feedback module; and Claude 3.7 for both pipelines in the deep-level feedback module. To mimic human feedback mechanisms, an interactive UI was also developed (details in Appendix H).

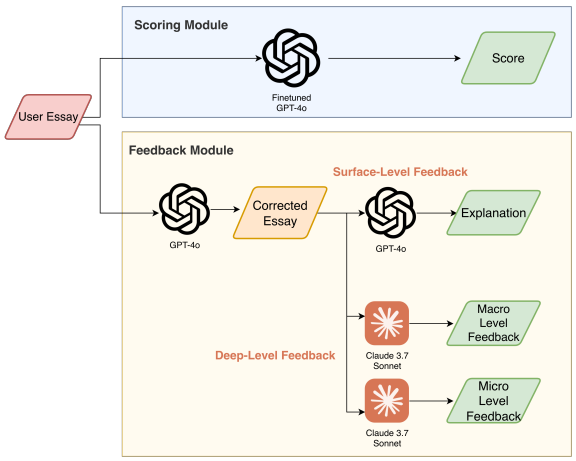


Figure 2: Final system architecture

The findings in this study are supported by Stahl et al. (2024) and Chu et al. (2024), who have demonstrated that specialized models handling different aspects of writing evaluation can produce better assessments than unified approaches.

Human evaluation of surface-level feedback shows that GEC tasks can be addressed effectively by well-designed prompts without the need for fine-tuning, which is supported by the findings of Zeng et al. (2024). Particularly, as shown by the high precision rate of 96.14%, we successfully minimized over-correction through our targeted prompt, an issue previously identified as a key limitation of LLM-prompting-based GEC approach (Katinskaia and Yangarber, 2024; Davis et al., 2024).

Human validation of deep-level feedback suggests that well-structured few-shot prompting can generate effective and structured comments on higher-order thinking, and may even be more effective than fine-tuned models. Here, fine-tuning did not yield satisfactory results, because the fine-tuned GPT model provided feedback to almost every word, phrase, and sentence, most of which were unnecessary. As the fine-tuning parameters were unknown, it can only be speculated that supervised fine-tuning with non-deterministic output is not well-suited for GPT-4o. For the fine-tuned LLaMA model, despite the data parsing issues, it was found that the feedback comments were actually quite similar to those in the fine-tuning subset, though still not so good as those from Claude 3.7. Data formatting issue is likely an inherent issue with LLaMA, as previously highlighted by Yao et al. (2025). If the data formatting issue can be addressed, it may be possible to effectively use fine-tuned LLaMA for generating structured feedback.

Pedagogical implications The system can benefit students engaged in argumentative writing. They can receive prompt feedback not available in traditional classroom settings, where it usually takes teachers days to provide comprehensive feedback on student writing. Teachers can also use this tool to help with their workload, particularly for surface-level feedback. Instead of spending time correcting basic grammatical and mechanical errors, they can focus more on guiding students' higher-order thinking in argumentative writing, aided by the feedback generated by the system.

6 Conclusion

By decomposing AWE into scoring, surface-level feedback, and deep-level feedback, *WrAFT* achieved strong performance in scoring on TOEFL writing dataset with satisfactory results on human validation of system-generated feedback. These findings provide a strong foundation for research in AWE systems that provide scoring and feedback.

Limitations

There are a few limitations in the design of this study. First, the system was developed using a proprietary dataset that has only been used in one prior AWE study. This limits the scope of direct comparisons on scoring performance. Second, in the validation scheme, a key limitation is that human evaluation only focused on precision without considering recall. For both surface-level and deep-level feedback, the necessity and effectiveness of feedback were assessed but it was not evaluated whether the system captured all errors that should have been identified or all text elements that should have required feedback comments. A more robust evaluation would require establishing a gold standard of all possible errors and feedback opportunities against which system performance could be measured. Unfortunately, the extensive human resources required to achieve this was beyond our capacity. But this research provides a necessary contribution for future studies to focus on measuring actual impact on student learning outcomes by including experimental studies tracking writing improvement over time when using the system compared to traditional feedback methods.

There are limitations with the system as well. First, the system is limited in its generalizability, as it was developed and tested exclusively on argumentative essays with writing prompts. It cannot be directly applied to other types of argumentative writing, such as source-based writing where writers are required to incorporate or respond to source information. Second, the feedback offered by the system is solely in English, creating barriers for learners who might struggle to comprehend feedback comments. Future iterations should incorporate multilingual feedback capabilities, including translation of comments into learners' first languages to facilitate comprehension and implementation of suggestions.

From a practical implementation perspective, there are limitations arising from the reliance on commercial API services. There are potential scalability issues related to cost, rate limits, and long-term sustainability. In addition, constant changes to commercial models may impact system performance. Data privacy concerns also emerge when student writing is processed through third-party services.

Declaration on Generative AI

During the preparation of this work, the authors used generative AI assistants for grammar and spelling checking. All suggestions were reviewed and edited by the authors, who take full responsibility for the content of the publication.

References

Yigal Attali and Jill Burstein. 2006. Automated essay scoring with e-rater v.2. *Journal of Technology, Learning, and Assessment*, 4(3).

John Bitchener and Ute Knoch. 2008. The value of written corrective feedback for migrant and international students. *Language Teaching Research*, 12(3):409–431.

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. Toefl11: A corpus of non-native english. *ETS Research Report Series*, 2013:i–15.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical error correction: A survey of the state of the art. *Computational Linguistics*, 49(3):643–701.

Yida Cai, Kun Liang, Sanwoo Lee, Qinghan Wang, and Yunfang Wu. 2025. Rank-then-score: Enhancing large language models for automated essay scoring. *arXiv preprint arXiv:2504.05736*.

SeongYeub Chu, JongWoo Kim, Bryan Wong, and MunYong Yi. 2024. Rationale behind essay scores: Enhancing s-llm's multi-trait essay scoring with rationale generated by llms. *arXiv preprint arXiv:2410.14202*.

Christopher Davis, Andrew Caines, Øistein E. Andersen, Shiva Taslimipoor, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. Prompting open-source and commercial language models for grammatical error correction of English learner text. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages

11952–11967, Bangkok, Thailand. Association for Computational Linguistics.

Paul Deane. 2013. On the relation between automated essay scoring and modern views of the writing construct. *Assessing Writing*, 18(1):7–24.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada. Association for Computational Linguistics.

John Maurice Gayed. 2025. Aiawe (version 1.0) [web application]. Accessed October 22, 2025.

Peiyuan Gong, Xuebo Liu, Heyan Huang, and Min Zhang. 2022. Revisiting grammatical error correction evaluation and beyond. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6891–6902, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kilem L Gwet. 2008. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48.

Kenji Imamura, Kuniko Saito, Kugatsu Sadamitsu, and Hitoshi Nishikawa. 2012. Grammar error correction using pseudo-error sentences and domain adaptation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 388–392, Jeju Island, Korea. Association for Computational Linguistics.

Cancan Jin, Ben He, Kai Hui, and Le Sun. 2018. TDNN: A two-stage deep neural network for prompt-independent automated essay scoring. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1088–1097, Melbourne, Australia. Association for Computational Linguistics.

Anisia Katinskaia and Roman Yangarber. 2024. GPT-3.5 for grammatical error correction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7831–7843, Torino, Italy. ELRA and ICCL.

Adnan Labib, Qiao Wang, Yixuan Huang, and Zheng Yuan. 2027. Multi-dimensional evaluation of llms for grammatical error correction. In *Artificial Intelligence in Education*, pages 536–545, Cham. Springer Nature Switzerland.

Thomas K. Landauer, Darrell Laham, and Peter W. Foltz. 2003. Automated scoring and annotation of essays with the intelligent essay assessor. In *Automated Essay Scoring: A Cross-disciplinary Perspective*, pages 87–112. Lawrence Erlbaum Associates.

J. Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Jianwei Li and Jiahui Wu. 2023. Automated essay scoring incorporating multi-level semantic features. In *International Conference on Artificial Intelligence in Education*, pages 206–211. Springer.

Shengjie Li and Vincent Ng. 2024. Automated essay scoring: A reflection on the state of the art. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17876–17888, Miami, Florida, USA. Association for Computational Linguistics.

Wenchao Li and Haitao Liu. 2024. Applying large language models for automated essay scoring for non-native japanese. *Humanities and Social Sciences Communications*, 11(1):1–15.

Sha Liu and Antony Kunnan. 2015. Investigating the application of automated writing evaluation to chinese undergraduate english majors: A case study of writetolearn. *Assessing Writing*, 24:1–15.

Watheq Mansour, Salam Albatarni, Sohaila Eltanbouly, and Tamer Elsayed. 2024. Can large language models automatically score proficiency of written essays? *arXiv preprint arXiv:2403.06149*.

Atsushi Mizumoto and Masaki Eguchi. 2023. Exploring the potential of using an ai language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2(2):100050.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2016. Gleu without tuning. *arXiv preprint arXiv:1605.02592*.

Austin Pack, Alex Barrett, and Juan Escalante. 2024. Large language models and automated essay scoring of english language learner writing: Insights into validity and reliability. *Computers and Education: Artificial Intelligence*, 6:100234.

Ellis B Page. 1967. Grading essays by computer: Progress report. In *Proceedings of the invitational Conference on Testing Problems*.

Ellis Batten Page. 2003. Project essay grade: PEG. In Mark D. Shermis and Jill C. Burstein, editors, *Automated Essay Scoring: A Cross-Disciplinary Perspective*. Lawrence Erlbaum Associates, Mahwah, NJ.

Les Perelman. 2014. When "the state of the art" is counting words. *Assessing Writing*, 21:104–111.

Donald E Powers, Jill C Burstein, Martin Chodorow, Mary E Fowles, and Karen Kukich. 2002. Stumping e-rater: challenging the validity of automated essay scoring. *Computers in Human Behavior*, 18(2):103–134.

Jim Ranalli and Junko Yamashita. 2021. L2 student engagement with automated feedback on writing. *System*, 99:102512.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically rich languages: The case of russian. *Transactions of the Association for Computational Linguistics*, 7:1–17.

Mark D Shermis and Joshua Wilson. 2024. Introduction to automated essay evaluation. In *The Routledge international handbook of automated essay evaluation*, pages 3–22. Routledge.

Julius Sim and Chris C Wright. 2005. The kappa statistic in reliability studies: Use, interpretation, and sample size requirements. *Physical Therapy*, 85(3):257–268.

Maja Stahl, Leon Biermann, Andreas Nehring, and Henning Wachsmuth. 2024. Exploring llm prompting strategies for joint essay scoring and feedback generation. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 234–245.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 1882–1891, Austin, Texas, USA. Association for Computational Linguistics.

John Truscott. 1996. The case against grammar correction in l2 writing classes. *Language Learning*, 46(2):327–369.

Qiao Wang and John Gayed. 2024. Effectiveness of large language models in automated evaluation of argumentative essays. *Computer Assisted Language Learning*, 37(1):1–25.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

David M Williamson, Xiaoming Xi, and F Jay Breyer. 2012. A framework for evaluation and use of automated scoring. *Educational Measurement: Issues and Practice*, 31(1):2–13.

Nahathai Wongpakaran, Tinakon Wongpakaran, Danny Wedding, and Kilem L Gwet. 2013. A comparison of cohen's kappa and gwet's ac1 when calculating inter-rater reliability coefficients: a study conducted with personality disorder samples. *BMC Medical Research Methodology*, 13(1):61.

Jiashu Yao, Heyan Huang, Zeming Liu, Haoyu Wen, Wei Su, Boao Qian, and Yuhang Guo. 2025. Reff: Reinforcing format faithfulness in language models across varied tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25660–25668.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California. Association for Computational Linguistics.

Zheng Yuan, Shiva Taslimipoor, Christopher Davis, and Christopher Bryant. 2021. Multi-class grammatical error detection for correction: A tale of two systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8722–8736, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Min Zeng, Jiexin Kuang, Mengyang Qiu, Jayoung Song, and Jungyeul Park. 2024. Evaluating prompting strategies for grammatical error correction based on language proficiency. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6426–6430, Torino, Italia. ELRA and ICCL.

A Model and Training Configuration

The model was developed by applying full-parameter fine-tuning to LLaMA 3.3 70B. It was trained to generate fine-grained feedback on student argumentative essays. The training parameters are presented in Table 7.

Parameter	**Value**
Base model	LLaMA 3.3 70B
Fine-tuning method	Full-parameter fine-tuning
Number of epochs	20
Batch size	32
Learning rate	3×10^{-6}
Optimizer	AdamW
Learning-rate scheduler	Cosine scheduler

Table 7: Training parameters

The tokenizer parameters are presented in Table 8.

Parameter	**Value**
Tokenizer class	`PreTrainedTokenizerFast`
Configured maximum length	8,192 tokens
Model input fields	`input_ids`, `attention_mask`
Padding side	Right
Beginning-of-sequence token	`<|begin_of_text|>`
End-of-sequence token	`<|eot_id|>`
Padding token	`<|finetune_right_pad_id|>`
Additional special tokens	`<|eot_id|>`, `<|eom_id|>`
Clean up tokenization spaces	True
Split special tokens	False

Table 8: Tokenizer parameters

B Explanations of surface-level corrective edits by GPT-4o, Claude 3.7 and DeepSeek-R1

ID	Original / Correction	Claude 3.7 Sonnet	GPT-4o	DeepSeek-R1
0	*serve → serves*	Subject–verb agreement error. "Academic writing" is singular and requires "serves".	Subject–verb agreement error. "Writing" is singular, so verb should be "serves".	Subject–verb agreement error. "Academic writing" is singular and requires "serves".
1	*corner → cornerstone*	Word choice improvement. "Cornerstone" conveys the foundation or essential component of scholarly communication.	"Corner" is semantically incorrect; "cornerstone" is the correct metaphor for something fundamental.	"Corner" is incomplete; "cornerstone" better conveys fundamental importance.
2	*enable → enables*	Subject–verb agreement. "It" is singular and requires "enables".	Subject–verb agreement. "It" is singular, so verb should be "enables".	Subject–verb agreement. "It" is singular and requires "enables".
3	*they're → their*	Incorrect homophone. "They're" (they are) should be replaced with possessive "their".	Incorrect homophone. "They're" is incorrect; should be possessive "their".	Wrong word choice. "They're" (they are) vs. "their" (possessive).
4	*accurate → accurately*	Incorrect part of speech. Adverb "accurately" modifies verb "convey".	Adverb needed. "Accurate" → "accurately".	Adverb required. "Accurate" should be "accurately" to modify verb.
5	*spends → spend*	Subject–verb agreement. Plural "Scholars" requires plural "spend".	Subject–verb agreement. "Scholars" is plural → "spend".	Subject–verb agreement. "Scholars" plural → "spend".
6	*hone → honing*	Verb form consistency. "Honing" maintains parallel structure with "learning".	Verb form error. Change to gerund "honing" for parallelism.	Gerund required after "years". "Honing" shows ongoing action.
7	*learn to balance technic → learning to balance technical*	Verb form consistency & spelling. "Learning" parallels "honing"; "technic" → "technical".	Grammar and clarity. "Learn to balance technic" → "learning to balance technical".	Parallel structure and word correction. "Learning" matches "honing"; "technical" corrects "technic".
8	*discipline → disciplines*	Pluralization. Remove comma; use plural "disciplines".	Number agreement. "Many" → plural "disciplines".	Plural required. "Many" → plural "disciplines".
9	*help → helps*	Subject–verb agreement. "Process" is singular → "helps".	Subject–verb agreement. "The peer review process" is singular → "helps".	Subject–verb agreement. "Process" is singular → "helps".
10	*distribute → distributed*	Verb form. Passive voice requires past participle "distributed".	Verb tense. Use past participle "distributed" in passive construction.	Passive voice requires past participle. "Is distributed" correct form.

Table 9: Explanations of corrective edits generated by Claude 3.7 Sonnet, GPT-4o, and DeepSeek-R1.

C Fine-tuning and inference prompt for the scoring module

```
<instructions>
As a language expert, your task is to evaluate
argumentative essays on a scale of 0 to 5
(with 0.5 increments) based on the rubrics below.

<rubric>
[Complete rubrics here]
</rubric>
</instructions>

<input_data_structure>
{{
  "essay_prompt": "...", // The essay prompt
  "essay_text": "...", // The essay text
}}
</input_data_structure>

<output_data_structure>
{{
  "score": 0.5 // The score
}}
</output_data_structure>

Here's the input data:
<input_data>
{{
  "essay_prompt": {essay_prompt},
  "essay_text": {essay_text}
}}
</input_data>
```

D Surface-level prompt

```
<instructions>
You are an English linguist and your task is to
correct the grammatical
and mechanical errors in an English essay. Do not
alter word choices
unnecessarily (e.g., replacing words with
synonyms) or make stylistic
improvements. Keep the original paragraphing and
DO NOT remove or add
any paragraphs.

Requirements:
1. The output should be in JSON format.
2. The output should be in the same format as the
output_data_structure.
</instructions>

<output_data_structure>
{{
  "corrected_paragraphs": [
    "...", // The corrected text of the first
    paragraph
    "...", // The corrected text of the second
    paragraph
    ...
  ]
}}
</output_data_structure>

Here's the input data:
<input_data>
{{
  "essay_paragraphs": {essay_paragraphs}
}}
</input_data>
```

E Deep-level prompts

E.1 Prompt for macro feedback

Instructions:

You are an English teacher in argumentative writing and your job is to give macro feedback to a student's argumentative essay structure. An essay should typically include three major sections: Introduction, Body and Conclusion.

The Introduction part should open with some background of the essay topic, followed by a thesis statement of the author and possibly end with a predictor of sub-points for later paragraphs.

The body part should contain detailed argumentation of sub-points of the thesis statement, probably divided into several paragraphs, with each paragraph having a clear topic sentence.

The conclusion part should efficient summaries the essay and preferably end with a forward looking statement.

You need to first identify the three sections, and then give feedback to each paragraph. Specifically, you need to identify the issues of essay structure targeting specific paragraph(s).

- In the introduction paragraph, there may be a lack of background information, thesis statement (main claim) or a predictor for following content, or the introduction does not effectively introduce the topic;
- In body paragraphs, there may be a lack of topic sentences and issues in argumentation. Specifically for argumentation, focus on ideas that are irrelevant to the thesis statement or the claim; repeated and thus redundant ideas, underdeveloped arguments where there is a lack of evidence and support, e.g, in the form of examples, or the support is poor or irrelevant; where there is a lack of warrant; where the author's stance is confusing; and where handling of counter-argument is poor, etc.
- In the conclusion paragraph, there may be undiscussed ideas, ideas that shouldn't be placed in the conclusion, such as further arguments or counter-arguments, and lack of effective summary of main points, etc.
- For all the three section, please consider whether proper paragraphing is there. Improper paragraphing may include putting content that should be in the body part, e.g., de-

velopment of main points, examples, counter-arguments, etc., with the introduction or conclusion part, and vice versa.

For each paragraph, write down feedback comments, including suggestions for solutions, possibly with an example of how to rewrite. Please make sure the number of comments your return is equal to the number of paragraphs and use the second person pronoun "you" to sound like you are talking to the student directly. Do not specifically divide your comments into "issue:" and "solution:" points; instead write it as a coherent piece of text.

Input and Output

You will be given the essay prompt, the essay text of the student in the user input. Your response should be in the following JSON format:

```
  {
"comments": [
  {
    "id": 0, // The id of the comment
    "Paragraph": , // The number of the
    specific paragraph
    "data": "...", // The teacher's feedback
    comment
  },
  ...
]
}
```

E.2 Prompt for micro feedback

You are an English teacher specializing in argumentative writing. Your task is to provide detailed micro feedback on a student's argumentative essay, focusing on grammar, clarity, formality, and coherence. You will analyze the essay and provide specific, actionable feedback for improvement.

```
Here is the essay prompt:
<essay_prompt>
{{essay_prompt}}
</essay_prompt>

And here is the student's essay:
<essay_text>
{{essay_text}}
</essay_text>
```

Please carefully read and analyze the essay, then provide feedback according to the following guidelines:

1. Focus Areas:

a) Grammar:

- Identify and correct grammatical and mechanical errors

- Address context-dependent grammar issues (e.g., appropriate tense, aspect)

b) Clarity:

- Highlight unclear, inaccurate, or redundant meanings

- Point out lengthy or confusing sentence structures

- Identify poor or unclear referents c) Formality:

- Note expressions requiring hedging

- Identify informal language use (e.g., coordinating conjunctions at sentence beginnings, subjective wording, informal interrogatives, colloquial language, second-person pronouns, contractions)

d) Coherence:

- Evaluate connections between sentences and ideas

- Assess transitions between paragraphs

- Check consistency in person perspective

2. Feedback Structure:

For each issue you identify, provide the following information:

- The type of issue (grammar, clarity, formality, or coherence)

- The specific paragraph number (starting from 0) and the micro text span (preferably words or phrases, avoiding clauses and sentences when possible)

- A clear explanation of the issue

- A suggested solution

3. Additional Guidelines:

- If similar issues occur multiple times, acknowledge this in your feedback (e.g., "As mentioned previously...")

- Address the student directly using the second-person pronoun "you"

- Write your feedback as coherent paragraphs rather than dividing into explicit "issue" and "solution" sections

- Ensure your comments are not repetitive

- Highlight the shortest text spans possible for feedback, unless a longer span is absolutely necessary

4. Output Format:

Your final output should be in JSON format, structured as follows:

```
{
  "comments": [
 {
   "id": 0, // The id of the comment
        "Paragraph": , // The number of the
        specific paragraph,
                  // starting from 0
   "start": 22, // The start character
   position of the
                // highlighted text in the
                paragraph
```

```
    "end": 36, // The end character position of
    the
               // highlighted text in the
               paragraph
    "highlighted_text": "...", // The
    highlighted text
                              // span to be
                              commented on
    "data": "...", // The teacher's feedback
    comment
                  // (explanation of the issue
                  and solution)
    "type": "..." // The type of issue, one of
    "grammar",
                  // "clarity", "formality",
                  "coherence"},
     ...// More comments...
  ]
}
```

Before providing your final JSON output, wrap your analysis inside <essay_analysis> tags in your thinking block. Follow these steps:

1. Grammar analysis:
- List specific grammatical errors found
- Note any context-dependent grammar issues
2. Clarity analysis:
- Identify unclear or redundant phrases
- List confusing sentence structures
- Note any poor referents
3. Formality analysis:
- List instances of informal language use
- Identify expressions that require hedging
4. Coherence analysis:
- Evaluate connections between sentences and paragraphs
- Note any inconsistencies in perspective
- Avoid commenting on general essay structure (general comments on introduction, body and conclusion paragraphs)

For each issue, consider multiple perspectives and potential solutions. This thorough analysis will ensure accurate and helpful feedback.

Begin your analysis now, followed by the JSON-formatted feedback. Your final output should consist only of the JSON-formatted feedback and should not duplicate or rehash any of the work you did in the thinking block.

F Dataset Splits

The dataset splits used for fine-tuning and testing are shown in Table 10

G GEC System performance

The GEC system performance, as stated in (Labib et al., 2027), is provided in Table 11

H Interactive UI

The feedback page shows the predicted score of an essay and includes a surface-level feedback tab and a deep-level feedback tab. The surface-level feedback tab allows three display modes of surface-level feedback: (1) the original writing with erroneous elements highlighted, (2) a combination of the original and corrected writing with edit operations shown, and (3) the corrected writing with corrections highlighted (see Figure 3). For the deep-level feedback tab, a comment-based interface design was adopted, modeled after Microsoft Word's Comment feature. Macro-level comments are displayed in a left-side panel and are aligned with paragraph boundaries. Micro issues under coherence, clarity, grammar, argumentation and formality are highlighted within the essay text using color-coded categories. When a user hovers the mouse over a highlighted section, the corresponding feedback comment appears on the right-side panel (see Figure 4).

Module - Data Subset	Number of Essays	Data Content
Scoring - Fine-tune	120	Essay prompts, essays, scores, rubrics
Scoring - Test	360	Essay prompts, essays, scores, rubrics
Surface-level - Test	40	Essay prompts, essays
Deep-level - Fine-tune	90	Essay prompts, essays, target text elements, comments
Deep-level - Test	40	Essay prompts, essays (same with surface-level testing)

Table 10: Datasets used in fine-tuning and testing

Model	ERRANT	GLEU	PT-ERRANT
DeepSeek-V3-671B	0.4926	0.7677	0.5666
GPT-4o	0.4592	0.7420	0.5484
LLaMA-3.3-70B-Instruct	0.4826	0.7345	0.4973

Table 11: GEC Performance of prompting SOTA LLMs

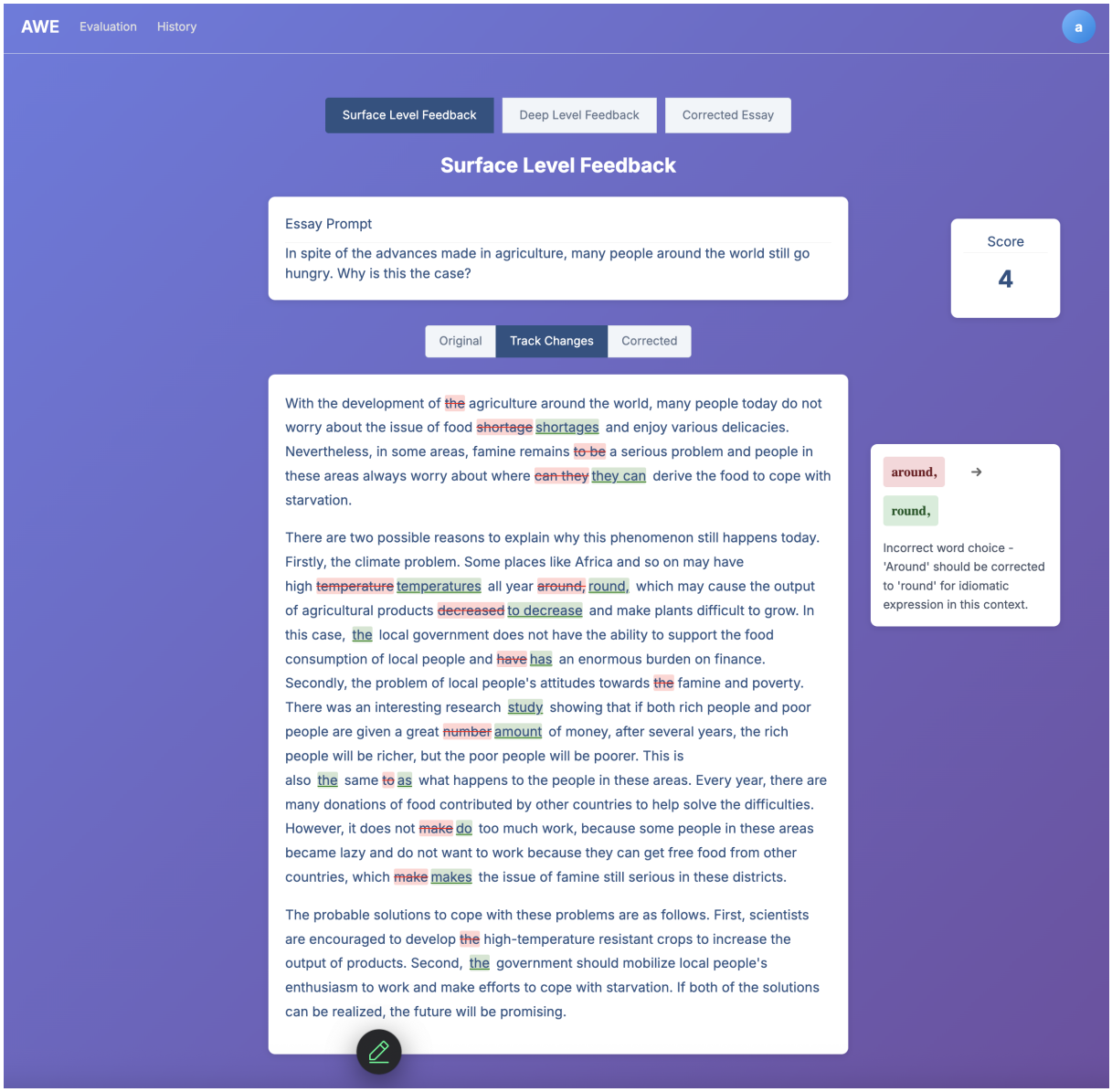


Figure 3: Surface-level feedback page - track changes

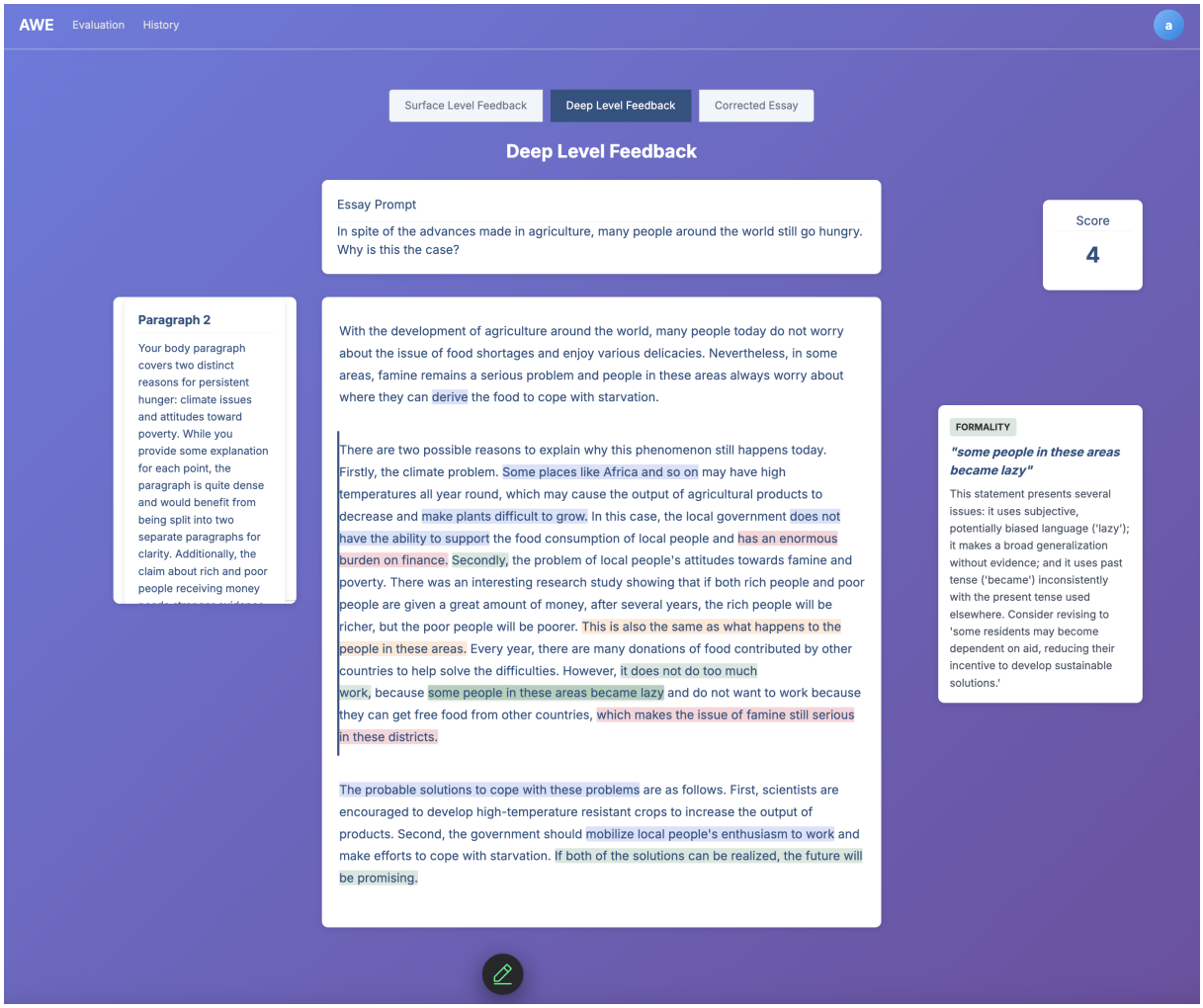


Figure 4: Deep-level feedback page

Beyond Accuracy: What Language-Subspace Interventions Capture, Remove, and Improve

Shaanil Punglia[1], Dadhichi Telwadkar[1], Vaibhav Rathore[2,1], Siddharth Sureka[1], Neha Bhargava[1]
[1]Motilal Oswal Financial Services, India
[2]Indian Institute of Technology Bombay, India

Abstract

Language-subspace interventions aim to improve multilingual reasoning by suppressing language-related components in mid-layer activations. Using MATH and MGSM across four open-weight LLMs, we ask three questions: do estimated language subspaces capture translation-associated variation? Does projecting them out remove language information? And do they improve accuracy without compromising response language? For the first question, we build subspaces using the existing MEAN-SVD construction and our DIFFPCA. Both capture substantial translation-associated variation and show similar patterns across model depth. For the second, we project out each subspace and train linear probes to classify language from the projected activations. The probes still identify language almost perfectly, showing that neither subspace exhaustively captures language information. For the third, we evaluate several intervention configurations on Qwen2.5-7B, by varying subspace creation methods and intervention operators. We find that the intervention that most improves MGSM accuracy sacrifices response-language fidelity, with only one intervention improving joint accuracy–answers that are correct and in the target language–slightly, by 1.60 points. Interestingly, subtle changes in the prompt alone can reproduce the accuracy gains albeit with significant damage to language fidelity. We argue that causal interventions must be judged on more than accuracy, to fully account for language substitution trade-offs.

1 Introduction

Mechanistic interventions can produce the desired metric improvement for an unintended reason. In multilingual reasoning, an intervention can raise accuracy simply by moving generation into English or Chinese. The answer may be mathematically correct, yet the model has silently changed the task. Multilingual reasoning therefore offers a useful stress test for causal interpretability: a behavioral gain need not validate the mechanism an intervention was designed to target.

Language-subspace ablation makes the distinction concrete. Zhao et al. (2025a) estimate a low-rank span from multilingual activation means, suppress it in middle layers, and amplify it near the output. This raises three separate questions: Does the span track language variation? Does removing the span eliminate linearly decodable language information? Does editing it improve the requested behavior? These are questions of geometry, containment, and behavioral effect, and each requires its own evidence.

We test those properties separately. We compare mean-based and translation-difference subspaces, probe what projection disrupts and what language information survives, and evaluate generation with *joint accuracy*: an answer must be correct and remain in the problem language. Additionally, we construct prompt-language controls that leave the problem untouched but switch its worked examples and answer cue to English or Chinese, testing whether contextual language alone can reproduce the apparent gain.

The evidence reveals a consistent split. The subspaces summarize translation structure without containing language exhaustively, and the accuracy-maximizing edit performs worse once output language is part of the task. Prompt-language switches reproduce the trade-off, suggesting that the gain comes from changing the elicited linguistic regime rather than from activation editing itself. The broader lesson is methodological: a clean intervention in activation space is not yet a clean causal explanation. A strong causal account requires evidence connecting the edited geometry, the information affected, and the resulting behavioral change.

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 19–32
November 9, 2026 ©2026 Association for Computational Linguistics

2 Related work

Language-subspace interventions grow from evidence that multilingual representations contain both shared and language-sensitive geometry. Low-rank analyses recover language-associated structure (Xie et al., 2022; Chang et al., 2022), while neuron-level studies identify units that predict or steer output language (Tang et al., 2024; Zhao et al., 2024). Intermediate states can also favor English concept realizations even when the input is not English (Wendler et al., 2024). Activation-patching results separately suggest that concept representations can be partially decoupled from output language (Dumas et al., 2025). Together, these results motivate compact language directions, but do not show that such directions contain language exhaustively or control how they are used.

Activation steering turned this descriptive geometry into an intervention surface: low-dimensional edits can redirect high-level behavior (Turner et al., 2023; Rimsky et al., 2024). Zhao et al. (2025a) turn this idea into an intervention: they estimate a language subspace from translated mathematics problems, suppress it in middle layers, amplify it near the output, and evaluate its effects across multilingual tasks. Their work is the direct premise of ours, while leaving open how effects depend on the subspace construction and removal operator.

Removal, however, is not automatically explanation. Probe accuracy can reflect readout capacity rather than representation quality (Hewitt and Liang, 2019), and information accessible to a readout need not be behaviorally used by the model (Elazar et al., 2021). Iterative Nullspace Projection (INLP) introduced repeated removal of probe directions to guard representations from linear prediction (Ravfogel et al., 2020), and amnesic probing connected removal to behavioral counterfactuals (Elazar et al., 2021). LEACE later provided closed-form linear erasure (Belrose et al., 2023); subsequent comparisons stress that a useful causal intervention must remove the target selectively rather than broadly distort the representation (Dobrzeniecka et al., 2025). This lineage motivates our retrained probes and operator comparison: defeating a fixed probe, exhausting linearly decodable information, and changing generation are distinct claims.

Behavioral evaluation adds a final complication. In multilingual mathematics, the language of demonstrations and cues affects accuracy, and determines the language the model is prompted to reason in (Shi et al., 2023). Accordingly, an intervention cannot be judged by correctness alone when remaining in the problem language is part of the task. Prior work motivates each link in this chain, but usually examines them separately. We compare mean-based and translation-difference subspaces, expand the operator axis beyond orthogonal projection, probe what survives removal, evaluate complete task success, and ask whether prompt language reproduces the same pattern without activation editing.

3 Setup

3.1 Datasets

We use the same mathematical problems expressed in multiple languages, so that problem content stays fixed while language changes. The MATH dataset contains 12,500 competition-level problems (Hendrycks et al., 2021). Following Zhao et al. (2025a), we use its 7,500-problem training split and aligned machine translations from English into ten additional languages to estimate language subspaces. MGSM provides 250 grade-school problems professionally translated across the same eleven languages, with native-language worked examples for generation (Shi et al., 2023; Cobbe et al., 2021). We use it to test the subspaces on a different corpus and to measure their behavioral effects. Thus MATH supplies scale for finding directions, while MGSM asks whether those directions transfer and matter during generation.

We use four open-weight models of similar scale to carry out our study: Qwen2.5-7B (Qwen Team, 2024), Qwen3-8B (Qwen Team, 2025), R1-Distill-7B (DeepSeek-AI, 2025), distilled from DeepSeek-R1 (Guo et al., 2025), and GLM-Z1-9B (Z.ai, 2025).

3.2 Language subspaces

The same problem expressed in two languages lets us ask a clean geometric question: how does its representation move when the linguistic realization changes? Let $h^{\ell}_{i,k} \in \mathbb{R}^d$ be the residual-stream activation at the final prompt token for problem i, language ℓ, and layer k. We fit a rank-r basis U independently at each layer and omit the layer index from the notation below.

Translation differences. For two renderings of the same problem, define

$$v_i^{s \to t} = h_i^t - h_i^s. \tag{1}$$

These vectors hold problem identity fixed. They are therefore a direct test of whether a proposed subspace follows translation. Because translation also changes tokenization, wording, and possibly the representation of the mathematics, we call this *language-associated variation*, not pure language.

Mean-based subspace (MEAN-SVD). MEAN-SVD compresses each language to one mean activation, then factorizes the resulting matrix into a shared anchor and leading directions that distinguish the language means (Piratla et al., 2020; Xie et al., 2022; Zhao et al., 2025a). Averaging suppresses individual problems and emphasizes stable offsets between languages. With eleven language means, its language-varying component has rank at most ten.

Within-problem subspace (DIFFPCA). DIFFPCA instead keeps problem identity explicit. For each problem, it subtracts the mean of its translations and normalizes the remaining language-specific residuals:

$$R_i^\ell = h_i^\ell - \frac{1}{L}\sum_{t=1}^{L} h_i^t, \qquad z_i^\ell = \frac{R_i^\ell}{\max(\|R_i^\ell\|, \epsilon)}. \tag{2}$$

Its basis is the top r principal components of the centered z_i^ℓ. This favors directions repeatedly traversed by translations rather than differences between languages only in the aggregate, and it permits ranks above ten. The two estimators therefore encode different ideas: MEAN-SVD treats language as a stable population offset, whereas DIFFPCA treats it as a recurring within-problem movement. Agreement across them suggests that the observed translation-associated geometry is not an artifact of either construction.

Controls. We compare both with high-variance directions learned without language labels (ACTIVATION-PCA) and a random orthonormal span (RANDOM). ACTIVATION-PCA asks whether the apparent language structure is specific, or merely a consequence of projecting onto the model's dominant directions. Full algorithms and fitting details appear in Appendix A.

4 How much translation variation is captured?

For a translation vector v defined in Equation 1, we measure the fraction of its squared norm captured by a subspace with orthonormal basis U:

$$\rho_U(v) = \frac{\|U^\top v\|^2}{\|v\|^2}. \tag{3}$$

We average over evaluation problems and all 55 unordered language pairs. Appendix B gives the data partitions and full results.

Figure 1 shows a decline in rank-10 capture through intermediate layers followed by an increase toward the output. Both estimators exhibit this pattern across all four models on held-out MATH and cross-corpus MGSM. Their interior minima coincide within each model–corpus pair, at 51–67% of model depth. On MGSM, MEAN-SVD captures 24.0–44.8% of translation variation at these minima, and DIFFPCA captures 29.6–46.2%. Thus, on this corpus, more than half of the variation remains outside either rank-10 span.

ACTIVATION-PCA provides a stronger comparison than random directions. At the MGSM minima, it captures 26.5% for R1-Distill and 34.6% for GLM, exceeding MEAN-SVD's 24.0% and 32.8%, respectively. DIFFPCA captures more than ACTIVATION-PCA at all four models' MGSM minima, but substantial capture is already obtained without language labels or within-problem residualization. Translation differences therefore overlap substantially with high-variance activation directions; capture alone does not establish language specificity.

Increasing DIFFPCA's rank extends coverage beyond the rank-ten limit of MEAN-SVD. At Qwen2.5 layer 18 on held-out MATH, capture rises from 47.9% at rank ten to 74.0% at rank 100, compared with 2.8% for a rank-100 random span. We next test whether language remains linearly decodable after removing these spans.

5 Language decodability after projection

We test whether language identity remains linearly decodable after removing the estimated subspaces. Using subspaces fitted on MATH, we ablate zero-shot MGSM activations at each layer:

$$h' = h - \lambda U U^\top h, \tag{4}$$

where U is an orthonormal subspace basis. At $\lambda = 1$, the component within the subspace is re-

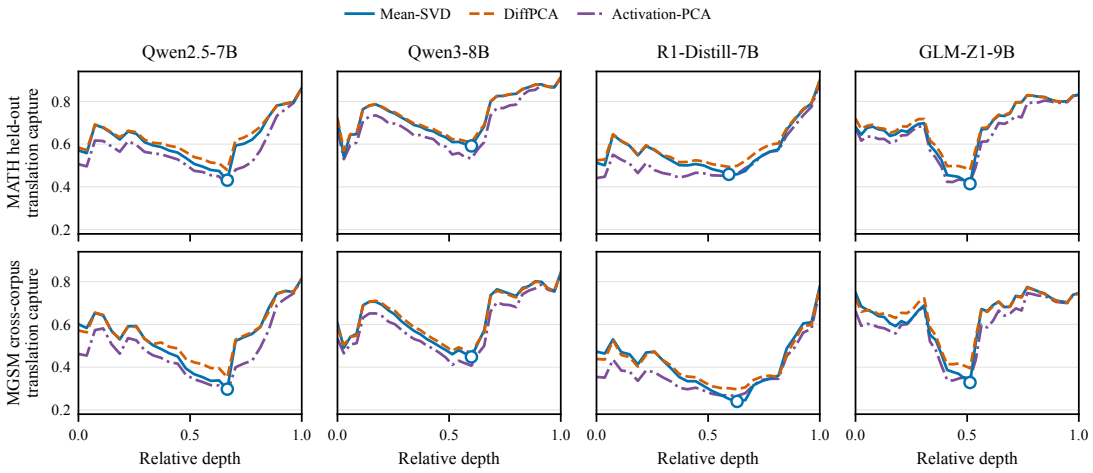


Figure 1: Rank-10 capture of translation differences over model depth. Columns are models and rows are held-out MATH and cross-corpus MGSM. Circles mark the MEAN-SVD interior minima.

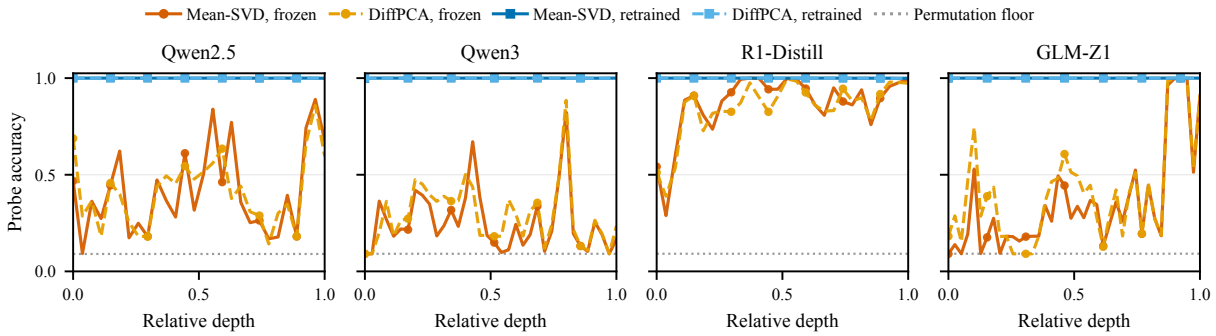


Figure 2: The retrained probe's accuracy curves remain near the ceiling even where the frozen probe's accuracy curves approach the permutation floor (gray). Although the gap varies irregularly with layer depth, it remains pronounced for all models except R1-Distill, whose frozen probe is comparatively robust.

moved completely. For $0 \leq \lambda < 1$, the transformation is invertible and preserves linear separability in principle. We evaluate each layer independently, without propagating edited activations through subsequent blocks.

We train eleven-way logistic-regression classifiers using five-fold cross-validation, keeping all translations of each problem in the same fold. A *frozen probe* trains on unedited activations and evaluates on ablated activations. A *retrained probe* trains and evaluates on ablated activations, using the same partitions. The former measures sensitivity of the original classifier to ablation; the latter tests whether language can still be decoded after ablation (Elazar et al., 2021). Appendix B.3 gives the fitting details.

Figure 2 shows that the accuracy of the retrained probe remains above 99.7% across the tested depth, rank, and strength settings, while the accuracy of the frozen probe varies substantially. After ablation of the language components, successful classification uses the component outside the removed span. Frozen-probe sensitivity also differs across models: at the designated MATH capture minima, rank-10 MEAN-SVD projection reduces accuracy to 24.3–35.7% for Qwen2.5, Qwen3, and GLM, whereas R1-Distill retains 94.6% (Table 5).

ACTIVATION-PCA also reduces frozen accuracy substantially for the first three models, while the rank-matched random control retains 100% accuracy at the designated layers. Disrupting a frozen classifier therefore does not distinguish the language-based subspaces from directions fitted without language labels. Across depth, frozen MEAN-SVD accuracy is lowest at layer 0 or 1, rather than at the intermediate-layer capture minima.

Language identity thus remains linearly decodable after full ablation of either estimated subspace. These local measurements do not establish whether the surviving information contributes to generation. We examine generation outcomes next.

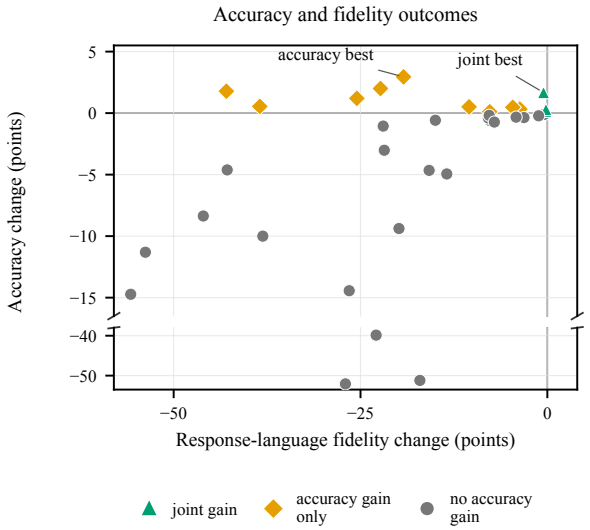


Figure 3: Accuracy gains usually lie far to the left: the intervention appears to reason better precisely when it abandons the requested language. Only a narrow corner improves the joint objective.

Setting	Accuracy (%) ↑	Fidelity (%) ↑	Joint (%) ↑
No intervention	62.80	99.82	62.62
MEAN-SVD, rank 10 projection, dose 0.5	65.75	80.58	53.82
MEAN-SVD, rank 1 projection, dose 1	64.47	99.31	64.22

Table 1: No intervention and the MEAN-SVD configurations with the highest accuracy and joint accuracy on Qwen2.5 across eleven languages.

6 Influencing generation through language subspaces

We now turn from representation to behavior: what changes when these subspaces are edited during generation? We map the intervention design space, reassess earlier results under a stricter task criterion, and compare configurations by answer accuracy, response-language fidelity, and joint success. Appendix C gives the complete intervention and generation protocol.

6.1 Defining the intervention design space

We construct the intervention in stages. First, we estimate a language-specific subspace using either MEAN-SVD or DIFFPCA. For each construction, we vary the rank to test how large the subspace must be: a low-rank edit targets a compact language-related direction, whereas a higher-rank edit removes a broader span of activation variation.

Given a subspace, we then vary how it is edited during generation. Qwen2.5 has 28 transformer layers, numbered 0–27. Following Zhao et al. (2025a), layers 10–19 form the middle band and layers 20–27 the upper band. This split treats mid-network language variation as interference to reasoning, while reserving the upper layers for expressing the result in a particular language. At each edited layer, we apply

$$h' = h - \lambda\,\delta(h), \tag{5}$$

where δ is the intervention operator and λ its signed dose. The reference schedule uses $\lambda = 0.3$ in the middle band and $\lambda = -0.3$ in the upper band, so the selected language component is diminished in the middle layers and amplified in the upper layers. Orthogonal projection sets $\delta(h) = UU^\top h$; LEACE motivates covariance-aware oblique removal with minimal distortion under its guardedness objective (Belrose et al., 2023), while our WIENER-style operator shrinks directions according to their estimated language signal-to-noise ratio. We vary the operator and dose, calibrating dose on MATH activations without observing MGSM outcomes. This design lets us ask whether an effect is robust to how the subspace is defined, how much of it is retained, and how aggressively it is edited, rather than being tied to one particular intervention recipe. Appendix A.2 defines the operators and complete grid.

6.2 Interpreting prior intervention results

On Qwen2.5, Zhao et al. (2025a) report gains in both mean MGSM accuracy and response-language fidelity using language-wise coefficient tuning under a composite objective (Appendix C). We instead evaluate fixed configuration procedures by whether correctness and target-language output improve on the same responses; because the protocols differ, our fidelity losses do not reinterpret their published gains.

6.3 Accuracy gains can mask language substitution

Figure 3 reveals that accuracy gains concentrate where response-language fidelity falls. The accuracy-maximizing configuration exemplifies this trade-off (Table 1). Accuracy alone therefore cannot tell whether an intervention makes the model more capable or merely changes the language in which that capability is expressed. To diagnose the difference, we define *joint accuracy*: a response succeeds only when its answer is correct and it remains in the original problem language.

	Causal intervention*		English switch		Chinese switch	
Model	Accuracy	Fidelity	Accuracy	Fidelity	Accuracy	Fidelity
Qwen2.5	+0.94	+4.69	+1.78	−64.71	+3.42	−95.45
Qwen3 (thinking)	+1.27	+0.03	+1.87	−80.44	−1.73	−77.33
Qwen3 (no thinking)	–	–	+1.42	−72.36	+0.36	−50.89
R1-Distill	+2.40	+1.20	+1.07	−22.49	+2.58	−31.09
GLM-Z1	+1.52	+0.08	+6.53	−76.76	−0.27	−72.76

Table 2: Accuracy and fidelity changes under causal intervention and prompt-language switches. *Causal-intervention values are imported from Zhao et al. (2025a); switch values average the nine non-English/Chinese MGSM languages. Comparable accuracy gains thus conceal radically different language behavior.

Viewed through this joint criterion, the accuracy-maximizing intervention loses its apparent practical advantage: its gain in numerical correctness is accompanied by enough language drift that it produces fewer answers satisfying the task as posed. The intervention has not simply improved behavior; it has traded one part of the task contract against another.

The reversal is most striking for the simplest configuration. A rank-one MEAN-SVD subspace with ordinary orthogonal projection yields a modest joint gain, whereas broader or more elaborate edits do not. This cautions against equating stronger erasure with a cleaner causal intervention. A narrow edit may perturb a behaviorally relevant direction while preserving the distributed features that route generation into the requested language; a broader removal may cut across features jointly involved in reasoning and language control. The grid does not by itself identify that mechanism, but it points to *selectivity*, rather than intervention strength, as the quantity that matters. Appendix Figure 5 shows the joint criterion across all 37 configurations.

7 Prompt language as a behavioral control

Language-subspace ablation is motivated by a mechanistic hypothesis: language-specific variation in middle-layer activations may behave as noise relative to a shared reasoning computation. If that shared computation is disproportionately shaped by high-resource languages, removing the source-language component could move lower-resource activations toward a regime the model uses more effectively in English or Chinese. The same movement, however, has an obvious failure mode: it may pull the language of the output toward that high-resource anchor as well.

We test this possibility without editing activations. In a *prompt-language switch*, the math problem remains in its original language, but its worked examples and answer cue are replaced with English or Chinese versions. This asks whether placing the model in a high-resource linguistic context is sufficient to reproduce both sides of the intervention: greater access to reasoning behavior and drift away from the requested output language. Appendix D illustrates the exact prompt transformation and gives the generation protocol.

Table 2 places these prompt-language switches beside the gains reported for the best causal interventions of Zhao et al. (2025a). The comparison is diagnostic rather than a leaderboard because the generation protocols differ. Its significance is that an accuracy gain of comparable scale can be produced by contextual language routing alone. Yet the resulting answers follow the switched prompt language, so accuracy rises while joint accuracy falls. A better numerical score therefore does not, by itself, identify a better reasoning mechanism; it may instead reveal that the model has entered a linguistic regime in which it is easier to answer but less able to satisfy the original task.

This raises a simple question: what if we explicitly ask the model to answer in the target language? We test this on GLM-Z1 while retaining the English prompt-language switch. Relative to the native-prompt reference, whose accuracy and fidelity are 70.76% and 94.53%, the explicit instruction yields 74.62% accuracy and 97.02% fidelity. The result suggests that the linguistic context used to elicit computation and the language used to realize the answer can be partially decoupled at inference time. We do not optimize the instruction further, since that would turn the analysis into prompt engineering; here, it serves as evidence that language substitution is not inevitable.

8 Conclusion

Estimated language subspaces are easy to overinterpret. Subspaces built from multilingual means or paired translation differences recover translation-associated structure, yet projection leaves language linearly decodable, and geometric salience does not predict behavioral leverage. A subspace can therefore be a useful coordinate system without being a container for the feature, or the mechanism controlling it.

The behavioral results expose the consequence. The edit that appears to improve reasoning most also abandons the requested language; prompt switches reproduce this pattern, and an explicit instruction can reverse it. Apparent gains may thus reflect entry into an easier linguistic regime rather than improved language-independent computation. Multilingual interventions must be judged against the whole task contract. The goal is not to make models reason in one privileged language, but to separate access to useful computation from the language in which users receive it.

Limitations

Our conclusions are bounded by task and model scope. MATH and MGSM are both mathematical reasoning benchmarks, and all four models are similarly scaled at 7–9 billion parameters. Whether the findings generalize across domains or to larger models remains open.

Additionally, the rank-one intervention was selected and evaluated on the same benchmark. Its apparent joint advantage therefore requires held-out confirmation.

Ethical considerations

Our analysis of Zhao et al.'s coefficient-selection procedure is based on their released analyzer. Because the release does not identify the exact code revision used for the paper, we treat this as a finding about the public implementation—not a claim about how the published numbers were produced.

Declaration on Generative AI

Generative AI systems, including Claude and OpenAI Codex, assisted with manuscript editing and verification. They did not conduct the experiments; the authors remain responsible for the work.

References

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. 2023. LEACE: Perfect linear concept erasure in closed form. In *Advances in Neural Information Processing Systems*, volume 36, pages 66044–66063.

Tyler Chang, Zhuowen Tu, and Benjamin Bergen. 2022. The geometry of multilingual language model representations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 119–136.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

DeepSeek-AI. 2025. DeepSeek-R1-Distill-Qwen-7B. Model card.

Alicja Dobrzeniecka, Antske Fokkens, and Pia Sommerauer. 2025. Improving causal interventions in amnesic probing with mean projection or LEACE. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12981–12993.

Clément Dumas, Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2025. Separating tongue from thought: Activation patching reveals language-agnostic concept representations in transformers. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31822–31841.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, et al. 2025. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743.

Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. 2020. Efficient domain generalization via common-specific low-rank decomposition. In *Proceedings*

of the 37th International Conference on Machine Learning, volume 119, pages 7728–7738. PMLR.

Qwen Team. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In *International Conference on Learning Representations*.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Steering language models with activation engineering. *Preprint*, arXiv:2308.10248.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394.

Zhihui Xie, Handong Zhao, Tong Yu, and Shuai Li. 2022. Discovering low-rank subspaces for language-agnostic multilingual representations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5617–5633.

Z.ai. 2025. GLM-Z1-9B-0414. Model card.

Weixiang Zhao, Jiahe Guo, Yang Deng, Tongtong Wu, Wenxuan Zhang, Yulin Hu, Xingyu Sui, Yanyan Zhao, Wanxiang Che, Bing Qin, Tat-Seng Chua, and Ting Liu. 2025a. When less language is more: Language–reasoning disentanglement makes LLMs better multilingual reasoners. In *Advances in Neural Information Processing Systems*, volume 38, pages 43235–43269.

Weixiang Zhao et al. 2025b. Language–reasoning disentanglement: Released grid-search analyzer. GitHub repository. Commit 059b57d8895a7ce5b525431b0abbff620ebd33f1.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? In *Advances in Neural Information Processing Systems*, volume 37, pages 15296–15319.

A Subspace and operator details

A.1 Subspace construction

As Table 3 makes explicit, the MEAN-SVD output is the orthonormal language basis $\widetilde{U}_r$, not the shared anchor a. Our implementation matches the released implementation we inspected for Zhao et al. (2025a). DIFFPCA is an angular construction because each problem–language residual has equal norm before centering; this prevents a few unusually large activations from dominating the fit. ACTIVATION-PCA applies the same normalized PCA to all activation rows without residualizing problem identity or observing language labels. RANDOM is obtained by QR decomposition of a Gaussian matrix with a fixed seed. All bases are fitted independently at each layer.

A.2 Edit operators

All generation edits have the affine form $h' = h - \lambda\delta(h)$. Orthogonal projection uses $\delta(h) = UU^\top h$. Our LEACE-style operator estimates the pooled covariance Σ, shrinks it by 0.1 toward a scaled identity, and orthonormalizes the target span after whitening. Writing μ for the pooled mean and Q for the resulting orthonormal basis, it applies

$$\delta(h) = \Sigma^{1/2} QQ^\top \Sigma^{-1/2}(h - \mu). \tag{6}$$

With the centered language-mean span and full rank, this is the minimal-MSE linear guardedness map of Belrose et al. (2023). With a supplied DIFFPCA basis, it is an oblique removal of that span and does not inherit the supervised guardedness guarantee.

The WIENER operator uses the random-effects decomposition $h = \mu + c_i + g_\ell + q_{i\ell} + \epsilon$, where c_i, g_ℓ, $q_{i\ell}$, and ϵ denote problem, language, problem–language interaction, and residual effects. For the mean-based target, its edit is

$$\delta(h) = \Sigma_g(\Sigma_g + \Sigma_c + \Sigma_\epsilon)^{-1}(h - \mu), \tag{7}$$

with the covariance terms estimated from the parallel MATH activations. This attenuates directions according to their estimated signal-to-noise ratio instead of assigning every direction the same coefficient. For the DIFFPCA cross, the basis directions are scaled by their residual-covariance energy before the same marginal-covariance shrinkage. Mean-based WIENER is intrinsically rank at most $L - 1$ and has no separate rank knob. Consequently the cross is deliberately unbalanced: the 37 cells comprise 6 mean-projection, 6 mean-oblique, 4 mean-Wiener, and 7 cells for each DIFFPCA operator. It should not be read as a complete factorial comparison of estimator and operator.

B Representation analyses: protocols and supplementary results

B.1 Activation extraction and data partitions

Representation analyses use zero-shot residual-stream activations at the final prompt token; the eight demonstrations used in generation are absent. Layers are indexed from zero over transformer block outputs, with relative depth $k/(K-1)$ for a model with K blocks. Qwen3 uses non-thinking mode. GLM's chat template appends an opening thinking prefix, so its final prompt token follows that prefix. Section B.5 examines this choice.

For held-out MATH capture, we split the 7,500 problems into 3,750 fitting and 3,750 evaluation problems using seed zero. All translations of each problem stay in the same partition. For cross-corpus capture, we fit on all 7,500 MATH problems and evaluate on 250 MGSM problems. The fitting sample size and evaluation corpus both change between these tests, so their absolute capture levels do not isolate a corpus effect.

The probe experiment uses a third partitioning scheme: subspaces are fitted on all 7,500 MATH problems, but classifiers are trained and evaluated within MGSM using grouped cross-validation. Thus the subspace transfers across corpora; the classifier itself is not trained on MATH and transferred to MGSM.

B.2 Translation-capture results

Capture is the mean of the per-vector squared-norm fractions in Equation 3, with equal weight for every evaluation problem and each of the 55 unordered language pairs. It is not a ratio of pooled squared norms. Table 4 reports both estimators at the layer minimizing rank-10 MEAN-SVD capture within $0.15 \le k/(K-1) \le 0.95$, selected separately for each model and evaluation corpus. Only Qwen3/MATH differs from the unrestricted minimum, which occurs at layer 1. The correlations summarize agreement between full depth profiles, not alignment of the underlying subspace directions.

Rank sensitivity. At Qwen2.5 layer 18 on held-out MATH, ranks 10, 20, and 100 yield DIFFPCA

In: At fixed layer k, activations $h^{\ell}_{i,k}$ for n problems and L languages; rank r ($r \le L-1$ for (a)). Method (b) requires parallel activations.

(a) MEAN-SVD:
Common-specific decomposition

Approximate the language means at rank r.

$m_\ell \leftarrow \frac{1}{n}\sum_i h^{\ell}_{i,k}$; $W \leftarrow [m_1, \dots, m_L]$.

$a_0 \leftarrow \frac{1}{L} W\mathbf{1}$; $U\Sigma V^\top \leftarrow \mathrm{SVD}(W - a_0\mathbf{1}^\top)$.

$\widehat{W} \leftarrow a_0\mathbf{1}^\top + U_r\Sigma_r V_r^\top$.

Refine the shared anchor and recover the language basis.

$a \leftarrow (\widehat{W}^+)^\top\mathbf{1}$; $a \leftarrow \frac{a}{a^\top a}$.

$\widetilde{U}\widetilde{\Sigma}\widetilde{V}^\top \leftarrow \mathrm{SVD}(\widehat{W} - a\mathbf{1}^\top)$.

Return the first r columns of $\widetilde{U}$.

Out: Orthonormal language basis $\widetilde{U}_r \in \mathbb{R}^{d\times r}$.

(b) DIFFPCA:
Within-problem PCA

Remove each problem's mean and normalize its translation residuals.

$\bar{h}_i \leftarrow \frac{1}{L}\sum_\ell h^{\ell}_{i,k}$.

$R^{\ell}_i \leftarrow h^{\ell}_{i,k} - \bar{h}_i$; $z^{\ell}_i \leftarrow \frac{R^{\ell}_i}{\max(\|R^{\ell}_i\|, \epsilon)}$.

Fit centered PCA to the normalized residuals.

Stack the z^{ℓ}_i as rows of Z; $Z_c \leftarrow Z - \mathbf{1}\bar{z}^\top$.

$A\Sigma V^\top \leftarrow \mathrm{SVD}(Z_c)$.

Return the first r columns of V.

Out: Orthonormal translation-residual basis $V_r \in \mathbb{R}^{d\times r}$.

Table 3: Layerwise construction of MEAN-SVD and DIFFPCA subspaces. Subscript r selects the leading r singular components; W^+ is the Moore–Penrose pseudoinverse and $\epsilon > 0$ is a numerical norm floor.

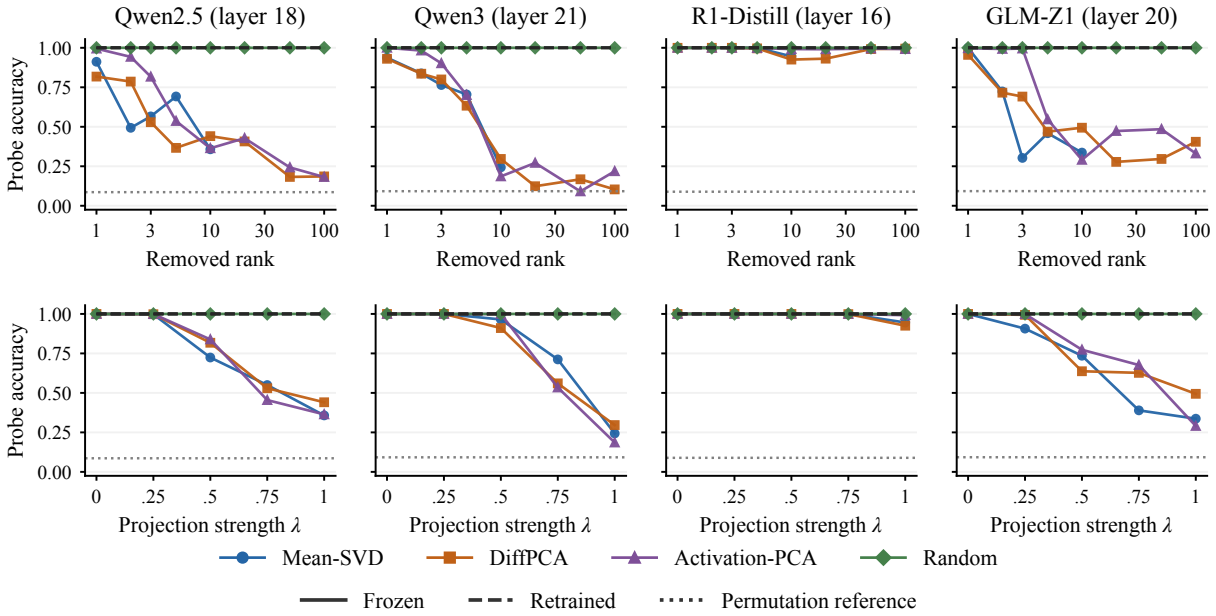


Figure 4: Probe sensitivity at each model's held-out MATH capture minimum. Top: rank varies at full projection. Bottom: strength varies at rank ten. Solid curves are frozen probes; dashed curves are retrained probes and overlap near 1. The dotted gray line is the measured permutation reference at that layer. MEAN-SVD ends at rank ten.

capture of 0.479, 0.565, and 0.740, versus 0.0028, 0.0056, and 0.0276 for random spans.

B.3 Language-probe protocol

We classify eleven languages using logistic regression on unscaled activations. The implementation calls scikit-learn's `LogisticRegression` with $C = 1.0$, `max_iter=3000`, and `random_state=0`; other arguments use library defaults. Five-fold `GroupKFold` partitions the 250 MGSM problem indices. Each fold trains on 200 problems (2,200 activations) and evaluates on 50 (550 activations), keeping all eleven translations together. Accuracy is averaged over the five held-out folds.

The clean probe trains and evaluates on unedited activations. Within the same folds, the frozen probe trains on clean training activations and evaluates on projected held-out activations; the retrained probe trains and evaluates on projected activations. Every edit is applied analytically at the measured layer using Equation 4. No edited state is propagated through subsequent blocks. Subspace fitting never

Model/data	MEAN-SVD	DIFFPCA	k/depth	r_P
Qwen2.5/MATH	0.431	0.479	18/0.67	0.9971
Qwen2.5/MGSM	0.297	0.352	18/0.67	0.9918
Qwen3/MATH	0.592	0.608	21/0.60	0.9995
Qwen3/MGSM	0.448	0.462	21/0.60	0.9972
R1/MATH	0.459	0.494	16/0.59	0.9940
R1/MGSM	0.240	0.296	17/0.63	0.9882
GLM-Z1/MATH	0.415	0.479	20/0.51	0.9976
GLM-Z1/MGSM	0.328	0.395	20/0.51	0.9911

Table 4: Rank-10 translation capture at each model–corpus pair's MEAN-SVD interior minimum. k is the block index; depth is $k/(K-1)$. r_P is the Pearson correlation between the two estimators' full depth profiles.

uses the held-out probe examples.

The three sweeps vary one factor at a time:

- **Depth:** every block, with rank ten and $\lambda = 1$, for MEAN-SVD and DIFFPCA.
- **Rank:** $r \in \{1, 2, 3, 5, 10, 20, 50, 100\}$ at the designated layer and $\lambda = 1$, for both estimators and the ACTIVATION-PCA and RANDOM controls. MEAN-SVD is restricted to $r \leq 10$.
- **Strength:** $\lambda \in \{0, 0.25, 0.5, 0.75, 1\}$ at the designated layer and rank ten, for both estimators and both controls.

The designated layers are the rank-10 MEAN-SVD interior minima on held-out MATH: 18, 21, 16, and 20 for Qwen2.5, Qwen3, R1-Distill, and GLM, respectively. In particular, R1-Distill's probe sweeps use layer 16, even though its MGSM capture minimum is at layer 17. The sweeps do not cover the full depth–rank–strength product.

For the permutation reference, we globally shuffle the language-label vector five times using seed zero and repeat grouped cross-validation on clean activations. We measure this reference at the first, designated, and final layers. In Figure 2, unmeasured layers use the mean of these three measured values; the gray reference is therefore not independently measured at every layer. The references in Table 5 and Figure 4 are measured at the designated layer. They provide an empirical chance reference, not a statistical lower bound on accuracy.

B.4 Probe results and controls

Table 5 reports the model-level results. The rank sweeps in Figure 4 show that frozen accuracy need not fall monotonically with the number of removed dimensions: R1-Distill's frozen DIFFPCA probe has accuracy 0.925 at rank ten and 0.995 at rank 100. Across all tested intervention cells, retrained accuracy remains at least 0.9971, and the random control retains 1.000 accuracy in every tested rank and strength condition.

B.5 GLM extraction-boundary sensitivity

GLM's forced-thinking template ends with an opening `<think>` prefix. On 64 MGSM problems per language, we compare block outputs at the token immediately before this prefix with those at the final token after it, within the same forward pass. For representations a and b at these positions, the diagnostics are cosine similarity and relative change $\|b - a\|/\|a\|$. We first average over problems within each language and layer, then equally over all eleven languages and 40 layers.

The mean cosine similarity is 0.645 and the mean relative change is 0.865. Per-language cosine similarities, averaged over layers, range from 0.634 to 0.661. Capture and probe analyses use the post-prefix position; their robustness across the two extraction positions was not tested.

C Intervention selection and evaluation

The reference schedule uses the middle/high layer bands of Zhao et al. (2025a) and one fixed point within their coefficient search grid: $+0.3$ at layers 10–19 and -0.3 at layers 20–27. Their Appendix F specifies the coefficient grid for Qwen2.5-7B. The released analyzer, pinned to the inspected revision in Zhao et al. (2025b), takes the no-intervention cell as the baseline and, among intervention cells, selects a separate pair for each language by maximizing

$$2(A - A_0) + (F - F_0), \tag{8}$$

where all scores are proportions and A_0 and F_0 are the no-intervention scores. Selection excludes the baseline and is performed independently for each language. Thus "reference" in our grid means our single fixed $+0.3/-0.3$ schedule applied to every language, not a coefficient pair reported as universal by Zhao et al. Off-center schedules are calibrated using activation-edit energy on MATH, without observing MGSM outcomes.

For each construction–operator pair, dose 1 is the reference energy. The other dose rungs target 0.5, 1.5, and 2 times that energy by changing the coefficient magnitude for each language. Both layer bands are scaled together. Rank rungs are evaluated at the central coefficients; dose rungs use rank ten.

Model	k	MEAN-SVD	DIFFPCA	ACTIVATION-PCA	RANDOM	Perm.	Min. retr.
Qwen2.5	18	0.357 / 1.000	0.441 / 1.000	0.364 / 1.000	1.000 / 1.000	0.086	0.9993
Qwen3	21	0.243 / 1.000	0.296 / 1.000	0.186 / 1.000	1.000 / 1.000	0.092	0.9971
R1-Distill	16	0.946 / 1.000	0.925 / 1.000	0.992 / 1.000	1.000 / 1.000	0.089	0.9985
GLM-Z1	20	0.337 / 1.000	0.494 / 1.000	0.291 / 1.000	1.000 / 1.000	0.093	0.9993

Table 5: Language probes at the held-out MATH capture minimum, using rank ten and full projection ($\lambda = 1$). Each method cell gives frozen / retrained held-out accuracy. Clean accuracy is 1.000 for every model. The permutation reference uses shuffled labels on clean activations. The final column is the lowest retrained accuracy over all tested intervention cells across the depth, rank, and strength sweeps, reported to four decimals.

DIFFPCA additionally admits rank 32, whereas MEAN-SVD is capped at ten. All intervention generations use a 512-token limit. Edits operate on every prompt token during prefill and are removed before autoregressive decoding begins.

C.1 Scoring and uncertainty

Answers are extracted after any completed visible reasoning block. Native decimal digits are mapped to ASCII, and the last localized answer-field number is compared numerically with the gold answer. To classify response language, we first strip mathematics. Kana identifies Japanese, while dominant Han, Cyrillic, Bengali, Telugu, or Thai script identifies the corresponding language. Remaining Latin prose must contain at least eight alphabetic characters; we then take the highest-ranked `langdetect` candidate among German, English, Spanish, French, and Swahili, with `DetectorFactory.seed=0`. A missing or unidentifiable label receives $F_i = 0$ in the strict metric.

Each pooled statistic first averages problem indices within a language and then averages languages equally. For intervals and tests, 4,000 paired bootstrap replicates resample the 250 matched problem indices; every draw carries together all languages and all conditions being contrasted for an index. Reported 95% intervals are pointwise percentile intervals, not simultaneous intervals over the grid. Approximate two-sided bootstrap p-values compare each intervention with baseline using twice the smaller proportion of nonpositive or nonnegative bootstrap differences, capped at one. For the reported all-language comparisons, Holm adjustment uses one family of 37 intervention arms for each of accuracy, fidelity, and joint accuracy. We flag a cell when its truncation rate differs from baseline by more than 0.05; this is a sensitivity gate, not a correction for high truncation shared with the baseline.

C.2 Grid-wide outcomes and truncation sensitivity

Across the 407 configuration–language cells, 152 have positive accuracy changes and 110 have positive joint changes. After averaging all eleven languages, 12 of 37 configurations exceed baseline accuracy and three exceed baseline joint accuracy. One comparison for each metric survives Holm correction. Selection and evaluation use the same benchmark; no independent confirmation set is used.

Baseline Telugu and Bengali responses reach the 512-token limit in 78.8% and 36.8% of cases, respectively. Of the 407 cells, 112 change truncation by more than five percentage points.

D Prompt-language experiments

D.1 Prompt conditions and evaluation set

All conditions use eight worked examples. The native condition presents the examples, question, and answer cue in the original problem language. A switch replaces the examples and cue with English or Chinese while leaving the question in its original language. Translate-test also translates the question into the switch language, using the benchmark's existing translations. In the cue control, the examples remain in the switch language but the cue is restored to the native language. The target-instruction control retains the switched examples and cue and prepends an explicit request for the final answer and explanation in the native language.

Table 6 uses the same nine languages as the prompt-switch columns of main-text Table 2: Bengali, German, Spanish, French, Japanese, Russian, Swahili, Telugu, and Thai. Each condition contains 250 problems per language. Excluding both English and Chinese fixes the evaluation set across the two switches. Scores average languages equally; each change is relative to that model and mode's native-prompt baseline on these same problems.

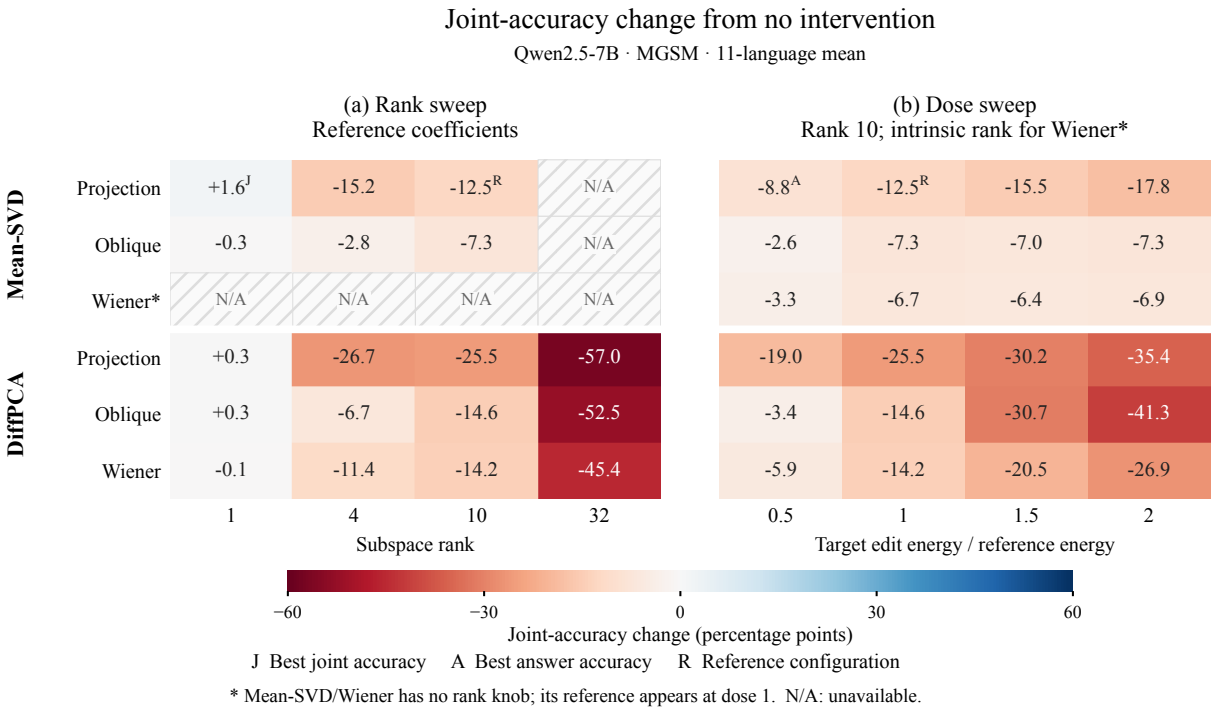


Figure 5: Joint-accuracy changes across 37 intervention configurations. Both panels use the same symmetric color scale. Five reference settings appear in both sweeps; Mean-SVD/Wiener's intrinsic-rank reference appears only at dose 1. Letters identify the best joint accuracy (J), best answer accuracy (A), and reference projection (R).

	Prompt-language switch			Translate-test		
Model	Accuracy	Fidelity	Joint	Accuracy	Fidelity	Joint
English						
Qwen2.5	+1.78	−64.71	−40.49	+27.02	−99.82	−61.33
Qwen3 (thinking)	+1.87	−80.44	−65.24	+14.09	−99.60	−82.04
Qwen3 (non-thinking)	+1.42	−72.36	−53.29	+20.04	−99.96	−74.89
R1-Distill	+1.07	−22.49	−6.71	+32.58	−97.85	−48.71
GLM-Z1	+6.53	−76.76	−55.73	+25.91	−94.49	−70.40
Chinese						
Qwen2.5	+3.42	−95.45	−58.22	+19.51	−99.82	−61.33
Qwen3 (thinking)	−1.73	−77.33	−63.16	+6.36	−99.60	−82.04
Qwen3 (non-thinking)	+0.36	−50.89	−37.87	+10.84	−99.96	−74.89
R1-Distill	+2.58	−31.09	−9.47	+28.53	−97.85	−48.71
GLM-Z1	−0.27	−72.76	−52.76	+17.51	−94.53	−70.44

Table 6: Prompt-language switches and translate-test on the same nine MGSM languages. Values are percentage-point changes from the matched native-prompt baseline. Fidelity and joint accuracy are scored against the original problem language, including for translate-test.

D.2 Generation and scoring

Qwen2.5 uses greedy decoding with a 1,024-token limit. Qwen3, R1-Distill, and GLM use sampling with a 16,384-token limit; Qwen3's thinking and non-thinking settings are reported separately. GLM uses vLLM 0.8.5.post1, temperature 0.6, and top-p 0.95. These budgets differ from the 512-token intervention runs, so comparisons use changes from each protocol's own baseline.

Accuracy measures numerical correctness, fidelity measures output in the original problem language, and joint accuracy requires both on the same response. Unidentified responses count as fidelity and joint failures, using the scoring procedure in Appendix C.1.

D.3 Switch and translate-test results

Table 6 extends the main comparison with joint accuracy and both translate-test conditions. Every prompt-language switch reduces joint accuracy, including settings that improve numerical accuracy.

Translate-test increases accuracy for every model and mode in both languages.

D.4 Target-language instruction and answer cue

Prompt	Seed 0	Seed 1
Native prompt	70.44	73.78
English switch	14.71	15.47
English + target instruction	72.49	72.04
Chinese switch	17.69	17.47
Chinese + target instruction	59.69	57.60

Table 7: GLM joint accuracy (%) on the same nine languages, with and without an explicit target-language instruction.

The target-language instruction restores much of the lost joint accuracy in both GLM seeds (Table 7).

Changing only the answer cue also affects language fidelity. With English examples, accuracy changes by -0.44 to $+2.36$ points and fidelity falls by 3.97–48.80 points; with Chinese examples, the corresponding ranges are -0.44 to $+2.18$ and 18.57–48.22 points.

Ground Before Correcting: Evidence-Guided Visual Chinese Grammatical Error Correction at Multi-LLL 2026

Xiaochuan Xie **Keran Huang**
Chongqing University of Posts and Telecommunications
Chongqing, China

Abstract

We describe the GroundCorrect system of team xc-皮波 for the **Sentence-Level Track of VisCGEC-v2.0** at Multi-LLL 2026. The system combines frozen Qwen3-VL-4B-Instruct global recognition, rank-8 LoRA verification of full-line and local-crop evidence, and YOLO11s character detection with deterministic edit grounding. According to the retained team record, the official V3 competition submission ranked **second** with score **0.5114**. This result belongs exclusively to the official submission, not to the local reconstruction or the post-submission V3+ extension. Our primary local adapter comparison holds inputs, prompt, decoding, and grounding fixed: the development proxy rises from 0.1254 to 0.2234. Other configuration changes are reported as system development. V3+ is an exploratory case study that changes one development example, with no internal-selection or grounded-edit improvement. Local results are conditional configuration analyses: selection images also train the detector, and development data overlap training at the essay-group level.

1 Introduction

Team xc-皮波 participated in the **Sentence-Level Track of VisCGEC-v2.0** at the Multi-LLL 2026 Shared Task with GroundCorrect. VisCGEC takes learner handwriting and requires corrected Chinese text with spatially grounded edits (Wang et al., 2025). Unlike text-only Chinese grammatical error correction (CGEC), it couples recognition, correction, and localization: even a correct sentence can have incorrect edit coordinates.

Three difficulties motivate our design. Global resizing can obscure similar characters in long, narrow images. A vision–language model (VLM) can correct text yet generate unreliable coordinates. Further correction can introduce unsupported rewrites or inconsistent edit lists.

Our pipeline (Figure 1) separates *global understanding*, *local visual verification*, and *coordinate assignment*. A frozen pass proposes source and target sentences; a format-matched LoRA inspects retrieved regions with the full line. A separate detector grounds text differences, and guarded finalization handles output risks. An optional post-submission rank check is analyzed separately.

Competition submission and later analyses. The retained team record identifies V3 as the August 29 official submission: second place, score 0.5114. The September 11 local reconstruction is not assumed byte-identical: original weights and the complete August run directory were not preserved (Appendix D). V3+ is the September 12 exploratory extension without an official hidden-test result. The rank and score belong only to the official submission.

Our contributions are threefold:

- An evidence-guided correction pipeline that combines a frozen global pass with format-matched local verification.
- A deterministic grounding mechanism that separates linguistic edits from coordinate generation through character detection and monotonic alignment.
- A controlled local adapter comparison with fixed inputs, prompt, and grounding, accompanied by explicit data-use and evaluation disclosures. System-development records and the exploratory V3+ case study are reported separately.

2 Related Work

Chinese GEC. NLPCC 2018 established a common Chinese GEC setting (Zhao et al., 2018); FCGEC supplies fine-grained operations (Xu et al.,

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 33–46
November 9, 2026 ©2026 Association for Computational Linguistics

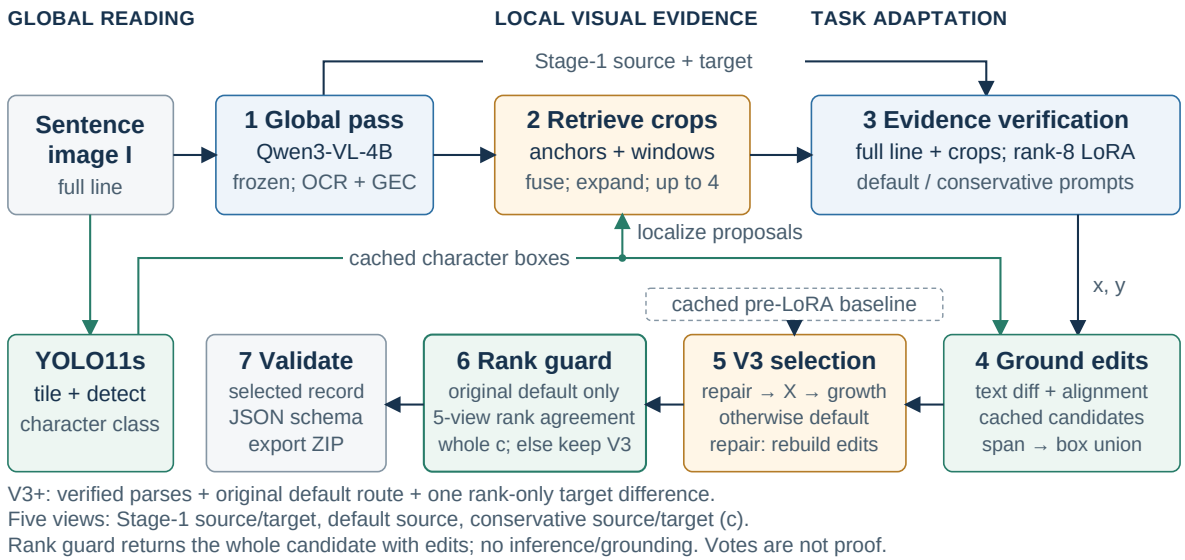


Figure 1: GroundCorrect: frozen global recognition, shared Stage-2 LoRA verification, YOLO11s grounding, and V3 candidate finalization. The rank-check block is optional post-submission V3+ work, not part of the official V3 submission. Candidates undergo schema validation.

2022), and MuCGEC emphasizes multi-reference, character-level evaluation (Zhang et al., 2022). These resources contextualize VisCGEC's image-grounded setting; we do not add their data to training.

Local visual evidence. TrOCR combines pre-trained image and text Transformers for recognition (Li et al., 2023). LLaVA-UHD handles high-resolution, variable-aspect-ratio images through slices (Guo et al., 2025). Our crops instead use task hypotheses and coverage windows outside the VLM. We adapt Qwen3-VL (Bai et al., 2025) with LoRA (Hu et al., 2022) and use a separate YOLO11 character detector (Jocher and Qiu, 2024) for grounding.

Conservative correction and grounding. Align-and-Predict decoding adjusts the precision–recall trade-off in GEC (Sun and Wang, 2022), and Alirector uses alignment to address overcorrection in Chinese GEC (Yang and Quan, 2024). The exploratory V3+ case study selects complete candidates after generation. Slicing-aided detection (Akyon et al., 2022) offers a related motivation for preserving small visual details, although our sentence tiling is independently implemented. TrOCR, LLaVA-UHD, APD, Alirector, and slicing-aided detection are related work, not used components or evaluated baselines.

3 Method

3.1 Task and system overview

Given a sentence image $I \in \mathbb{R}^{H\times W\times 3}$, the system returns corrected text $\hat{y}$ and grounded edits $\hat{E} = \{(o_k, t_k, b_k)\}_{k=1}^{K}$, where o_k, t_k, and b_k are an operation, replacement text, and pixel-space box. Internally, a candidate is $q = (x, y, E)$: retaining the recognized source x makes its connection to the edit list explicit. The output schema allows Insert, Delete, Substitute, and WordOrder; the released sentence annotations contain only the first three types.

Figure 1 summarizes three functional modules. Evidence-guided correction produces a source–target pair from the full line and local crops. Deterministic grounding maps its text differences to character boxes. V3 finalization selects or repairs candidates before schema validation; the optional V3+ rank check is a later extension. Only the Stage-2 adapter and the character detector are trained; the base VLM and its visual encoder remain frozen.

3.2 Evidence-guided correction

Global hypothesis generation. The first pass uses frozen Qwen3-VL-4B-Instruct. A Chinese prompt asks the model to recognize the handwriting and make only high-confidence grammatical changes while preserving meaning. Its JSON output contains the recognized source, provisional target, and provisional edits. Writing G_θ for greedy

generation and p_0 for the global prompt, we retain the text fields

$$(x^{(0)}, y^{(0)}) = \pi_{x,y} G_\theta(I, p_0). \tag{1}$$

The projection $\pi_{x,y}$ parses the JSON fields, rather than taking the VLM's provisional boxes as final edits. This full-line view supplies global syntax and identifies regions for closer inspection.

Contextual region retrieval. We form candidate anchors from two test-time signals: localized differences between the provisional source and target (weight 1.0), and coordinates directly proposed by the VLM (weight 0.55). Proposals merge when their IoU is at least 0.15 or horizontal centers differ by at most 2.5% of image width. Deterministic overlapping coverage windows supplement a short candidate list. We retain at most four regions and expand each to the full image height and at least 22% of its width, preserving neighboring text (Figure 2, left). Reference targets and gold edit boxes are not used for test-time retrieval.

Format-matched verification. Stage 2 receives the full image, up to four crops, a manifest of their global coordinates, and the Stage-1 source and target. The prompt treats retrieved regions as evidence, not guaranteed errors: local views resolve ambiguous characters, while the full line supplies grammatical context.

Let $z = (I, C, x^{(0)}, y^{(0)}, p_2)$ denote this input, including crop set C and the default or conservative prompt p_2. For each adapted linear map, LoRA (Hu et al., 2022) parameterizes

$$W' = W + \frac{\alpha}{r} BA, \qquad r = 8, \quad \alpha = 16, \tag{2}$$

with $A \in \mathbb{R}^{r \times d_{\text{in}}}$ and $B \in \mathbb{R}^{d_{\text{out}} \times r}$. The base W is frozen. This adds $r(d_{\text{in}} + d_{\text{out}})$ parameters instead of updating $d_{\text{in}} d_{\text{out}}$ entries per map. We adapt attention and feed-forward projections, with dropout 0.05; the visual encoder remains frozen.

Training uses the default prompt and the inference input format. The assistant answer is text-only JSON containing `source_text` and `target_text`. Let $a_{1:T}$ be the unmasked assistant suffix, including the chat-template terminator. With trainable adapter parameters ϕ, the token-average objective is

$$\mathcal{L}(\phi) = -\frac{1}{T} \sum_{t=1}^{T} \log p_{\theta,\phi}(a_t \mid z, a_{<t}). \tag{3}$$

This follows the autoregressive factorization of the supervised answer. Prompt labels are masked to -100, so image and prompt tokens condition generation but receive no direct prediction loss. Reference text occurs only in the assistant answer; there is no coordinate loss or joint backpropagation through retrieval and grounding. Stage 2 is accepted only if parsing succeeds without an error flag and both strings are nonempty. Otherwise, the system uses the cached Stage-1 hypothesis and rebuilds its edits. A conservative prompt provides an alternative candidate using the same adapter, not a second trained model.

3.3 Deterministic edit grounding

Linguistic correctness and spatial correctness need not coincide. We therefore discard the VLM's provisional edit semantics and coordinates when constructing grounded candidates. Instead, character-level sequence alignment between source x and target $\hat{y}$ determines INSERT, DELETE, and SUBSTITUTE spans.

A single-class YOLO11s detector supplies m boxes sorted by horizontal center. For $n, m > 0$, let source-character width w_i be 0.45 for whitespace, 0.60 for ASCII alphanumerics, and 1 otherwise. Expected character centers and normalized box centers are

$$u_i = \frac{\sum_{k<i} w_k + w_i/2}{\sum_{k=1}^{n} w_k}, \qquad v_j = \frac{c_j - c_{\min}}{c_{\max} - c_{\min}}, \tag{4}$$

where c_j is a box's horizontal center and $v_j = 0.5$ if all centers coincide. This compares relative positions without requiring a detector class for every character identity.

Let $D_{i,j}$ be the minimum cost of consuming i characters and j boxes. A monotonic alignment ends in exactly one of three transitions: match both, skip a character, or skip a box. Thus,

$$D_{i,j} = \min \begin{cases} D_{i-1,j-1} + 3(u_i - v_j)^2, \\ D_{i-1,j} + 0.10, \\ D_{i,j-1} + 0.10. \end{cases} \tag{5}$$

Initialize $D_{0,0} = 0$, $D_{i,0} = 0.10i$, and $D_{0,j} = 0.10j$. DP uses $O(nm)$ time and space; backtracking takes $O(n + m)$. Without detections, characters map to null. Optimality concerns the geometric cost, not recognition (Appendix A.5).

A non-insertion edit receives the bounding union of available boxes in its source span. An insertion uses its source index, capped at the final

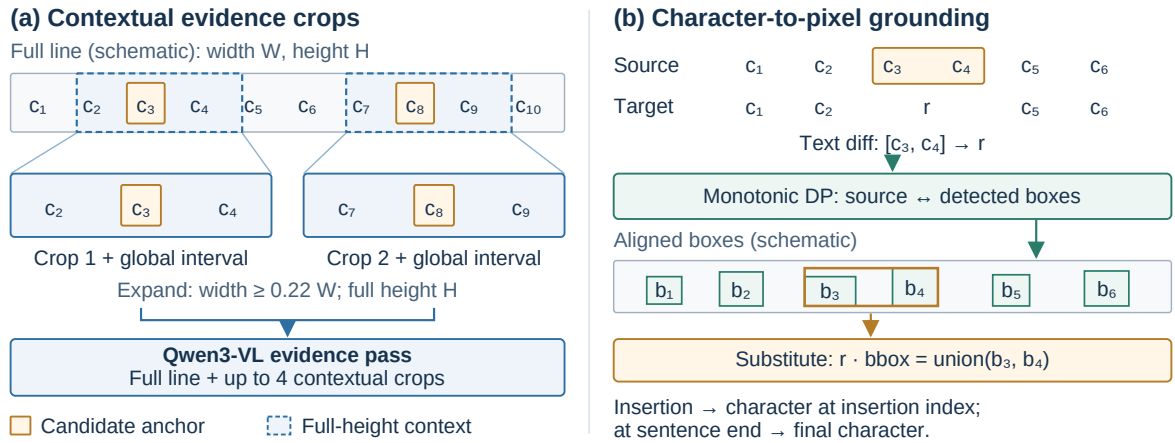


Figure 2: Evidence retrieval and edit grounding (schematic). Left: narrow anchors expand into full-height contextual crops. Right: a text difference identifies a source span, and monotonic character–box alignment maps it to a box union. Symbols illustrate the mechanism, not benchmark examples.

character. If no required aligned box exists, the implementation uses a single uniform-width source cell over the full image height. Partially matched spans use their available boxes without filling every unmatched character. Text differences use deterministic sequence matching, not an optimization of the downstream correction score. Figure 2 illustrates this separation of text differencing from coordinate assignment.

3.4 Priority-based V3 finalization

The finalizer applies rules in a fixed, first-match order. It first checks two narrow consistency repairs: restoring a Stage-1 comma at a preserved *if*-clause boundary, and completing an asymmetric comparison from the activity stated on its other side. A successful repair returns immediately with edits rebuilt from the default source and cached detector boxes.

Otherwise, it checks for an uppercase `X` newly introduced relative to the Stage-1 target, followed by target–source growth of at least three characters. A flagged output selects the conservative candidate if that candidate removes the specific trigger; otherwise it selects the retained baseline. Unflagged outputs retain the default candidate. These selection branches preserve their existing grounded edit lists. Length growth is a risk signal, not proof of hallucination.

We call the default matched-LoRA configuration V1, the risk guard V2, and the finalizer with both repairs V3. The optional V3+ post-check leaves earlier repair and fallback routes unchanged.

3.5 Exploratory post-submission rank check

V3+ was developed after submission to examine a known same-length rank substitution that the length and placeholder checks missed. It compares rank expressions across existing predictions and, under narrow parsing, agreement, and single-change conditions, selects the complete conservative candidate only on V3's unchanged default route. This keeps each selected sentence together with its own source and grounded edits. The observed evidence is one changed development example, with unchanged internal-selection and grounded-edit scores; it supports a targeted case study rather than a general factual-preservation claim. Section 4.5 reports its scope, and Appendix B contains the rule-level definitions and equations.

4 Experiments

4.1 Experimental setup

Data and preparation. We use the released Vis-CGEC sentence-level train, dev, and test splits (Wang et al., 2025), summarized in Table 1. Train/dev provide source transcriptions, target text, edit types, and boxes; the shared-task test input provides identifiers and images, not labels. Official train/dev character annotations are converted to a single detection class. Tiling at a 12:1 aspect ratio with 20% horizontal overlap produces 4,431 training tiles with 63,545 box instances and 573 dev tiles with 8,269 instances. No public test targets or hidden labels are used for training or scoring.

(a) Released annotations

Split	Images	Edits	Ins.	Del.	Sub.
Train	1,960	3,533	1,079	761	1,693
Dev	245	435	134	89	212
Test	246	–	–	–	–

(b) Component and selection uses

Subset	Images	LoRA	Detector	Other use
LoRA fit	1,746	Fit	Fit	Training
Selection	214	–	Fit	Select/report
Dev	245	–	Validate	Select/report
Test	246	–	Infer	Official eval.

Table 1: Released data and component-specific uses. The two train subsets sum to 1,960; both fit the detector. Selection excludes LoRA fitting only; dev validates the detector. Selection/dev also guide configurations and local reporting. Test labels are unavailable. Neither local split is an independent end-to-end generalization test.

Training and inference. The local reconstruction uses one NVIDIA GeForce RTX 5090 (32GB). LoRA trains 16,515,072 parameters (0.371%) for one epoch on 1,746 examples from 499 essay groups, reserving 214 examples from 55 groups. It uses learning rate 5×10^{-5}, batch 1, accumulation 8, cosine decay, bfloat16, and seed 20260829. Implementation uses PyTorch (Paszke et al., 2019), Transformers (Wolf et al., 2020), PEFT, and Ultralytics.

YOLO11s is reconstructed from public weights for 30 epochs using 1,280-pixel tiles, batch 16, and seed 3407. Tiled FP16 inference uses 160×1280 inputs and global NMS IoU 0.50. VLM decoding is greedy, with 12,544–200,704 pixels per view and token caps 384/256 for Stages 1/2. Appendix A records complete settings, software versions, and cache dependencies.

Evaluation measures. Because the private evaluator is unavailable, we implement an equal-weight local proxy for text correction and grounded edits:

$$S_{\text{proxy}} = \frac{1}{N} \sum_{i=1}^{N} \frac{F^{\text{text}}_{0.5,i} + F^{\text{edit}}_{0.5,i}}{2}, \qquad (6)$$

where $F_{0.5} = 1.25PR/(0.25P + R)$. Both component scores are computed per sentence and then macro-averaged; the composite averages their per-sentence mean, rather than pooling edit counts over the corpus.

Text scoring. Reference and predicted edits are extracted from the same released *gold source transcription*, `source_ground_truth`, to the reference and predicted targets respectively. Character-level `SequenceMatcher` with `autojunk=False` yields insertion, deletion, and substitution spans. Matches require identical type, half-open source span, and replacement string; multiset intersection limits repeated matches to reference multiplicity. We use this local differencer, not ChERRANT (Zhang et al., 2022) or ERRANT (Bryant et al., 2017).

Grounded matching. Predictions are visited in stored order and greedily matched to the highest-IoU unused reference with identical type and replacement text, requiring IoU ≥ 0.5. Equal-IoU ties select the first eligible reference in annotation order. Each side can match once; this is a greedy, order-dependent assignment. Duplicate predictions remain in the precision denominator, including any copy without an eligible unused reference.

Empty sets and strings. For either component, two empty edit sets receive precision, recall, and $F_{0.5} = 1$; if only one is empty, all three are zero. Punctuation and whitespace participate literally in alignment and replacement matching. Text fields are not trimmed or normalized for Unicode, case, traditional/simplified characters, or punctuation. Other edge conventions are in Appendix A.4. Private evaluator conventions are unknown, so these remain local proxies.

Text evaluation uses gold source, whereas inference grounding uses each candidate's *predicted source*. Recognition can therefore hide a necessary source edit despite a correct target (Section 4.7).

Split and operating-point scope. Table 1 separates LoRA fitting, detector fitting/validation, and selection/reporting uses. Essay groups are identifier prefixes before the first underscore. Only 12 of the 245 dev sentences belong to essay groups absent from the full released training split; the other 233 belong to groups present there. The 214-example internal selection set excludes its 55 groups from LoRA training, but all its images occur in detector training, and it also guided model and rule selection. Development images serve detector validation and local configuration analysis. Neither local split is an untouched, fully group-disjoint end-to-end evaluation.

All matched-LoRA crop manifests use confidence-0.20 detections. Final grounding

Configuration	Text	Ground	Composite
Dev: 245 examples, grounding 0.50			
Frozen Stage 2	.2158	.1125	.1641
Matched LoRA (V1)	.2767	.1699	.2233
Guarded V2 / V3	.2787	.1699	.2243
V3+ (post-submission)	.2805	.1699	.2252
Selection holdout: 214, grounding 0.20			
Stage 1	.2135	.0794	.1464
Matched LoRA (V1)	.2654	.1416	.2035
Guarded V2 / V3	.2723	.1437	.2080
V3+ (post-submission)	.2723	.1437	.2080

Table 2: Local system-development record, not isolated component ablations or official scores. Dev's frozen-to-LoRA step changes both adapter and output prompt; dev/selection grounding confidences are 0.50/0.20 and guard fallbacks are frozen Stage 2/Stage 1. The selection set isolates LoRA training only. V3+ is exploratory post-submission work.

in the cached progression uses 0.50 on dev but 0.20 on holdout. The holdout guard falls back to Stage 1, whereas dev/test use the retained frozen Stage-2 baseline. The fresh adapter control instead fixes final grounding at 0.50 on both splits and applies no guard. We keep these protocols separate throughout.

4.2 Official competition outcome

According to the retained team record, the August 29 V3 submission ranked **second** with official score **0.5114**. Official component scores were unavailable. Hidden test labels and a private evaluator separate this outcome from our labeled dev/selection analyses. The reconstruction is not assumed byte-identical, and V3+ has no official score (Appendix D).

Our primary evidence for the adapter is the fixed-configuration comparison in Table 3, discussed first below. Table 2 is retained as a system-development record. Its rows can change prompts, grounding operating points, or fallback behavior and must not be read as isolated component ablations.

4.3 Primary comparison: controlled adapter effect

Table 3 is our primary evidence for the adapter's local effect. To separate it from the output-prompt change, we freshly generate both arms with one base model and the same preprocessed tensors per example, toggling only the LoRA adapter. Both use the default text-only prompt, identical cached Stage-1 outputs and crops, greedy decoding

Setting	Text	Ground	Composite
Dev adapter OFF	.2047	.0460	.1254
Dev adapter ON	.2770	.1699	.2234
Holdout adapter OFF	.2034	.0473	.1253
Holdout adapter ON	.2661	.1585	.2123

Table 3: Fresh paired adapter control. Both arms use the same text-only prompt and final grounding confidence 0.50, with no guard. The dev set contains 245 examples; the 214-example holdout isolates only LoRA training and previously served for selection. Its confidence differs from the original 0.20 holdout rows in Table 2.

capped at 256 tokens, and confidence-0.50 grounding, without V2/V3 post-processing. We disable TF32 and record the runtime, input, and weight hashes. This post-submission control does not reuse either arm's old predictions.

On dev, the adapter raises the composite by 0.0981 (Table 3); 59 examples improve, 16 worsen, and 170 tie. Both arms parse all 245 responses without fallback. A 10,000-replicate paired bootstrap over 188 essay groups gives a 95% percentile gain interval of [0.0621, 0.1337]. This estimates the adapter's effect under the text-only contract, not a gain over the best frozen prompt. On the 214-example holdout, the corresponding gain is 0.0870, with 51 improvements, 20 regressions, and 143 ties. All responses parse without fallback. The same bootstrap over its 55 groups gives [0.0456, 0.1327]. These results retain the split limitations described above. This comparison holds inputs, prompt, decoding, and grounding fixed; Table 2 instead records system development with configuration changes. The fresh dev ON output differs from the old cache on four targets and scores 0.2234 rather than 0.2233; we therefore keep the runs separate. Both intervals are conditional on a single trained adapter and fixed predictions, not training-seed uncertainty or an official-score estimate.

4.4 System-development record

Table 2 documents cached local configurations and their combined changes. On dev, the frozen Stage-2 baseline scores 0.1641, matched LoRA 0.2233, and V3 0.2243; on the internal selection set, matched LoRA scores 0.2035 and V3 0.2080. V3's two extra repair rules never trigger on either labeled split, so V2 and V3 tie there. These observations do not establish a measured benefit from the additional repairs.

The dev frozen-to-LoRA configuration differ-

ence is 0.0592 absolute (36.0% relative). A paired bootstrap over 188 essay groups, with 10,000 replicates, gives a 95% percentile difference interval of [0.0243, 0.0940], positive in 99.96% of replicates. This comparison changes both the adapter and the output prompt, from full JSON to text-only JSON. Its interval describes the combined configuration change, not an isolated adapter effect or training-seed uncertainty.

4.5 Exploratory post-submission V3+ case study

The September 12 extension examines a known development error after competition submission. The rank-only and exploratory rank-plus-numbers variants route all 245 dev and 214 internal-selection records before reference labels are loaded for scoring; only predictions enter the selector, not IDs or correctness scores. This prediction-only execution does not remove the experiment's known-error-driven selection bias.

V3+ changes only `1587_15`, improving one dev sample with no regressions and 244 ties. All 214 internal-selection records remain identical. Its dev composite gain is 0.000907 absolute; grounded-edit scores are unchanged. The numeric extension produces exactly the same outputs, so it provides no measured benefit over the rank-only variant.

These observations support a targeted case study, not a general factual-preservation or end-to-end generalization claim. The unchanged grounded-edit score is important: correcting the rank in this case does not recover the missing necessary edit. Section 4.7 examines the case; rule definitions and replay checks are in Appendix B. V3+ has no official competition score.

4.6 Component analysis

Evidence reinspection. Figure 5(a) compares frozen Stage 1 with contextual crops plus a second VLM pass, holding predicted YOLO11n boxes fixed in a test-matched reconstruction. Text $F_{0.5}$ rises from 0.2076 to 0.2158, grounded-edit $F_{0.5}$ from 0.0944 to 0.0982, and the composite from 0.1510 to 0.1570 (0.0060 absolute; 4.0% relative). This isolates the combined reinspection step, not crops alone or the submitted detector choice.

Corpus OCR character error rate decreases from 0.0531 to 0.0524. Expanded crops cover the centers of 87.59% of gold edits, although only 13.79% of raw anchors reach IoU 0.5. This distinction supports using recall-oriented crops for evidence and a separate detector for precise final grounding.

Risk guard and additional probes. Figure 5(b) and Table 2 show that the V2 guard adds 0.0045 on holdout and 0.0010 on dev. V3's two extra repairs affect two test inputs, but no labeled examples; hidden test labels are unavailable, so we make no score-based claim for those changes. Appendix C separately records early auxiliary-context experiments, a discarded direct-image LoRA, threshold sweeps, and post-hoc detector-capacity probes. They are not additional controlled ablations of the final system.

4.7 Qualitative analysis

Figure 3 illustrates cached predictions at grounding confidence 0.50. The first two cases are unchanged from V3 to V3+; the third shows the patch's only observed intervention. They are selected examples, not an unbiased accuracy estimate or isolated adapter effect.

Text recovery versus localization. In `543_6`, V3 recovers the correct phrase about resting at home, but also normalizes the erroneous character in its predicted source. The source–target difference therefore emits no edit: the text is correct, but the gold correction is not localized. In `1969_7`, V3 instead inserts the missing character in "reason" and matches both the reference text and insertion location.

Exploratory rank correction without complete grounding. In `1587_15`, old V3 changes "runner-up" to "champion." Neither length nor placeholder checks detect the substitution. The five prediction views agree on "runner-up," so V3+ selects the complete conservative record and removes the erroneous rank-edit box. Text $F_{0.5}$ rises from 0.5556 to 1, but edit $F_{0.5}$ remains zero: both candidates already normalize a different erroneous source character in "championship," hiding that necessary correction from differencing. The composite therefore rises from 0.2778 to 0.5, not to 1. This one case illustrates the distinction between restoring a rank value, faithfully recognizing the source, and grounding all necessary edits. It does not establish general factual-preservation behavior.

5 Conclusion

Team xc-皮 波's GroundCorrect separates recognition, verification, and grounding for the

Sentence-Level Track of VisCGEC-v2.0. The retained record reports second place and 0.5114 for official V3. Fixed-configuration controls provide the adapter evidence; other runs document system development. Exploratory V3+ changes one dev example, without selection-set or grounded-edit improvement. Recognition can hide necessary edits despite correct targets. Broader generalization requires repeated training and fully separated end-to-end evaluation.

Limitations

Recognition and retrieval errors remain possible; the differencer does not explicitly recover WordOrder. The local scorer is not the private evaluator. Dev overlaps training essays; holdout excludes only LoRA training and informed selection. Controls use one checkpoint, not repeated training seeds. V3's two repairs never fire on labeled splits. The known-error-driven fact patch uses correlated views and restricted expressions: one improvement does not establish unseen-data safety. Missing original weights and run records prevent claims of byte-exact reproduction of the scored archive.

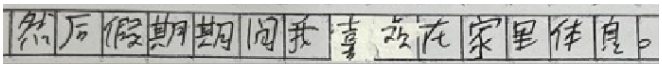
Ethical Considerations

We use released learner handwriting and train/dev annotations without personally identifying metadata. The archived run used no manual test corrections. Later reconstruction audits inspected some public test images, without test targets or hidden labels. This system should assist teachers and learners, not make high-stakes judgments without review.

Declaration on Generative AI

AI tools assisted writing, figures, and code. The authors verified results against saved artifacts and remain responsible for all claims, citations, and implementation.

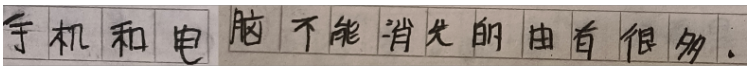
(a) OCR recovery: improved text, incomplete localization — 543_6



		F0.5: text / edit
Reference	然后假期期间我喜欢在家里休息。	
Frozen S2	然后假期期间我改为在家里休息。	0.56 / 0.00
V3+	然后假期期间我喜欢在家里休息。	1.00 / 0.00

V3+ = V3: text matches the reference, but source normalization hides the gold edit.

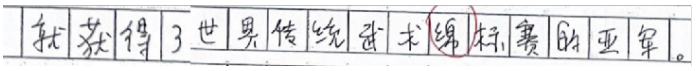
(b) Missing-character insertion: correct text and localization — 1969_7



		F0.5: text / edit
Reference	手机和电脑不能消失的理由有很多。	
Frozen S2	手机和电脑不能消失的有很多。	0.00 / 0.00
V3+	手机和电脑不能消失的理由有很多。	1.00 / 1.00

V3+ = V3: the missing character is inserted; the baseline deletes its neighbor.

(c) Rank protection: correct text, remaining localization failure — 1587_15



		F0.5: text / edit
Reference	就获得了世界传统武术锦标赛的亚军。	
Old V3	就获得了世界传统武术锦标赛的冠军。	0.56 / 0.00
V3+	就获得了世界传统武术锦标赛的亚军。	1.00 / 0.00

V3+ restores runner-up via the whole conservative record; the gold edit remains unlocalized.

Figure 3: Selected dev cases with unretouched images and cached outputs. Ruled lines and marks are original; amber marks baseline differences, green/red matching/nonmatching outputs. Scores are text/edit $F_{0.5}$. Only the third case changes under exploratory V3+.

References

Fatih Cagatay Akyon, Sinan Onur Altinuc, and Alptekin Temizel. 2022. Slicing aided hyper inference and fine-tuning for small object detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 966–970. IEEE.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Zonghao Guo, Ruyi Xu, Yuan Yao, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, and Gao Huang. 2025. LLaVA-UHD: An LMM perceiving any aspect ratio and high-resolution images. In *Computer Vision – ECCV 2024*, volume 15141 of *Lecture Notes in Computer Science*, pages 390–406, Cham. Springer Nature Switzerland.

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*.

Glenn Jocher and Jing Qiu. 2024. Ultralytics YOLO11.

Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2023. TrOCR: Transformer-based optical character recognition with pre-trained models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):13094–13102.

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*, volume 32. Curran Associates, Inc.

Xin Sun and Houfeng Wang. 2022. Adjusting the precision-recall trade-off with align-and-predict decoding for grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 686–693, Dublin, Ireland. Association for Computational Linguistics.

Xiaoman Wang, Dan Yuan, Xin Liu, Yike Zhao, Xiaoxiao Zhang, Xizhi Chen, and Yunshi Lan. 2025. VisCGEC: Benchmarking the visual Chinese grammatical error correction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5054–5068, Albuquerque, New Mexico. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 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, Online. Association for Computational Linguistics.

Lvxiaowei Xu, Jianwang Wu, Jiawei Peng, Jiayu Fu, and Ming Cai. 2022. FCGEC: Fine-grained corpus for Chinese grammatical error correction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1900–1918, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Haihui Yang and Xiaojun Quan. 2024. Alirector: Alignment-enhanced Chinese grammatical error corrector. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2531–2546, Bangkok, Thailand. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 11328–11339. PMLR.

Yue Zhang, Zhenghua Li, Zuyi Bao, Jiacheng Li, Bo Zhang, Chen Li, Fei Huang, and Min Zhang. 2022. MuCGEC: a multi-reference multi-source evaluation dataset for Chinese grammatical error correction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3118–3130, Seattle, United States. Association for Computational Linguistics.

Yuanyuan Zhao, Nan Jiang, Weiwei Sun, and Xiaojun Wan. 2018. Overview of the NLPCC 2018 shared task: Grammatical error correction. In *Natural Language Processing and Chinese Computing*, volume 11109 of *Lecture Notes in Computer Science*, pages 439–445, Cham. Springer.

A Reproducibility Configuration

This appendix specifies the preserved local reconstruction and its evaluation inputs. The accompanying supplement contains complete UTF-8 prompt templates, an anonymized machine-readable configuration, split identifiers, artifact checksums, command examples, and a result index. These materials document the available reconstruction; they do not establish byte identity with the scored archive. They are supplied with the paper, rather than described as an already-public repository.

A.1 Verified training configuration

Table 4 combines the final training report, adapter configuration, and the saved training arguments at step 219. This distinction matters: the executed run saved every 75 steps, whereas the current command-line default is 100. Stage-1 training-data preparation used an inference batch of eight; the LoRA collator uses one example per device.

Parameter	Preserved value
Base / visual encoder	Frozen Qwen3-VL-4B
LoRA rank / alpha	8 / 16
LoRA dropout / bias	0.05 / none
Target projections	q, k, v, o; gate, up, down
Trainable parameters	16,515,072 (0.371%)
Device batch / accumulation	1 / 8
Epochs / optimizer steps	1 / 219
Learning rate	5×10^{-5}
Schedule / warmup ratio	Cosine / 0.05
Optimizer / weight decay	Fused AdamW / 0
Adam betas / epsilon	$(0.9, 0.999)$ / 10^{-8}
Gradient clipping norm	1.0
Precision	bfloat16; TF32 enabled
Checkpointing	Enabled; non-reentrant
Training / data seed	20260829 / 20260829
Save / logging interval	75 / 5 steps

Table 4: Verified local LoRA settings. The effective batch size is eight on one device. Projection names denote the corresponding `_proj` modules.

The assistant target is a compact JSON object containing only `source_text` and `target_text`. The collator verifies that the tokenized prompt is an exact prefix of the complete conversation and sets all prompt-token labels to -100. Consequently, image and user-prompt tokens receive no direct language-model loss. The supervised suffix includes the assistant answer and its chat-template terminator. The base parameters and visual encoder remain frozen.

The local environment used Python 3.11, PyTorch 2.8.0 with CUDA 12.8, Transformers 4.57.3, PEFT 0.18.1, Accelerate 1.10.1, Pillow 11.3.0, and Ultralytics 8.4.118. The full base-model revision is `ebb281ec70b05090aa6165b016eac8ec08e71b17`. The adapter tensor SHA-256 is `20ceecc9db7e62fd26a6003842d87b5a5586b9022e55ebc70d0abd32707ff578`. Hashes of the reconstructed detector, fallback predictions, and final output are supplied in the configuration.

A.2 Prompt and inference contract

Both passes use the same Chinese system instruction. Stage 1 receives the full image followed by its user instruction. Stage 2 receives the full image, up to four crops, and then a compact context object containing crop coordinates, retrieval scores and sources, and the Stage-1 source and target. The retained `pegasus_target_text` field is an empty string in this path. No reference target is placed in this context. The default Stage-2 prompt is used during LoRA training; the conservative prompt is an inference-only alternative using the same adapter.

The reproduced inference settings use greedy decoding, bfloat16, scaled dot-product attention, and per-image pixel bounds of 12,544–200,704. The preserved scripts specify 384 generated tokens for Stage 1 and 256 for text-only Stage 2. These token limits are made explicit in the supplied commands, because the saved inference reports did not serialize the complete original command line. Invalid or empty Stage-2 responses fall back to the cached Stage-1 hypothesis before deterministic grounding.

Candidate retrieval uses cached YOLO11s detections at confidence 0.20. The released-dev main comparison re-grounds fixed predictions at confidence 0.50; the recorded internal holdout instead uses the confidence-0.20 detections stored in its training manifest. Final test grounding uses a separate confidence-0.50 cache. These operating points should not be silently unified when comparing the preserved scores.

A.3 Split scope and cached dependencies

Essay groups are the identifier prefix before the first underscore. The training script sorts the 554 groups, shuffles them using seed 20260829, and reserves 55 groups, yielding 1,746 LoRA-training and 214 holdout examples. The supplement lists the reserved groups and hashes the ordered exam-

ple IDs independently of machine-specific paths. This isolation applies to LoRA training only: the detector was trained using annotations from the complete released training split, including these holdout images. The same holdout also informed guard selection; it is not an untouched end-to-end test set.

V3 selects among the default LoRA output, the conservative output, and a retained baseline submission at test time. For the labeled guard comparison, the holdout fallback is cached Stage 1, whereas dev uses frozen Stage 2. The supplement includes a CPU-only replay and per-example routing records for both operating points. The test baseline's complete prediction records, including boxes, are an input dependency. Only the linguistic-repair branch recomputes edits with the cached detector boxes. Exact output reproduction therefore requires the cached Stage-1 outputs, crop manifests, detector boxes, and baseline JSON, not merely the adapter weights. The result index separates available artifact evidence from command defaults and notes dependencies that are not bundled. Fresh inference and retraining can test the procedure but are not promised to reproduce all stored bytes.

A.4 Local evaluation edge conventions

Text edits are compared as exact tuples of operation, half-open source span, and replacement text. Text precision and recall use multiset intersection, so repeated exact predictions match only up to gold multiplicity. Grounded matching is greedy in prediction order; equal-IoU ties retain the first unused eligible reference in annotation order. Coordinate IoU uses differences without an inclusive-endpoint $+1$; a zero-union pair has IoU zero. An empty dataset returns zero macro aggregates. These implementation conventions do not claim identity with the private evaluator.

JSON parsing trims its outer generated representation and Markdown wrappers, not parsed string values. Scoring retains literal punctuation, whitespace, Unicode forms, case, and traditional/simplified characters. The source field for text scoring is the released gold transcription, whereas candidate grounding uses predicted source; no gold source is introduced into test-time grounding.

A.5 Alignment optimality and complexity

For Equation 5, removing the last match/skip transition leaves an optimal consumed-prefix alignment: a cheaper prefix would otherwise lower the full path cost. This gives the dynamic program's optimal substructure. The algorithm uses $O(nm)$ time and space, with $O(n + m)$ backtracking. Its optimum concerns the defined positional cost, not the correctness of source recognition or detector boxes.

B V3 Finalization and Exploratory V3+ Details

Figure 4 shows V3's first-match routing followed by V3+'s default-only fact check. Candidates d and c use default and conservative prompts with the same LoRA; b is the retained pre-LoRA baseline.

B.1 Priority consistency repairs

V3 first checks two narrow repairs:

1. Restore a Stage-1 comma at a preserved *if*-clause boundary.
2. Complete an asymmetric comparison's missing activity from the opposite predicate, limited to five activities: calling, spending time together, communicating, chatting, and meeting.

A match returns the repaired default target, rebuilding edits from the default source and cached detector boxes. Only this branch recomputes edits; all other branches retain their grounded records.

B.2 Placeholder and length routing

Otherwise, the V2 guard first checks for uppercase `X` in the default target but absent from the Stage-1 target. When triggered, it returns c if the conservative target has no `X`, and b otherwise.

Only without this trigger does it check default length growth of at least three characters. Growth is each candidate's target length minus its own source length. This branch returns c if its growth is less than three, and b otherwise. Growth signals risk, not proven hallucination. Without any trigger, return d. Clearing a branch's specific trigger suffices; returning a conservative candidate after the placeholder check does not require passing the later length check. Both risk branches select complete cached records rather than merging targets with another candidate's edits.

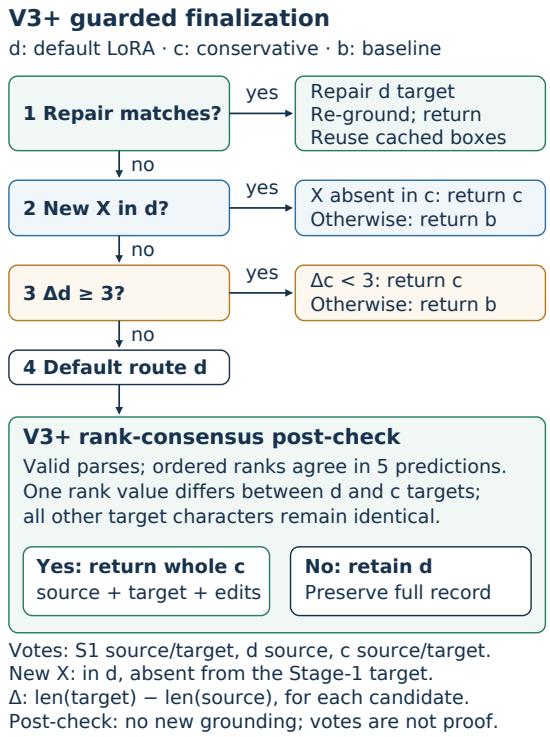


Figure 4: V3+ selection. Original V3 repairs and fallbacks are preserved; only the original default route reaches the new rank check. A fact-triggered switch copies the entire conservative candidate without recomputing edits.

B.3 Observed routing and scope

The unpatched V3 local test audit counts 232 default, two repaired, five conservative, and seven baseline records. The original holdout proxy improves from 0.2035 to 0.2080, using confidence-0.20 grounding and Stage-1 fallback; dev/test use frozen Stage-2 fallback. This holdout isolates LoRA, not detector training, and served model/guard selection. Neither V3 repair fires on dev/holdout, so V2 and V3 tie there. The two test interventions have no measured hidden-test score improvement.

B.4 Prediction-only fact post-check

The post-check is exploratory work from September 12, after competition submission. The rule-level definitions below are retained for reproducibility, not as evidence of a general factual-preservation capability.

The current V3+ configuration enables rank protection only. It canonicalizes the three named podium ranks and supported positive ordinal-place numerals. Specified compound suffixes such as those denoting competitions or teams are excluded. Mixed Arabic/Chinese or malformed ordinal expressions reject the entire decision rather than contributing partial consensus.

The Stage-1 parse-success field must exist and indicate no error. Both Stage-2 candidates must have the expected generation provenance, directly parseable raw JSON, and source/target strings matching the preserved cache. Missing fields, additional target rewrites, fact-count changes, or conflicting signatures preserve the V3 record. The exploratory numeric extension covers only Arabic quantities with enumerated units, treats the literal sign/unit as part of the type, and rejects comma-separated number fragments.

The dev audit changes one of 237 original default routes to the conservative candidate: final provenance counts are 236 default, five conservative, and four baseline records. Holdout preserves 203 default, eight conservative, and three baseline records. Neither variant changes any earlier repair or fallback branch. Both variants' full outputs, sample scores, and routing audits are included in the supplement. These counts do not describe a new official test submission.

B.5 Formal post-check definition

Let q_3 be V3's selected complete record and ρ_3 its recorded route. Let $F(s)$ contain the ordered recognized fact types and normalized values in string s; spans and literal surfaces are stored separately. For default and conservative records $q_d = (x_d, y_d, E_d)$ and $q_c = (x_c, y_c, E_c)$, the support views are

$$\mathcal{V} = (x^{(0)}, y^{(0)}, x_d, x_c, y_c). \tag{7}$$

Consensus $\mathcal{A}$ requires $F(\mathcal{V}_1)$ to be nonempty and equal to $F(v)$ for every $v \in \mathcal{V}$. These are correlated predictions, not independent votes or a calibrated correctness probability.

Define $M(s)$ as a literal text skeleton obtained by replacing each recognized span with its fact type. Condition $\mathcal{B}$ requires y_d, y_c to contain the same ordered fact types, exactly one different value, exactly one different fact surface, and $M(y_d) = M(y_c)$. Punctuation, non-fact wording, and all other fact surfaces must remain unchanged. With P denoting verified parsing and valid inputs, the post-check is

$$q_+ = \begin{cases} q_c, & \rho_3 = d \ \wedge \ P \ \wedge \mathcal{A} \wedge \mathcal{B}, \\ q_3, & \text{otherwise.} \end{cases} \tag{8}$$

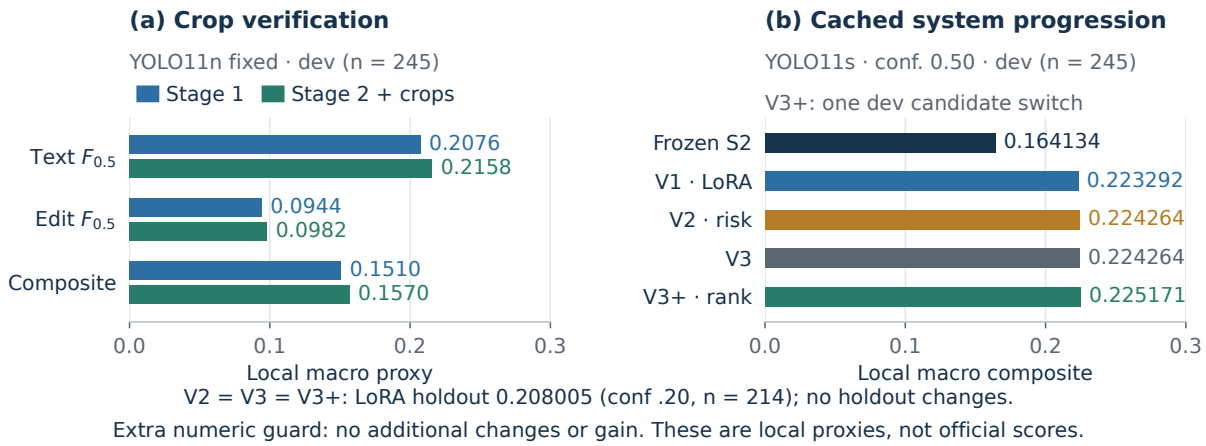


Figure 5: Two local dev comparisons with separate grounding setups. (a) Frozen Stage 1 versus evidence reinspection with fixed YOLO11n boxes. (b) Cached system progression through V3+ with YOLO11s confidence 0.50; the frozen-to-LoRA step changes the output prompt as well as the adapter. The adapter-only control is Table 3. The footer's selection score uses grounding 0.20.

Here $\rho_3 = d$ means the original, unrepaired default route, not merely a text match. Parsing must explicitly succeed at Stage 1; both Stage-2 outputs must be non-fallback generations whose parsed source/target fields equal the cache.

Equation 8 preserves every non-default V3 route and ensures $q_+ \in \{q_3, q_c\}$. Selecting y_c also selects x_c and E_c; no candidate's sentence is paired with another candidate's edits. These are structural guarantees, not guarantees of semantic accuracy: shared OCR errors and missing edits can survive consensus. Output schema checks cover IDs, operation types, nonempty text, integer coordinates, and image bounds.

The replay verifies all original V3 per-sample scores and aggregates with zero differences and checks whole-record identity. The retained experiment reports 31 passing synthetic boundary tests covering parsing, invalid facts, extra rewrites, and preservation of earlier routes. These implementation checks do not extend the empirical result beyond the single changed development example.

C Auxiliary Development Analyses

These development-only and post-submission diagnostics do not measure the scored archive.

C.1 Development-only textual context and oracle boxes

An early configuration used public OCR disagreement, a Pegasus target (Zhang et al., 2020) for proposals and Stage-2 context, public source transcription, and annotated character boxes. Neither textual input was available to the submitted test runner. Its proxy rose from 0.1561 at Stage 1 to 0.1637 after verification. These are auxiliary, not controlled or main development results. Gold targets and edit boxes are never used for test-time crop selection.

C.2 Early direct-image adaptation

A one-epoch direct-image Qwen3-VL LoRA trained 11.8M language-attention parameters (0.265% of 4.45B), freezing the visual encoder (Hu et al., 2022). On the same first 64 auxiliary-dev examples, its evidence-pipeline proxy was 0.1586 versus frozen 0.1876. Training–inference prompt mismatch is one possible explanation; the adapter was discarded. The later adapter differs in output prompt, trainable modules, and evaluation setup, preventing attribution solely to format matching.

C.3 Fixed-prediction confidence sweep

With fixed frozen-Qwen predictions and reconstructed YOLO11s boxes, thresholds 0.45, 0.48, 0.50, and 0.52 all score 0.1641. Five-fold essay-group selection chooses 0.50 throughout, with no concatenated held-out gain. Retaining 0.50 does not establish a threshold benefit for V3.

C.4 Detector-capacity probe

Post-submission YOLO11n/s retrainings used identical data and 30 epochs, yielding dev $\text{mAP}_{50:95}$ of 0.801/0.800. Frozen Qwen outputs scored 0.1570/0.1641 with n/s localization. This

compares post-hoc retrainings, not an n-to-s upgrade of the award system, which already used s.

D Retained-Artifact Integrity Checks

The September 12 V3+ experiment preserves all older archives and records hashes and predictions for 245 dev and 214 holdout examples. Its official hidden-test score is unknown; its local gain is not attributed to older runs.

The retained August 29 candidate has 246 unique records, 245 edits, 85 empty edit lists, nonempty targets, and valid in-image integer boxes. Its timestamps predate upload, but do not prove that its exact bytes were scored. The September 11 reconstructed V3 archive has 246 records and 252 edits with the same schema checks. Both have one root member, `submission_sentence.json`.

Hashes and the older frozen-pipeline audit are in the companion supplement; that audit is not evidence for V3+ accuracy.

Cross-Lingual TTS for Indian Languages: A Benchmark and Inference-Time Duration Estimation Strategies

Adarsh Arigala* **Arjun Gangwar*** **Srinivasan Umesh**
SPRING Lab, Indian Institute of Technology Madras, India
arigalaadarsh780@gmail.com arjungangwar@gmail.com
umeshs@ee.iitm.ac.in

Abstract

Recent alignment-free Text-to-Speech (TTS) models achieve high synthesis quality by removing explicit alignment modules and simplifying training. However, they often struggle in cross-lingual scenarios due to differences in character durations across languages, leading to duration mismatch and degraded speech quality. Existing approaches typically address this issue using explicit duration predictors, increasing model complexity. Systematic evaluation of TTS performance for Indian languages in cross-lingual settings remains under-explored due to the lack of standardized evaluation benchmarks. In this work, we construct a cross-lingual evaluation benchmark covering English and nine Indian languages and propose a language-aware inference-time duration estimation strategy for alignment-free TTS that incorporates language-specific duration-per character statistics without retraining or architectural modifications. We further develop heuristic formulations that refine duration estimation based on language-dependent duration variations. Experiments on bilingual and multilingual A-DMA models demonstrate improved speech naturalness, intelligibility, and speaker similarity while reducing duration mismatch. [1]

1 Introduction

Text-to-Speech (TTS) systems have achieved significant progress with the emergence of neural architectures, enabling natural and expressive speech synthesis. Recent alignment-free models such as E2-TTS (Eskimez et al., 2024), F5-TTS (Chen et al., 2025) and A-DMA (Choi et al., 2025), simplify training by removing explicit alignment modules while maintaining high synthesis quality. Although these models perform well when reference and target speech is in the same language, they often struggle in cross-lingual scenarios where reference and target languages differ. This happens due to differences in writing systems, orthographic characteristics, and language-specific duration-per character variations across languages. Moreover, in F5-TTS, duration estimation is based on the number of UTF-8 bytes obtained by encoding the input text, rather than the actual number of characters. Since characters from different writing systems occupy different numbers of UTF-8 bytes and languages exhibit different duration-per-character patterns, the estimated duration can become inaccurate, leading to duration mismatch and degraded synthesis quality.

To address these challenges, recent works have explored various strategies to improve duration modeling and generation quality in alignment-free TTS. Cross-Lingual F5-TTS (Liu et al., 2026) removes the dependency on reference transcripts in flow-matching TTS and enables cross-lingual voice cloning through forced alignment during training. However, it introduces additional training complexity by requiring MMS-based alignment (Pratap et al., 2024) preprocessing and separately trained speaking-rate predictors, making the overall pipeline more complex than standard alignment-free TTS systems.

Another work, PFluxTTS (Pankov et al., 2026) proposes a hybrid flow-matching framework that combines a duration-guided model and an alignment-free model through inference-time fusion, improving the trade-off between alignment stability and naturalness. However, it relies on an explicit duration predictor, which reduces the benefits of alignment-free TTS by reintroducing duration modeling.

Most people in India are bilingual and frequently switch between English and their regional languages during everyday conversations. Consequently, cross-lingual TTS between English and

*equal contribution

[1] The dataset and code recipe are available at: https://github.com/Cross-Lingual-TTS-for-Indian-Languages/Cross-Lingual-TTS-Indic-Test-Set/

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 47–55
November 9, 2026 ©2026 Association for Computational Linguistics

Indian languages has become increasingly important. These languages exhibit substantial differences in orthographic characteristics and character-to-speech duration, making accurate duration estimation essential for stable cross-lingual speech synthesis. Despite its practical importance, systematic evaluation of TTS performance for Indian languages in cross-lingual settings remains underexplored due to the lack of standardized evaluation benchmarks and limited investigation of duration adaptation strategies for alignment-free synthesis.

To address these limitations, we propose a cross-lingual evaluation benchmark covering nine Indian languages and English and introduce language-aware inference-time duration estimation strategies for alignment-free TTS. Our approach incorporates language-specific duration-per character statistics directly during inference without requiring retraining or additional model components. Specifically, we design heuristic formulations that adjust the estimated duration according to language-specific duration-per-character statistics, enabling effective cross-lingual duration adaptation.

This work makes the following contributions:

- We construct a cross-lingual evaluation benchmark covering nine Indian languages and English, evaluating two settings: **(i) English Reference and Indic Generation**, where English reference speech is used to generate Indian language speech and **(ii) Indic Reference and English Generation**, where Indian language reference speech is used to generate English speech, to systematically analyze TTS performance in cross-lingual scenarios.

- We train bilingual and multilingual A-DMA models for Indian languages from scratch to investigate their performance in cross-lingual settings.[2]

- We propose inference-time language-aware duration estimation strategies for alignment-free TTS in cross-lingual settings, addressing duration mismatch by incorporating language-specific duration-per character information into the generation process.

[2]**Demo Samples**: https://cross-lingual-tts-for-indian-languages.github.io/Cross-Lingual-TTS-Demo/

Language Indic+Eng	Duration (Hrs)	No. of Indic Utt.	No. of Eng Utt.	Total Utt.
Kannada	2.00	540	588	1128
Telugu	2.00	531	588	1119
Tamil	2.00	591	588	1179
Hindi	1.35	171	588	759
Malayalam	2.00	554	588	1142
Punjabi	2.00	520	588	1108
Bengali	2.00	543	588	1131
Odia	2.00	532	588	1120
Marathi	2.00	531	588	1119

Table 1: Cross-lingual evaluation set statistics.

2 Methodology

2.1 Baselines

We adopt A-DMA (Choi et al., 2025) framework for our experiments. It is based on the F5-TTS (Chen et al., 2025) backbone and introduces a training objective that combines conditional flow matching with dual-modality text and speech representation alignment losses to accelerate convergence. We utilize self-supervised representations (SSL) from the 21st layer of data2vec-AQC (Lodagala et al., 2023) to compute the speech-level alignment loss. The model is trained using the conditional flow matching loss from F5-TTS as the primary generative objective. A Connectionist Temporal Classification (CTC) based text alignment loss (Graves et al., 2006) is applied at an intermediate layer to for early character-level alignment, and a speech-level alignment loss is used at a later layer to enforce alignment in the speech representation space.

$$\mathcal{L} = \mathcal{L}_{\text{CFM}} + \lambda_{\text{text}}\mathcal{L}_{\text{text}} + \lambda_{\text{speech}}\mathcal{L}_{\text{speech}},$$

where $\lambda_{\text{text}} = 0.1$ and $\lambda_{\text{speech}} = 1.0$. This combined objective improves convergence and speech synthesis quality.

2.2 Cross-lingual Evaluation Dataset

To test the cross-lingual generalization capability of the A-DMA (Choi et al., 2025) model, we constructed a dedicated cross-lingual evaluation benchmark covering nine Indian languages and English. In this setting, the reference speech and the text being synthesized belong to different languages.

For each Indian-language and English pair in the cross-lingual evaluation setting, we constructed approximately two hours of evaluation data, allocating one hour to (i) English Reference and Indic Generation and one hour to (ii) Indic Reference and

English Generation. The Indian-language speech was selected from the Rasa (Srinivasa Varadhan et al., 2024) and IndicVoices-R (Sankar et al., 2024) test splits, while the English speech was sampled from the IndicTTS (Consortium et al., 2023) training split. [3]

We considered two cross-lingual evaluation scenarios: (i) Indian-language reference speech with English generation text, and (ii) English reference speech with Indian-language generation text. In both scenarios, reference and target utterances were randomly sampled subject to duration constraints of 4–7 seconds and 4–10 seconds, respectively. Detailed statistics of the constructed benchmark, including the total number of utterances for each language pair, are provided in Table 1.

2.3 Model Setup

We trained the original A-DMA (Choi et al., 2025) model for Indian languages in two settings: bilingual and multilingual. For the bilingual setup, each model was trained on one Indian language combined with a 30-hour subset of Indian English data (Consortium et al., 2023), while the multilingual model was trained on 24 languages as described in Section 3.1.

The bilingual model used the F5-TTS Small architecture with 18 DiT layers (Peebles and Xie, 2023), 12 attention heads, and 768-dimensional embeddings, together with a 4-layer ConvNeXtV2 (Woo et al., 2023) module with 512-dimensional embeddings, totaling 159M parameters. It was trained for 1.2M steps with 20k warmup updates and an effective batch size of 0.75 hours.

The multilingual model used F5-TTS Base with 22 DiT layers (Peebles and Xie, 2023), 16 attention heads, and 1024-dimensional embeddings, along with a 4-layer ConvNeXtV2 (Woo et al., 2023) module with 512-dimensional embeddings, totaling 335.8M parameters. It was trained for 300k updates with 45k warmup updates and an effective batch size of 2.09 hours. For both settings, 100-channel mel spectrograms were extracted at a 24 kHz sampling rate with a hop size of 256 samples. Waveform synthesis was performed using the pre-trained Vocos vocoder (Siuzdak, 2024), and optimization was carried out using AdamW (Loshchilov and Hutter, 2019) with a peak learning rate of 7.5×10^{-5}.

[3] Indian English is available only in IndicTTS-v2, so we take 500 utterances from its 250-hour training set as a held-out set. Each language has only two speakers, and all speakers also recorded Indian English.

Language	Indic Voices-R	Indic TTS-v2	Indic TTS-v3	Rasa	Total (Hrs)
Assamese	173.77	25.73	20.33	58.32	278.15
Bengali	110.20	20.09	20.63	56.76	207.68
Bodo	170.48	0	14.47	52.31	237.26
Gujarati	4.50	26.49	18.34	25.39	74.71
Hindi	72.75	26.23	22.51	0	121.49
Kannada	43.22	22.32	11.54	54.62	131.70
Kashmiri	63.07	0	0	0	63.07
Konkani	49.33	0	11.00	26.33	86.66
Maithili	80.32	0	9.73	29.34	119.39
Malayalam	80.64	20.91	25.02	51.63	178.20
Manipuri	21.00	17.65	16.97	0	55.62
Marathi	48.73	22.37	22.78	55.36	149.24
Nepali	104.31	0	7.24	55.09	166.64
Odia	69.32	19.19	22.41	47.91	158.84
Punjabi	71.62	11.21	22.63	55.84	161.30
Sanskrit	33.25	0	22.31	51.47	107.03
Santali	72.76	0	0	0	72.76
Sindhi	9.03	0	14.53	0	23.56
Tamil	98.03	26.59	23.14	29.94	177.70
Telugu	134.86	36.71	23.67	52.27	247.50
Urdu	76.36	0	0	0	76.36
Indian Eng.	0	251.09	0	0	251.09
Dogri	0	0	13.19	48.04	61.23
Rajasthani	0	20.39	20.92	0	41.31
Total	**1587.55**	**546.97**	**363.36**	**750.62**	**3248.50**

Table 2: Training dataset statistics across multilingual speech corpora in hours.

2.4 Language-Aware Heuristic Duration Estimation

2.4.1 Duration Estimation in Original F5-TTS

Alignment-free TTS models require predicting target speech duration prior to synthesis. A-DMA uses the F5-TTS duration estimation formulation defined as:

$$D_{F5} = D_{ref} + \left\lfloor \frac{D_{ref}}{L_{ref}} \cdot \frac{L_{gen}}{s} \right\rfloor \quad (1)$$

where D_{ref} denotes the reference speech duration, L_{ref} and L_{gen} represent the number of characters in reference and generation text, which are converted into UTF-8 byte encoding for duration estimation, and s denotes the local speed factor, which we set to 1.

While the standard F5-TTS formulation performs well when reference and generated speech share the same language, it computes text length using UTF-8 byte-level encoding, which is not suitable for multilingual settings. In cross-lingual generation, this becomes problematic because Indian languages typically require multiple bytes per character, whereas English characters generally require a single byte. Consequently, byte-level length estimation leads to inconsistent scaling in duration prediction across languages. Previous works have

addressed this limitation by incorporating explicit speaking-rate predictors (Liu et al., 2026) or duration predictors (Pankov et al., 2026), which require modifications to the original F5-TTS architecture and model training.

To overcome this issue, we reformulate Original F5-TTS in a character-level space and report results using this variant, referred to as ***Modified F5-TTS***. Additionally, we propose two language-aware duration modeling strategies, *Heuristic-1* and *Heuristic-2*, which incorporate duration-per character information into the F5-TTS formulation through fusion and additive interaction mechanisms, respectively.

2.4.2 Heuristic-1: Language-Aware Duration Model

The heuristic (H1) extends the original F5 duration estimation by incorporating language-specific duration-per-character statistics.

$$D_{H1} = D_{ref} + \alpha \left\lfloor \frac{D_{ref}}{L_{ref}} \cdot \frac{L_{gen}}{s} \right\rfloor + \beta(D_{ind} + D_{eng}) \quad (2)$$

where $\alpha = 0.7$ and $\beta = 0.3$ in all experiments. The effect of varying α and β is analyzed in Section 5.2. D_{F5} denotes the duration estimated by F5-TTS. Duration estimations for Indian languages and English are computed as follows to capture language-dependent duration variations:

$$D_{ind} = \frac{d^c_{ind} \times C_{ind}}{s} \qquad D_{eng} = \frac{d^c_{eng} \times C_{eng}}{s} \quad (3)$$

where D_{ref} denotes the reference speech duration, d^c_{ind}, d^c_{eng} denote the duration-per character statistics for Indic languages and English, respectively, C_{eng} and C_{ind} denote the number of English and Indic characters in the generated text, and s denotes the local speed factor, which we set to 1.

The language-specific duration-per character are estimated from multilingual training corpora. For each utterance, the duration-per character is computed as:

$$d^c = \frac{L_{audio}}{C_{text}} = \frac{N_{samples}/H}{C_{text}} \quad (4)$$

where $N_{samples}$ denotes the waveform length, H denotes the hop length, and C_{text} denotes the text length in characters.

To obtain robust language-specific statistics, duration-per-character values are aggregated across all speakers within each language using the corresponding training corpus (Table 2). For example, the Telugu statistic is computed from the Telugu training corpus, and similarly for the remaining languages. The language-specific average duration-per-character is computed as:

$$d^c_{lang} = \frac{1}{N_{\text{utt}}} \sum_{i=1}^{N_{\text{utt}}} D^c_i \quad (5)$$

where N_{utt} denotes the number of utterances per language.

Fig. 1 shows the utterance-level distribution of duration-per-character values for all languages in the training corpora. Although each language exhibits natural variability due to differences in speakers and speaking styles, we use the average duration per character, d^c_{lang}, for each language in the proposed heuristics. The resulting language-specific average duration-per-character statistics are:

Indian English (9.6), Kannada (8.7), Telugu (8.4), Tamil (7.0), Hindi (8.5), Malayalam (6.8), Punjabi (10.1), Bengali (8.1), Odia (8.7), and Marathi (9.8). This formulation combines the original F5 duration estimate with language-aware duration estimation to better handle cross-lingual synthesis scenarios.

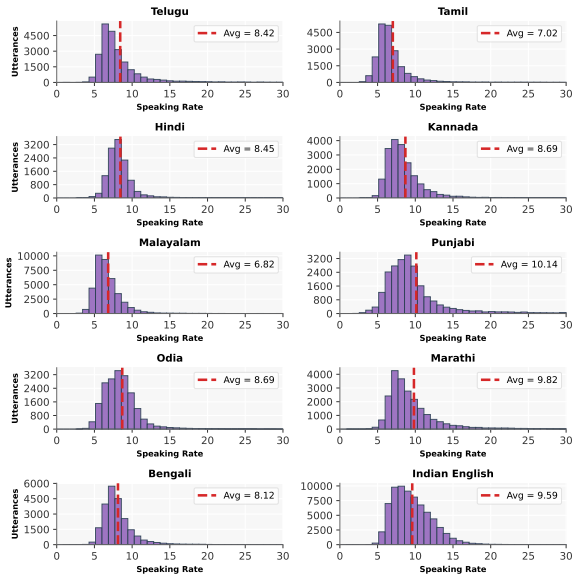


Figure 1: Utterance-level duration-per character distributions for the multilingual training data. The red dashed line indicates the average duration-per character (d^c_{lang}) used in the proposed heuristics.

Model	Original F5-TTS				Modified F5-TTS				Heuristic-1				Heuristic-2			
	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER
Kannada																
GT	33.38	–	3.461	5.88	33.38	–	3.461	5.88	33.38	–	3.461	5.88	33.38	–	3.461	5.88
Bilingual	109.56	0.630	3.070	57.07	26.98	0.613	3.435	4.87	26.91	0.615	3.459	4.79	27.85	0.619	3.468	5.23
Multilingual	55.88	0.641	3.383	22.22	26.45	0.662	3.460	4.17	26.22	0.664	3.475	4.16	25.15	0.667	3.495	3.91
Telugu																
GT	21.06	–	3.119	3.07	21.06	–	3.119	3.07	21.06	–	3.119	3.07	21.06	–	3.119	3.07
Bilingual	82.05	0.662	3.249	48.02	20.68	0.653	3.575	3.49	21.15	0.656	3.590	3.66	23.00	0.658	3.592	5.33
Multilingual	37.66	0.635	3.436	18.02	16.73	0.664	3.508	2.08	16.40	0.662	3.508	1.93	16.16	0.668	3.554	1.96
Tamil																
GT	25.39	–	3.357	4.87	25.39	–	3.357	4.87	25.39	–	3.357	4.87	25.39	–	3.357	4.87
Bilingual	105.57	0.630	2.728	52.20	25.17	0.626	3.387	5.29	23.91	0.624	3.390	4.88	26.49	0.627	3.388	6.01
Multilingual	50.95	0.600	3.140	24.02	20.52	0.650	3.346	3.56	20.39	0.646	3.328	3.52	20.98	0.652	3.347	3.63
Hindi																
GT	14.40	–	2.573	5.21	14.40	–	2.573	5.21	14.40	–	2.573	5.21	14.40	–	2.573	5.21
Bilingual	62.45	0.633	3.101	47.01	12.58	0.662	3.608	4.04	13.16	0.656	3.627	4.30	12.87	0.664	3.638	4.31
Multilingual	26.70	0.666	3.435	17.05	11.80	0.679	3.555	3.61	11.66	0.680	3.566	3.42	11.71	0.685	3.585	3.49
Malayalam																
GT	25.89	–	3.284	3.87	25.89	–	3.284	3.87	25.89	–	3.284	3.87	25.89	–	3.284	3.87
Bilingual	121.85	0.579	2.301	56.82	28.91	0.622	3.359	6.22	26.49	0.622	3.403	5.03	31.42	0.624	3.377	7.73
Multilingual	53.42	0.600	3.170	23.81	22.98	0.661	3.343	3.49	22.15	0.660	3.351	3.34	22.45	0.667	3.353	3.44
Punjabi																
GT	15.75	–	3.337	4.81	15.75	–	3.337	4.81	15.75	–	3.337	4.81	15.75	–	3.337	4.81
Bilingual	71.81	0.639	3.265	51.17	16.14	0.632	3.607	4.08	17.12	0.639	3.620	5.69	17.29	0.641	3.615	6.18
Multilingual	31.65	0.674	3.475	17.62	15.53	0.681	3.592	4.44	15.49	0.687	3.623	4.35	15.38	0.688	3.574	4.45
Bengali																
GT	24.78	–	3.062	6.48	24.78	–	3.062	6.48	24.78	–	3.062	6.48	24.78	–	3.062	6.48
Bilingual	108.12	0.657	2.990	70.10	24.23	0.633	3.546	4.95	24.20	0.634	3.590	6.66	25.72	0.636	3.595	7.59
Multilingual	46.78	0.668	3.494	21.57	26.54	0.692	3.578	7.21	26.60	0.688	3.427	7.07	26.92	0.699	3.584	7.13
Odia																
GT	23.52	–	3.131	4.51	23.52	–	3.131	4.51	23.52	–	3.131	4.51	23.52	–	3.131	4.51
Bilingual	79.53	0.676	3.399	49.29	27.51	0.646	3.513	5.83	27.83	0.649	3.531	5.83	28.29	0.653	3.542	6.16
Multilingual	43.25	0.651	3.467	18.90	26.57	0.656	3.477	5.27	26.17	0.655	3.512	5.19	26.09	0.665	3.529	5.18
Marathi																
GT	10.56	–	3.366	2.08	10.56	–	3.366	2.08	10.56	–	3.366	2.08	10.56	–	3.366	2.08
Bilingual	74.51	0.596	2.900	43.51	11.46	0.605	3.475	2.47	11.72	0.611	3.496	2.47	12.04	0.605	3.480	2.76
Multilingual	30.76	0.637	3.449	15.44	10.58	0.657	3.501	2.08	10.74	0.661	3.528	2.00	10.43	0.663	3.545	1.96

Table 3: Cross-lingual TTS evaluation (English Reference and Indic Generation) across Indian languages

2.4.3 Heuristic-2: Additive duration-per character Guided Scaling

The second heuristic extends the original character-based duration estimation by incorporating relative duration-per character differences between languages through an additive correction mechanism. The target duration is computed as:

$$D_{H2} = D_{ref} + \left(\frac{D_{ref}}{C_{ref}} + \frac{d^c_{gen}}{d^c_{ref}} \right) \frac{C_{gen}}{s} \qquad (6)$$

where D_{ref} denotes the reference speech duration, C_{ref} and C_{gen} denote the number of characters in the reference and generated text, and d^c_{ref} and d^c_{gen} denote the language-specific duration-per character statistics corresponding to the reference and target generation languages. $\frac{D_{ref}}{C_{ref}}$ captures the reference duration-per-character, while $\frac{d^c_{gen}}{d^c_{ref}}$ provides a relative duration adjustment that introduces a language-dependent correction to per-character scaling across languages.

3 Experimental Setup

3.1 Training Dataset

We follow the original A-DMA (Choi et al., 2025) training setup and adapt it for Indian languages in order to evaluate its multilingual performance. The model was trained using four multilingual speech corpora: IndicVoices-R (Sankar et al., 2024) containing 1,587.5 hours of speech from 21 languages; IndicTTS V2 and V3 versions (Consortium et al., 2023) containing 547 hours and 363.3 hours of speech, respectively, covering 14 languages in V2 and 20 languages in V3; and Rasa (Srinivasa Varadhan et al., 2024) containing 750.6 hours of speech across 16 languages. In total, the model was trained on 24 languages, comprising approximately 3,248.5 hours of multilingual speech data. Detailed per-language training hours are provided in Table 2. The trained model are evaluated using the proposed cross-lingual benchmark described in Section 2.2.

3.2 Evaluation Metrics

We evaluate synthesized speech using four objective metrics. Word Error Rate (WER) and Character Error Rate (CER) are computed from ASR tran-

scriptions generated using Whisper-large-v3 (Radford et al., 2023) for English and IndicConformer-600M (AI4Bharat, 2024) for Indian languages. Speaker Similarity (SIM) evaluates speaker identity preservation using WavLM-large-based speaker embeddings (Chen et al.). UTMOS (Saeki et al., 2022) measures speech naturalness via an objective MOS prediction model.

4 Results

The trained bilingual and multilingual A-DMA models are evaluated using the proposed benchmark under four duration strategies: Original F5-TTS, Modified F5-TTS, Heuristic-1, and Heuristic-2. We evaluate performance in two cross-lingual settings: (i) Indic Reference and English Generation and (ii) English Reference and Indic Generation. Although recent methods such as Cross-Lingual F5-TTS (Liu et al., 2026) and PFluxTTS (Pankov et al., 2026) have proposed alternative duration modeling strategies, their official implementations were not publicly available at the time of our experiments. Consequently, we compare our proposed methods against the publicly available A-DMA baseline.

4.1 English Reference and Indic Generation

Table 3 presents cross-lingual results for English reference and Indian-language speech generation. The original F5-TTS exhibits severe degradation in cross-lingual settings, primarily due to byte-level text length estimation, which leads to incorrect duration scaling and high WER/CER across all languages. For instance, in Kannada, the bilingual model yields a WER/CER of 109.56/57.07, while the multilingual model also exhibits degraded performance, with a WER/CER of 55.88/22.22.

The Modified F5-TTS substantially reduces these errors across all languages by correcting text-length estimation, resulting in large relative improvements in both WER and CER. In Kannada, the multilingual model achieves relative reductions of 52.7%/81.2% in WER/CER compared to the Original F5-TTS, indicating better linguistic alignment. Further, Heuristic-1 and Heuristic-2 for the multilingual model, provide relative WER/CER reductions of 0.9%/0.2% and 4.9%/6.2% over the Modified F5-TTS, respectively, while UTMOS increases from 3.460 to 3.475 and 3.495. Similar trends are observed across all languages.

4.2 Indic Reference and English Generation

Table 4 reports results for the reverse direction. The original F5-TTS again shows high error rates. For Kannada, the bilingual model yields a WER/CER of 57.01/52.16, while the multilingual model reduces these to 49.66/44.33, respectively.

The Modified F5-TTS significantly improves cross-lingual robustness across all languages. In Kannada, the multilingual and bilingual models achieves relative reductions of 93.2%/97.6% and 91.6%/96.1% in WER/CER over the Original F5-TTS. Further, Heuristic-1 and Heuristic-2 for the multilingual model, provide additional improvements over the Modified F5-TTS, achieving WER/CER reductions of 5.4%/6.6% and 3.0%/4.7%, respectively. SIM remains comparable, while UTMOS shows slight improvement. Similar trends are observed across all languages. Overall, all proposed duration strategies operate within a similar performance range, showing stable improvements across Indian languages.

Compared to bilingual models, multilingual models generally achieve lower WER and CER across most languages, indicating improved robustness due to exposure to diverse scripts and duration-per character variations during training.

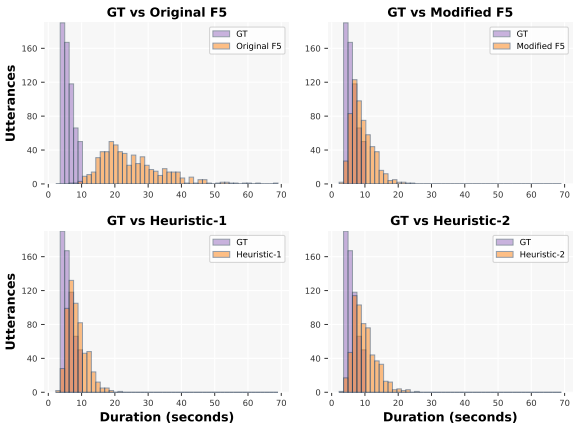


Figure 2: Comparison of ground-truth (GT) and predicted durations for English reference and Tamil speech generation.

5 Further Analysis

5.1 Qualitative Analysis

Fig. 2 and Fig. 3 show the predicted and ground-truth (GT) duration distributions on our benchmark for two cross-lingual settings: (i) English reference and Tamil speech generation and (ii) Tamil reference and English speech generation. GT durations correspond to natural multi-speaker utter-

Model	Original F5-TTS				Modified F5-TTS				Heuristic-1				Heuristic-2			
	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER
Kannada																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	57.01	0.630	3.613	52.16	4.78	0.697	3.756	2.02	4.09	0.700	3.767	1.45	4.08	0.700	3.776	1.67
Multilingual	49.66	0.609	3.573	44.33	3.36	0.707	3.710	1.06	3.18	0.708	3.704	0.99	3.26	0.708	3.689	1.01
Telugu																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	52.09	0.626	3.181	47.81	3.86	0.722	3.414	1.53	3.60	0.720	3.374	1.24	3.74	0.724	3.408	1.45
Multilingual	45.64	0.628	3.254	41.46	3.65	0.721	3.289	1.20	3.73	0.724	3.293	1.18	3.50	0.725	3.257	1.25
Tamil																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	60.46	0.583	3.564	54.95	5.66	0.685	3.750	2.66	4.11	0.689	3.797	1.64	4.33	0.694	3.783	1.62
Multilingual	56.75	0.602	3.440	50.59	4.11	0.701	3.636	1.51	3.49	0.708	3.635	1.14	3.35	0.706	3.635	1.14
Hindi																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	50.86	0.491	3.137	45.44	7.22	0.569	3.172	3.72	5.89	0.576	3.188	2.66	6.09	0.578	3.137	2.76
Multilingual	41.52	0.564	3.061	35.34	4.97	0.640	3.139	1.93	4.91	0.639	3.110	2.00	4.70	0.639	3.100	1.89
Malayalam																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	60.60	0.571	3.400	54.14	7.91	0.690	3.590	2.91	4.67	0.699	3.624	2.11	5.65	0.699	3.618	2.58
Multilingual	56.55	0.589	3.436	50.85	4.84	0.704	3.617	2.23	4.04	0.708	3.633	1.56	4.68	0.708	3.628	1.97
Punjabi																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	49.27	0.599	3.400	44.57	4.57	0.690	3.438	1.80	4.32	0.692	3.440	1.83	4.76	0.692	3.426	2.15
Multilingual	39.30	0.646	3.365	33.91	3.72	0.725	3.380	1.55	3.71	0.727	3.425	1.33	3.61	0.727	3.411	1.31
Bengali																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	52.98	0.571	3.407	48.31	4.19	0.661	3.444	1.68	4.15	0.663	3.407	1.81	4.45	0.663	3.462	2.15
Multilingual	47.22	0.595	3.324	42.60	4.14	0.687	3.325	1.54	4.19	0.687	3.315	1.51	4.06	0.687	3.282	1.42
Odia																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	50.99	0.599	3.151	45.84	5.29	0.687	3.277	2.32	5.15	0.689	3.288	2.17	5.23	0.699	3.287	2.39
Multilingual	46.57	0.606	3.243	41.51	4.17	0.691	3.317	1.50	3.93	0.708	3.634	1.44	3.92	0.695	3.314	1.44
Marathi																
GT	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33	3.54	–	4.248	1.33
Bilingual	49.27	0.566	3.382	44.98	4.35	0.676	3.423	1.71	4.19	0.681	3.431	1.69	4.44	0.680	3.419	1.68
Multilingual	42.28	0.603	3.324	37.96	3.16	0.699	3.397	1.10	3.04	0.700	3.415	0.92	3.20	0.701	3.404	0.94

Table 4: Cross-lingual TTS evaluation (Indic Reference and English Generation) across Indian languages

ances, while generated speech is conditioned on a reference utterance. Since GT and generated samples are not speaker-aligned, we compare duration distributions instead of absolute differences. The original F5-TTS underestimates durations relative to GT due to byte-level UTF-8 encoding. The modified F5-TTS alleviates this issue by switching to character-level encoding; however, it still underestimates durations, resulting in speech that sounds too fast in setting (ii). Meanwhile, the proposed Heuristic-1 and Heuristic-2 methods produce durations that align better with the GT distribution in both settings. This indicates that incorporating language-specific duration-per character statistics during inference improves cross-lingual duration modeling and yields better alignment with ground-truth.

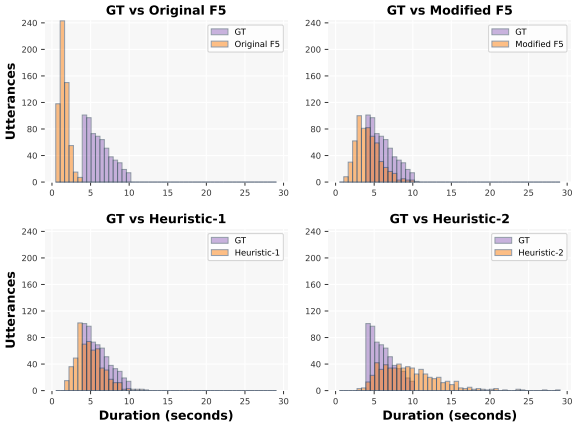


Figure 3: Comparison of ground-truth (GT) and predicted durations for Tamil reference and English speech generation.

5.2 Effect of $\alpha : \beta$ ratio in Heuristic-1

To investigate the sensitivity of Heuristic-1 to the interpolation coefficients α and β, we evaluate three ratio configurations: 6:4, 7:3, and 8:2 using the multilingual model across Telugu, Hindi, and Kannada in both cross-lingual settings. As shown in Table 5, the performance remains similar across ratio configurations, with minor improvements. For Telugu in the (i) English Reference and Indic Generation setting, the 7:3 ratio achieves the best WER/CER of 16.40/1.93, with relative improvements of 2.1%/2.5% over 6:4 and 0.2%/5.9% over 8:2, while SIM remains nearly unchanged.

In the reverse setting, (ii) Indic Reference and

$\alpha : \beta$	Ref-English and Gen-Indian				Ref-Indian and Gen-English			
	WER	SIM	UTMOS	CER	WER	SIM	UTMOS	CER
Telugu								
GT	21.06	-	3.119	3.07	3.54	-	4.248	1.33
6:4	16.75	0.661	3.243	1.98	3.21	0.725	3.379	1.03
7:3	16.40	0.662	3.508	1.93	3.73	0.724	3.293	1.18
8:2	16.43	0.662	3.243	2.05	3.49	0.723	3.379	1.21
Hindi								
GT	14.40	-	2.573	5.21	3.54	-	4.248	1.33
6:4	11.72	0.678	3.608	3.48	4.41	0.642	3.172	1.71
7:3	11.66	0.680	3.566	3.42	4.91	0.639	3.110	2.00
8:2	11.70	0.676	3.608	3.46	4.79	0.642	3.172	1.75
Kannada								
GT	33.38	-	3.461	5.88	3.54	-	4.248	1.33
6:4	25.70	0.660	3.435	4.00	3.20	0.708	3.756	0.97
7:3	26.22	0.664	3.475	4.16	3.18	0.708	3.704	0.99
8:2	26.04	0.660	3.435	4.12	3.19	0.709	3.756	1.00

Table 5: Heuristic-1 based cross-lingual TTS performance across $\alpha : \beta$ duration ratios.

English Generation, the 6:4 ratio performs slightly better in some cases. Overall, the optimal ratio varies slightly with the language pair and direction, and we **recommend experimenting with different values**. We adopt $\alpha = 0.7$ and $\beta = 0.3$ for all experiments to ensure fair comparisons.

6 Conclusion

We propose a cross-lingual evaluation benchmark and language-aware duration strategies for alignment-free TTS. Our approach introduces Modified F5 for improved duration estimation and two language-aware duration prediction methods, Heuristic-1 and Heuristic-2, which incorporate language-specific duration-per character statistics. Extensive experiments on both bilingual and multilingual models, evaluated across nine Indian languages and two cross-lingual directions, show that the proposed heuristic methods consistently improve duration alignment. Qualitative analysis further highlights the importance of language-aware duration modeling for robust multilingual speech synthesis.

7 Limitations

Our approach has a few limitations that we plan to address in future work. The proposed duration estimation relies on language-level average duration-per character rates and modifies the inference process without retraining; therefore, it does not explicitly account for speaker- or utterance-level variations in speaking style and prosody. In addition, our evaluation covers nine Indian languages and English, focusing on English to Indic and Indic to English cross-lingual directions. As future work, we plan to extend the evaluation to Indic to Indic language pairs to investigate broader cross-lingual generalization. Finally, the proposed duration adjustments are heuristic rather than learned, and our comparison is primarily with the publicly available alignment-free TTS such as A-DMA/F5-TTS baselines, as implementations of some recent cross-lingual TTS methods were unavailable during our experiments.

References

AI4Bharat. 2024. IndicConformerASR.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing. *IEEE Journal of Selected Topics in Signal Processing, 2022*, 16(6):1505–1518.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025. F5-TTS: A Fairytaler that Fakes Fluent and Faithful Speech with Flow Matching. In *Proc. 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271, Vienna, Austria. Association for Computational Linguistics.

Jeongsoo Choi, Zhikang Niu, Ji-Hoon Kim, Chunhui Wang, Joon Son Chung, and Xie Chen. 2025. Accelerating Diffusion-based Text-to-Speech Model Trainingwith Dual Modality Alignment. In *Proc. Interspeech*, pages 3459–3463.

Speech Technology Consortium, Hema A Murthy, and S Umesh. 2023. Indic TTS: A Text-to-Speech Database for Indian Languages.

Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, Yanqing Liu, Sheng Zhao, and Naoyuki Kanda. 2024. E2 TTS:

Embarrassingly Easy Fully Non-Autoregressive Zero-Shot TTS. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 682–689.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist Temporal Classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. International Conference on Machine Learning (ICML)*, page 369–376.

Qingyu Liu, Yushen Chen, Zhikang Niu, Chunhui Wang, Yunting Yang, Bowen Zhang, Jian Zhao, Pengcheng Zhu, Kai Yu, and Xie Chen. 2026. Cross-Lingual F5-TTS: Towards Language-Agnostic Voice Cloning and Speech Synthesis. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 17362–17366.

Vasista Sai Lodagala, Sreyan Ghosh, and S. Umesh. 2023. Data2vec-Aqc: Search for the Right Teaching Assistant in the Teacher-Student Training Setup. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *Proc. International Conference on Learning Representations (ICLR)*.

Vikentii Pankov, Artem Gribul, Oktai Tatanov, Vladislav Proskurov, Yuliya Korotkova, Darima Mylzenova, and Dmitrii Vypirailenko. 2026. PFluxTTS: Hybrid Flow-Matching TTS with Robust Cross-Lingual Voice Cloning and Inference-Time Model Fusion. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 17612–17616.

William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4172–4182.

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. *J. Mach. Learn. Res.*, 25(1).

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proc. International Conference on Machine Learning (ICML)*, pages 28492–28518.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge. In *Proc. Interspeech*, pages 4521–4525.

Ashwin Sankar, Srija Anand, Praveen Srinivasa Varadhan, Sherry Thomas, Mehak Singal, Shridhar Kumar, Deovrat Mehendale, Aditi Krishana, Giri Raju, and Mitesh Khapra. 2024. Indicvoices-r: Unlocking a massive multilingual multi-speaker speech corpus for scaling indian tts. *NeurIPS 2024 Datasets and Benchmarks*.

Hubert Siuzdak. 2024. Vocos: Closing the gap between time-domain and Fourier-based neural vocoders for high-quality audio synthesis. In *Proc. International Conference on Learning Representations (ICLR)*, pages 25719–25733.

Praveen Srinivasa Varadhan, Ashwin Sankar, Giri Raju, and Mitesh M Khapra. 2024. Rasa: Building Expressive Speech Synthesis Systems for Indian Languages in Low-resource Settings. In *Proc. Interspeech*, pages 1830–1834.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. 2023. ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16133–16142.

Rewrites Are Not Lessons: Teachable Edits as an Output Specification for Multimodal Language Learning

Wajdi Zaghouani
Northwestern University in Qatar
wajdi.zaghouani@northwestern.edu

Abstract

Multimodal large language models are increasingly described as tutors for language learning: they take a handwritten page, a spoken utterance, or a slide and return fluent target-language text. We argue that this description confuses a *correction engine* with a *teaching intervention*. For systems claiming to teach through corrective feedback, we propose a *teachable edit*: localized on the learner's production, typed by error category, conservative with respect to the learner's interlanguage, and accompanied either by a faithful, level-appropriate explanation or by a recorded, calibrated abstention. Fluency of a rewritten string is the wrong success criterion. Existing explainable and pedagogically targeted grammatical error correction (GEC) work already supplies evidence words, error types, and grammar-item evaluation on text. What remains underspecified is a single record that also localizes into a multimodal artifact, scores conservatism and explanation-to-edit faithfulness, and requires calibrated abstention. We illustrate the specification on Chinese and Arabic examples, using the latter to show how language-specific error categories and right-to-left script affect localization while leaving the functional structure of the record unchanged. We state that record, compare it with EXPECT, EXCGEC, and JELV, and separate correction correctness from feedback quality and from learning outcomes. The correction, localization, and reference-based estimates of conservatism can be measured on existing benchmarks; the explanation, level, and abstention fields need a small annotation pass, which we specify and do not run.

1 Introduction

Surveys of multimodal LLMs in education already describe these models as supporting "personalized learning paths" and language acquisition (Xing et al., 2024; Yin et al., 2024). The typical demonstration is straightforward. The user uploads a handwritten sentence; the model returns a fluent target-language string. This output is then presented as tutoring. It is not.

Consider a schematic page from a Chinese as Foreign Language (CFL) course whose intended meaning is "I have a teacher," written 我有一个老师 *Wǒ yǒu yí gè lǎoshī* where the task asked for the polite form 我有一位老师 *Wǒ yǒu yí wèi lǎoshī*. A correction engine emits 我有一位老师 *Wǒ yǒu yí wèi lǎoshī* and stops. A teaching move would instead point at the classifier slot on the page, name the register contrast, leave the rest of the sentence alone, and either explain that contrast or abstain. The first output can look like a tutor in a product screenshot. It is a rewrite, not a lesson.

The distinction is not specific to Chinese. A learner of Modern Standard Arabic writes الطالبة مجتهد (*al-ṭāliba mujtahid*), "the female student is diligent," where the target is الطالبة مجتهدة (*al-ṭāliba mujtahida*). A correction engine can return the corrected sentence and stop. A teachable edit points to the adjective مجتهد *mujtahid*, types the problem as gender agreement, adds the feminine ending, and explains that the predicate adjective must agree with the feminine subject الطالبة *al-ṭāliba*, leaving the rest of the sentence untouched. If the handwritten final character is visually ambiguous, the system coarsens the pointer to the whole adjective or abstains rather than presenting a recognition error as learner feedback. This Arabic example is constructed for illustration and is not drawn from a visual Arabic GEC benchmark.

Second-language writing research distinguishes direct correction, indirect marking, and metalinguistic explanation (Ellis, 2009; Ferris, 2002; Bitchener and Storch, 2016). Educational NLP has largely optimized the cleaned draft or the reference-matching edit set (Bryant et al., 2023; Ng et al., 2014; Bryant et al., 2017). Visual CGEC

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 56–64
November 9, 2026 ©2026 Association for Computational Linguistics

makes the same move on images (Wang et al., 2025).

This paper takes three linked positions.

Claim A (capability is not instruction). Showing that a model can read a page and emit a correct sentence does not show that it can teach. Generation quality and instructional validity come apart (Caines et al., 2023).

Claim B (rewrites are not lessons). Full-sentence generation obscures what the learner already knows (Selinker, 1972). A system that cannot show selective feedback fails low-proficiency learners: they cannot see which forms were already usable and which form was the lesson.

Claim C (explanation is a first-class output). The rationale is part of the intervention. A fluent wrong rule is worse than marking a span and stopping (Ellis, 2009; Fei et al., 2023).

We take VisCGEC as the main public visual CGEC benchmark built on CFL handwriting and explicitly framed for language learners (Wang et al., 2025). If that setting adopts a teachable-edit record, downstream visual GEC papers are likely to follow. The argument is stated so it can transfer to speech and slides; we do not treat those modalities as already solved.

If VisCGEC-style shared tasks were to report a teachable-edit record, downstream systems would have a reason to emit these fields; we have no evidence of uptake and claim none.

Two clarifications follow. First, standard M2/ERRANT $F_{0.5}$ is an edit-level metric against reference edits, not a raw string match (Bryant et al., 2017, 2019). Agreement with reference edits does not measure pedagogical appropriateness, explanation faithfulness, proficiency alignment, localization to a multimodal artifact, or learner uptake. Second, text-only explainable GEC already exists. Section 2 states what we add.

2 Related work: what is already specified

Explainable GEC. EXPECT annotates evidence words and grammatical error types, and evaluates whether explanations help second-language learners accept suggestions and understand rules (Fei et al., 2023). EXGEC/EXCGEC extend this for Chinese: edit-wise explanations pair each correction with an error type and free-text description. Their findings show that explaining *after* committing the correction outperforms explaining beforehand, and explicit edit extraction improves explanation faithfulness (Ye et al., 2025). Both systems implement parts of Claims B and C on tokenized text. Neither requires a pointer into a handwritten page, speech span, or slide region; neither scores conservatism as a first-class property; and neither treats calibrated abstention as a required field of one output record. The natural integration is to keep EXCGEC's edit-wise explanation framework and add localization plus abstention.

Arabic provides a complementary, text-based case. The QALB shared task established large-scale correction evaluation for Modern Standard Arabic (Mohit et al., 2014). ARETA performs automatic Arabic error type annotation over a taxonomy adapted from the Arabic Learner Corpus, designed around morphological richness and orthographic ambiguity (Belkebir and Habash, 2021). Nahw evaluates Arabic grammatical error detection, correction, and explanation jointly (Mubarak et al., 2026). These resources establish correction, error typing, and explanation as active Arabic targets. None of them provides pointers into handwritten artifacts, recognition-aware pseudo-error reporting, or the joint record proposed here. Arabic therefore tests whether the teachable-edit specification can hold a shared functional layer fixed while accommodating language-specific morphology, orthography, and script direction.

Proficiency-aligned evaluation. CTSEG evaluates GEC with CEFR-linked grammar-item minimal pairs (Koyama et al., 2025). Current models are much stronger on beginner items than on intermediate and advanced ones. Aggregate correction scores therefore hide instructional mismatch. CTSEG remains text-only and does not evaluate explanations; extending it to multimodal or explanation-aware settings would require additional annotation and design work.

Minimal edits, fluency edits, and LLM instructions. Annotation schemes already split minimal correction from fluency rewriting. JFLEG is fluency-oriented (Napoles et al., 2017); CoNLL-style and many BEA references aim at more local repairs (Ng et al., 2014; Bryant et al., 2019). MultiGEC-2025 separates a minimal-edit track from a fluency track (Masciolini et al., 2025). Recent work analyzes how instruction-following LLMs change behavior across editing

	EXPECT	EXCGEC	JELV	Ours
Edit-wise type	yes	yes	n/a	yes
Free-text expl.	limited	yes	no	yes
Edit-validity test	no	no	yes	yes
Pointer to artifact	no	no	no	yes
Scored abstention	no	no	no	yes
Split eval. targets	no	no	no	yes

Table 1: Closest prior instruments are text-only or validity-only. The specification is the joint record. "Limited" for EXPECT means evidence words plus an error type rather than a natural-language rationale. "Split eval. targets" means separate measurement of four items: correction correctness, feedback quality (pointer, type, conservatism, explanation, abstention), immediate uptake, and learning outcomes; EXPECT already includes correction and learner-facing evaluation, while EXCGEC reports separate edit and explanation measures that map to two of the four.

modes (Türker and Eryigit, 2026). Sequence-to-edit taggers already expose an edit inventory (Omelianchuk et al., 2020). JELV judges whether an individual edit is valid on grammaticality, faithfulness, and fluency, and can reclassify over-corrections that $F_{0.5}$ treats as false positives (Zhan et al., 2026). That is a ready instrument for the conservatism field. The remaining gap is that multimodal learning papers still present a fluent hypothesis as the educational object.

What we claim is new. The proposed specification adds five jointly required properties: (i) localization into the original multimodal artifact; (ii) conservatism as an evaluated property, not only a prompting style; (iii) faithfulness of the explanation to the proposed edit; (iv) calibrated abstention; (v) reporting these fields in one record, so feedback quality can be scored separately from correction correctness and from uptake. Table 1 places those properties against the closest prior systems.

3 Multimodal models are not tutors by default

A tutor locates a problem in the learner's production; chooses a feedback strategy matched to proficiency and goal (Ellis, 2009; Lyster and Ranta, 1997); leaves intact what is already successful; and supports uptake. None of these is implied by image-to-text or speech-to-text generation.

Surveys of multimodal LLMs emphasize architecture, alignment, and benchmarks (Yin et al., 2024). Surveys of MLLMs in education emphasize engagement and content generation (Xing et al., 2024). Work on LLMs for language assessment is more cautious: generation can improve while grading and error-correction validity do not automatically follow (Caines et al., 2023). Unless a paper specifies a learning construct (accuracy on a treatable form, uptake of a feedback move, reduction of a recurring L1-transfer error), it should not claim tutoring capability.

Pretraining, supervised instruction tuning, and preference optimization all change model behavior. None of those stages is a substitute for an explicit instructional objective.

VisCGEC contains 2,451 images of CFL handwriting with expert correction texts (Wang et al., 2025). It is the main public visual CGEC benchmark built on CFL handwriting, which is why we treat it as the place to set a reporting standard. Reported two-stage systems combine OCR or CLIP-based recognition with GECToR-style editors, sequence-to-sequence correctors, or fine-tuned LLMs. OCR followed by fine-tuned Qwen achieves 28.59 character-level correction $F_{0.5}$ and the highest sentence-level correction $F_{0.5}$, 15.85, among the reported baselines. That is a capability result on a hard visual task, not a learning result. A stronger future system that emitted only polished strings would still be a correction engine unless its output implemented a teaching move.

4 Rewrites are not lessons

Written corrective feedback research distinguishes direct correction, indirect marking, and metalinguistic comment (Ellis, 2009; Bitchener and Storch, 2016), and focused versus comprehensive scope (Ferris, 2002; Lee, 2019). Findings on which strategy wins are mixed (Bitchener and Knoch, 2008). The underspecified prior is that the form of the feedback is part of the treatment.

Full-sentence generation obscures what the learner already knows. The convenient object is a fluent hypothesis; the pedagogical cost is that successful interlanguage disappears into the rewrite (Selinker, 1972). Low-proficiency learners need selective feedback: one treatable form marked, the rest left standing so it can be reused (Lee, 2019; Ferris, 2002). A system that only returns a polished sentence cannot offer that view. Edit extraction can be applied after the fact (Bryant et al., 2017); the default object exposed to the learner is still the rewrite.

Over-correction and reference design. Some GEC references are local repairs; JFLEG authorizes fluency rewriting toward native-like text (Napoles et al., 2017; Bryant et al., 2023). LLM GEC often drifts toward the fluency pole unless constrained (Davis et al., 2024; Staruch et al., 2025; Türker and Eryigit, 2026). A model rewarded for matching a fluency reference will change grammatical but non-native wording. Interlanguage is evidence (Selinker, 1972), not noise.

Unfocused feedback. A CFL page may contain a lookalike character, an L1-calqued particle, and a discourse inconsistency. Comprehensive correction of every deviation in one pass is a strategy writing researchers often advise against for lower-proficiency learners (Lee, 2019). Focus is a teaching decision. Standard benchmark outputs and official metrics do not require a system to declare that decision.

Loss of the page. In visual GEC the learner's object is the image. Recognition-first pipelines reduce the page to a token sequence, so layout, visible revisions, or character formation never reach the correction stage. Wang et al. (2025) document recognition error as a major pipeline failure; they do not isolate those visual factors. An edit that cannot be found on the page is not feedback the learner can apply.

The default research target for multimodal learning tools should therefore be conservative, typed, localized edits. Conservative: change only what is erroneous relative to a stated goal. Typed: attach a category a learner or teacher can act on. Localized: point at a span, timestamp, or box on the original production. Direct correction of a treatable form, with the rest left alone, still satisfies this target.

5 Explanation is a first-class output

Ellis treats metalinguistic feedback as a distinct option (Ellis, 2009). EXPECT shows that evidence words plus error type can help learners decide whether to accept a suggestion (Fei et al., 2023). Generated rationales introduce a further failure: models invent grammatical stories that sound expert (Fei et al., 2023; Caines et al., 2023).

Consider the classifier contrast in 一位老师 *yí wèi lǎoshī* versus 一个老师 *yí gè lǎoshī*, both meaning "a teacher" in standard Chinese. Either classifier can count people, but they differ in register: the first is the polite form, while the second is grammatical in many contexts and less formal. A useful explanation names that register choice and points at the classifier slot. A harmful explanation invents a rule such as "一个老师 *yí gè lǎoshī* is impossible, because that classifier cannot be used for people." Faithfulness fails when the explanation describes a different phenomenon than the edit.

Explanation should therefore be constrained: it must describe the edit actually proposed; it must match the intended proficiency level (Koyama et al., 2025); and the system must be allowed to withhold a rule rather than fabricate one.

Field	Example	Evaluation
Pointer	box over the classifier slot on the page	box IoU sweep / span F_1; coarsening rate
Edit	个 → 位 (*context:* 我有一个老师 *I have a teacher* → 我有一位老师 *polite*)	edit P/R; pseudo-error rate
Error type	classifier / register	type accuracy (κ)
Target form	一位老师 *yí wèi lǎoshī*	match to references
Explanation	the polite classifier 位 matches the formal task requirement	see faithfulness row
Faithfulness	covers this edit / related / unrelated	3-way rating; span grounding
Factuality	correct rule / invented / abstained	3-way rating; entailment
Level	elementary band	lexical band; teacher rating
Abstention	no rule if type unknown	coverage-risk; teacher-anchored
Conservatism	no other tokens changed	unnecessary-edit rate; budget

Table 2: Teachable-edit record. The Chinese example is illustrative, not an item from VisCGEC.

6 An output specification and four evaluation targets

Table 2 states a teachable-edit record. The example is schematic: a handwritten CFL sentence intended as "I have a teacher," written with the general classifier where the task asked for the polite form.

The specification is compatible with ERRANT-style types (Bryant et al., 2017), EXPECT evidence (Fei et al., 2023), EXCGEC edit-wise descriptions (Ye et al., 2025), sequence-to-edit tags (Omelianchuk et al., 2020), and CEFR item labels

Field	Arabic instantiation
Pointer	adjective span مجتهد and its final-character boundary
Source	الطالبة مجتهد
Edit	insert feminine suffix: مجتهد → مجتهدة
Surface type	Arabic gender agreement
Functional type	form agreement
Explanation	the predicate adjective agrees with the feminine subject الطالبة
Abstention	withhold if the final handwritten character is uncertain
Conservatism	no other source word changed

Table 3: Arabic instantiation of the teachable-edit record. The example is constructed to illustrate Modern Standard Arabic gender agreement; it is not drawn from a visual Arabic GEC benchmark.

(Koyama et al., 2025). The missing piece is refusing to collapse the record into one fluent hypothesis before scoring.

6.1 Operational measures

The record can be started without new data, but not finished without it: the correction and localization fields can be retrofitted onto existing benchmarks. Conservatism can be evaluated using reference and validity judgments available now; teacher-validated pedagogical conservatism (determining whether the edit aligns with lesson goals for that learner) requires additional annotation. The explanation, level, and abstention fields need a small annotation pass, which Section 8 specifies and does not run.

Conservatism. Report three quantities because they fail differently. Edit rate is descriptive: a high rate on a heavily erroneous page is correct behavior, not overcorrection. Unnecessary-edit rate is the share of edits that neither match a reference nor survive validity checking; only genuine overcorrections count against conservatism. Budget compliance is the share of records respecting a declared feedback budget: the number of edits the system will surface for this learner at this level.

Faithfulness and factuality. Three checks replace proxy overlap with gold rationales, which correlates poorly with human usefulness (Schuff et al., 2022). *Span grounding*: every form the explanation cites must lie inside the pointer span or declared evidence tokens. *Entailment*: the explanation must justify the proposed edit using the source context, target form, and declared task goal. *Counterfactual sensitivity*: changing the target to instantiate a different linguistic phenomenon should change the corresponding explanation.

Abstention. Score abstention as selective prediction: a coverage-risk curve over edit-level confidence, the area under it, and calibration error (Guo et al., 2017). Two reporting points matter pedagogically. *Teacher-anchored* selective risk takes the error indicator from teacher judgment rather than reference match. *Abstention appropriateness* is the share of withheld cases a teacher agrees should have been withheld.

Types and levels. The type field carries two layers because no single scheme is both language-faithful and portable. The surface layer is language-specific and adopts whatever is established (ERRANT for space-delimited orthographies, character-level annotation for Chinese). The functional layer is a small closed set: form agreement, argument structure, selection and collocation, orthography and character formation, cohesion, and pragmatic appropriateness. Only the functional layer is claimed to transfer. Proficiency labels stay inside one framework per item (HSK or CEFR), not forced mappings, and bind the explanation on two computable axes: the share of explanation tokens inside the level's vocabulary, and the count of grammatical terms outside its declared metalanguage. Inter-rater agreement on types is measured with κ (Cohen, 1960).

6.2 Localization under recognition noise

The pointer is the specification's load-bearing multimodal commitment and the field most exposed to recognition error. Where a transcript exists and is aligned to the artifact, report span F_1; where the pointer is a region, report intersection over union against annotated boxes at several thresholds.

Uncertainty has to propagate rather than vanish. Distinguish recognition confidence (the recognizer's confidence over the spanned units) from geometric localization confidence (reliability of the region coordinates themselves). The specification imposes a two-stage gate applied in order: if recognition confidence falls below a declared threshold, the system abstains entirely. Otherwise, if fine-grained geometric localization confidence falls below its threshold, the system coarsens the pointer to the enclosing unit and records that change.

Arabic exposes a localization problem that Chinese hides. Arabic text is stored in Unicode logical order but displayed right to left, while an artifact

pointer is geometric image coordinates. A record must therefore keep text-span order and image-coordinate order distinct, and state the script direction explicitly. For Arabic handwriting a word-level region is often more reliable than independent character boxes.

Recognition failure also manufactures errors the learner did not make. These belong in a distinct pseudo-error rate rather than absorbed into precision. Spoken GEC already treats removal of transcription-induced edits as a precondition for usable feedback (Qian et al., 2025); the visual case has the same structure.

6.3 Retrofitting the specification

Nothing in the record requires new architecture. A team with a published visual CGEC pipeline can emit teachable-edit records by building on existing components: retain recognizer confidences and box coordinates; keep the corrector, but align source and hypothesis afterward to recover an explicit edit set; filter that set with a validity judge; explain each surviving edit; finally, project each edit back onto coordinates and apply the gate. Existing components support several steps, but their integration, multimodal projection, and calibration require additional engineering. The barrier to teachable edits is partly engineering, partly a reporting convention.

7 The same specification on speech

Writing and speech are both classroom artifacts. The specification does not change across modalities; the pointer does.

Speech. A learner says an English past-tense form as a bare stem. A correction engine returns a fluent transcript with the *-ed* supplied and no timestamp. A teachable edit marks the span, types the error as tense, leaves the rest of the utterance, and either names the regular past or abstains when ASR confidence is low. Spoken GEC already treats removal of transcription-induced edits as a precondition for usable feedback, which is the pseudo-error requirement under another name.

Cross-script transfer. The Chinese and Arabic examples share the record structure but not the surface analysis. Chinese permits fairly direct character-box localization; connected Arabic handwriting may require word-level regions and boundary anchors. These differences are not noise for a universal taxonomy to remove. They live in the language-specific surface layer while the record continues to require a pointer, an edit, a type, an explanation or an abstention, a declared level, and a conservatism decision.

Shared failure. In both modalities the attractive demo is end-to-end fluency. In both the missing object is a pointer the learner can use without rereading a polished string.

Four evaluation targets should be reported separately.

Correction correctness. Does the proposed edit match a reference repair? M2/ERRANT $F_{0.5}$ is appropriate for this target. It is not a feedback metric.

Feedback quality. Are the pointer, type, conservatism, explanation, and abstention appropriate for the stated learner and goal? EXPECT's learner study is a text-only instance of this target. Multimodal localization and explanation-to-edit faithfulness still need protocols.

Immediate uptake. Does the learner use the information on a later attempt in the same session (Lyster and Ranta, 1997)?

Learning outcomes. Does the learner retain or transfer the form to new contexts? Immediate uptake is not a proxy for delayed learning or transfer.

We do not claim that $F_{0.5}$ should disappear, or that every deployment must be indirect. Direct correction of a treatable form, localized and explained, satisfies the specification. VisCGEC already annotates bounding boxes, correction texts, and error types, so three fields of Table 2 can be scored on existing data. That is why we frame the record as an extension of current practice rather than a departure from it.

8 What M2/ERRANT $F_{0.5}$ cannot see

M2/ERRANT $F_{0.5}$ rewards agreement with reference edits and weights precision over recall. That design is right for a correction engine. It is the wrong primary metric for a claim about teaching.

The reference defines the lesson. Fluency-oriented datasets such as JFLEG normally use GLEU; they authorize rewording unrelated to any local error. Our argument applies if fluency references are evaluated using edit-based $F_{0.5}$. A system can raise $F_{0.5}$ by matching those rewrites and still hide the learner's interlanguage. Minimal-edit references constrain this less, but they still do

not record whether the edit was the right teaching move.

$F_{0.5}$ is also silent on explanation. An invented rule attached to a correct edit can score perfectly and still be harmful. Faithfulness and factuality are invisible to the metric. So is the pointer into the page: in visual CGEC the learner must find the intervention on an image. A correct string with a wrong box is a failed lesson and a successful $F_{0.5}$ item.

The metric is not misspecified for correction correctness. It cannot see pedagogical appropriateness, explanation quality, proficiency alignment, multimodal localization, or uptake. Papers that report only $F_{0.5}$ and then claim educational use are changing the target after the measurement.

VisCGEC is the main public visual CGEC benchmark built on CFL handwriting. Its published metric is character- and sentence-level $F_{0.5}$, but its annotation already carries boxes, corrections, and types, so localization can be scored without new data. The remaining distance to the full specification is explanation, level, conservatism, and abstention.

A checklist for workshop chairs. A workshop that wants this standard in the next cycle could (i) add feedback-quality and, where possible, uptake columns to the leaderboard without changing $F_{0.5}$ rankings; (ii) encourage system papers to report the record when they claim educational use; and (iii) reserve a short session for systems that score high $F_{0.5}$ but low explanation faithfulness.

A protocol we are not running here. A feasibility study needs only a few dozen VisCGEC-style pages, double-annotated by trained instructors: inter-rater agreement for error type, Krippendorff's α for ordinal faithfulness ratings (Krippendorff, 2018), and mean pairwise IoU for the pointer. We state in advance what would count against us: $\alpha < 0.67$ on faithfulness should be read as evidence that the rubric needs revision, not the annotators. In the proposed annotation pass, instructors rate factuality (correct rule, invented, abstained), level appropriateness (matches declared proficiency band), and abstention appropriateness (teacher agrees withheld cases should have been withheld).

9 Conclusion

Multimodal models can see a learner's page and hear a learner's voice. That is not the same as teaching. The Chinese and Arabic cases show why the record holds its pedagogical function fixed while leaving the linguistic analysis language-specific. For systems claiming to teach through corrective feedback, their output should be a teachable edit: localized into the original artifact, typed, conservative, explained or withheld, and scored as feedback rather than only as reference-matching correction.

10 Limitations

This is a position paper. We present no new experiments, annotation, or evidence that teachable-edit records improve uptake relative to rewrites. The WCF literature is mixed on direct versus indirect and on focused versus comprehensive correction. The running example is visual CGEC for CFL because that setting publishes multimodal GEC numbers; we do not empirically show transfer to speech, slides, or other scripts. The classifier example is constructed, not from VisCGEC. We do not analyze commercial product copy. Teacher-in-the-loop may be unavailable where automation is most tempting. Our Arabic discussion is illustrative and text-grounded: we identify no Arabic visual GEC resource with annotations equivalent to VisCGEC's character boxes, and we evaluate neither handwritten Arabic recognition nor Arabic localization. Connected writing, ligatures, optional diacritics, and variation between Modern Standard Arabic and regional varieties may require revisions to the pointer and type representations.

11 Ethical Considerations

A false metalinguistic explanation is educational harm. Systems offered to learners should abstain when they cannot ground a rule in the proposed edit (Caines et al., 2023; Fei et al., 2023).

Visual recognition and page-based tutors are sensitive to image quality and writing variation. Large gaps appear between OCR and weaker recognizers. Inclusive multimodal learning papers should report failure modes by input quality rather than only mean $F_{0.5}$.

The level tag can be repurposed for placement, streaming, or assessment. We intend it as a property of the feedback, not of the learner.

A homework photograph can identify a hand, a school, and a proficiency trajectory. Localization makes the image necessary; that necessity raises the duty of care. Teacher-in-the-loop review matches current published visual CGEC quality and this risk profile better than fully automatic page correction.

12 Declaration on Generative AI

A generative AI assistant was used for language editing and drafting support based on outlines and content specified by the author. The author developed the paper's claims, proposed record, evaluation framework, and substantive arguments. No data, annotations, experiments, or measurements were produced using generative AI. The author reviewed and verified all AI-assisted material and takes responsibility for the final manuscript.

References

Riadh Belkebir and Nizar Habash. 2021. Automatic error type annotation for Arabic. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 596–606, Online. Association for Computational Linguistics.

John Bitchener and Ute Knoch. 2008. The value of written corrective feedback for migrant and international students. *Language Teaching Research*, 12(3):409–431.

John Bitchener and Neomy Storch. 2016. *Written Corrective Feedback for L2 Development*. Multilingual Matters.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical error correction: A survey of the state of the art. *Computational Linguistics*, 49(3):643–701. MIT Press, Cambridge, MA.

Andrew Caines, Luca Benedetto, Shiva Taslimipoor, Christopher Davis, Yuan Gao, Øistein Andersen, Zheng Yuan, Mark Elliott, Russell Moore, Christopher Bryant, Marek Rei, Helen Yannakoudakis, Andrew Mullooly, Diane Nicholls, and Paula Buttery. 2023. On the application of large language models for language teaching and assessment technology. In *Proceedings of the Workshop on Empowering Education with LLMs – the Next-Gen Interface and Content Generation*, volume 3487 of *CEUR Workshop Proceedings*, pages 173–197, Tokyo, Japan. CEUR-WS.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Christopher Davis, Andrew Caines, Øistein E. Andersen, Shiva Taslimipoor, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. Prompting open-source and commercial language models for grammatical error correction of English learner text. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11952–11967, Bangkok, Thailand. Association for Computational Linguistics.

Rod Ellis. 2009. A typology of written corrective feedback types. *ELT Journal*, 63(2):97–107.

Yuejiao Fei, Leyang Cui, Sen Yang, Wai Lam, Zhenzhong Lan, and Shuming Shi. 2023. Enhancing grammatical error correction systems with explanations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7489–7501, Toronto, Canada. Association for Computational Linguistics.

Dana R. Ferris. 2002. *Treatment of Error in Second Language Student Writing*. University of Michigan Press, Ann Arbor.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Aomi Koyama, Masato Mita, Su-Youn Yoon, Yasufumi Takama, and Mamoru Komachi. 2025. Targeted syntactic evaluation for grammatical error correction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21108–21125, Vienna, Austria. Association for Computational Linguistics.

Klaus Krippendorff. 2018. *Content Analysis: An Introduction to Its Methodology*, 4th ed. SAGE Publications, Thousand Oaks, CA.

Icy Lee. 2019. Teacher written corrective feedback: Less is more. *Language Teaching*, 52(4):524–536.

Roy Lyster and Leila Ranta. 1997. Corrective feedback and learner uptake: Negotiation of form in communicative classrooms. *Studies in Second Language Acquisition*, 19(1):37–66.

Arianna Masciolini, Andrew Caines, Orphée De Clercq, Joni Kruijsbergen, Murathan Kurfalı, Ricardo Muñoz Sánchez, Elena Volodina, and Robert Östling. 2025. The MultiGEC-2025 shared task on multilingual grammatical error correction at NLP4CALL. In *Proceedings of the 14th Workshop on Natural Language Processing for Computer Assisted Language Learning*, pages 1–33, Tallinn, Estonia. University of Tartu Library.

Behrang Mohit, Alla Rozovskaya, Nizar Habash, Wajdi Zaghouani, and Ossama Obeid. 2014. The first QALB shared task on automatic text correction for Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 39–47, Doha, Qatar. Association for Computational Linguistics.

Hamdy Mubarak, Majd Hawasly, and Abubakr Mohamed. 2026. Nahw: A comprehensive benchmark of Arabic grammar understanding, error detection, correction, and explanation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6310–6328, Rabat, Morocco. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR: Grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA. Association for Computational Linguistics.

Mengjie Qian, Rao Ma, Stefano Bannò, Mark J. F. Gales, and Kate M. Knill. 2025. End-to-end spoken grammatical error correction. *arXiv preprint arXiv:2506.18532*.

Hendrik Schuff, Heike Adel, Peng Qi, and Ngoc Thang Vu. 2022. Challenges in explanation quality evaluation. *arXiv preprint arXiv:2210.07126*.

Larry Selinker. 1972. Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1-4):209–232.

Ryszard Staruch, Filip Graliński, and Daniel Dzienisiewicz. 2025. Adapting LLMs for minimal-edit grammatical error correction. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 118–128, Vienna, Austria. Association for Computational Linguistics.

Tolgahan Türker and Gülşen Eryigit. 2026. Instruction-following LLMs for grammatical error correction: Analyzing neutral-anchored instructional sensitivity across editing modes. In *Proceedings of the 21st Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2026)*, pages 234–247, San Diego, California, USA. Association for Computational Linguistics.

Xiaoman Wang, Dan Yuan, Xin Liu, Yike Zhao, Xiaoxiao Zhang, Xizhi Chen, and Yunshi Lan. 2025. VisCGEC: Benchmarking the visual Chinese grammatical error correction. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5054–5068, Albuquerque, New Mexico. Association for Computational Linguistics.

Weicheng Xing, Tianqing Zhu, Jenny Wang, and Bo Liu. 2024. A survey on MLLMs in education: Application and future directions. *Future Internet*, 16(12):467.

Jingheng Ye, Shang Qin, Yinghui Li, Xuxin Cheng, Libo Qin, Hai-Tao Zheng, Ying Shen, Peng Xing, Zishan Xu, Guo Cheng, and Wenhao Jiang. 2025. EXCGEC: A benchmark for edit-wise explainable Chinese grammatical error correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25678–25686.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. A survey on multimodal large language models. *National Science Review*, 11(12):nwae403.

Yuhao Zhan, Yuqing Zhang, Jing Yuan, Qixiang Ma, Zhiqi Yang, Yu Gu, Zemin Liu, and Fei Wu. 2026. JELV: A judge of edit-level validity for evaluation and automated reference expansion in grammatical error correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 34611–34619.

A Multi-Modal Interactive Narrative Game for French-Speaking Primary School Learners

Qiao Wang[1], **Adnan Labib**[2], **Thierry Geoffre**[3], **Yixuan Huang**[4], **Zheng Yuan**[5]
[1]Hosei University, Japan
[2]King's College London, United Kingdom
[3]University of Luxembourg, Luxembourg
[4]Waseda University, Japan
[5]University of Sheffield, United Kingdom

Abstract

In this study, we built a Large Language Model (LLM)-powered interactive narrative game for French-speaking primary school learners. The system consists mainly of three modules: an initialization module for story structuring, a generation module for iterative generation of story texts and illustrations based on player choices, and a language scaffolding module for providing contextualized explanations of player-selected textual elements. Text generation is prompted using five curriculum-informed proficiency bands spanning primary to early secondary reading levels. A classroom feasibility study was conducted with 35 students aged 9-10 and two teachers at two primary schools in French-speaking Switzerland. Results showed positive student attitudes towards the stories and illustrations and high perceived ease of use. However, students perceived the game's usefulness for French literacy activities as moderate, with many noting that the story language was below their level. Occasional issues in character consistency and repetitive illustrations across scenes were also mentioned. Teachers highlighted potential pedagogical benefits when the system is combined with additional classroom activities.

1 Introduction

Interactive narrative is a form of game-like storytelling in which narratives unfold dynamically in response to player choices, allowing readers to influence plot progression rather than follow a fixed sequence of events. From *Choose Your Own Adventure*™ books to open-world video games, branching narratives are a central element of interactive fiction that enable readers to change the outcome of a narrative (Walton, 2019; Huang et al., 2024). Such narratives provide increasing engagement and agency by transforming reading into an active, decision-driven process. However, the content of traditional interactive narratives remains limited by creators' perspectives and the scope of pre-authored material (Tian et al., 2024).

The emergence of Large Language Models (LLMs) has renewed interest in interactive narrative by enabling dynamic text generation at scale. Rather than requiring authors to pre-write extensive branching scripts, LLMs can generate story continuations on demand, accommodating player input within narrative constraints defined by the author (Huang et al., 2024; Wang et al., 2025). This shift has substantially reduced authorial overhead and enabled interactive storytelling systems that respond flexibly to user actions. Such LLM-powered interactive narratives also offer pedagogical potential for young readers by creating engaging, adaptive reading experiences. However, existing systems are often English-centric, primarily text-based, not designed for younger users, and lack explicit mechanisms for conditioning linguistic difficulty (e.g., Wu et al., 2024; Xu et al., 2025; Huang et al., 2024; Wang et al., 2026). This creates a gap for systems that can generate age-appropriate reading experiences at different intended difficulty levels.

To address such limitations, we built an LLM-powered multi-modal interactive narrative game for French-speaking primary school learners. The novelty of the system lies in combining: (1) curriculum-informed proficiency conditioning for French story generation, (2) dynamically generated illustrations accompanying story texts, and (3) an in-context language assistant that explains text elements upon player query. The source code is released under the CC BY 4.0 license and is available at `https://github.com/judywq/interactive-narrative-game`. The web application uses proprietary LLMs, whose weights and use are subject to the respective providers' terms of service.

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 65–76
November 9, 2026 ©2026 Association for Computational Linguistics

2 Related Work

Recent years have seen growing interest in the use of LLMs for interactive storytelling, driven by their ability to generate fluent narrative text in response to user input. Existing systems differ in how they structure narrative control, integrate player agency, and manage long-term coherence.

One prominent line of work generates interactive narratives by branching from an existing linear story. WHAT-IF (Huang et al., 2024) introduces a meta-prompting framework that constructs alternative narrative trajectories grounded in a pre-written source text, producing "what-if" timelines that explore counterfactual developments of an original plot. By anchoring generation to a fixed narrative backbone, such systems achieve strong thematic consistency and enable systematic exploration of branching structures, but presuppose the availability of high-quality source stories. Related approaches similarly focus on expanding latent or un-narrated branches within existing story worlds.

A second group of systems emphasizes authorial control through structured narrative representations. Dramamancer (Wang et al., 2026) exemplifies this approach by allowing authors to define story schemas, including characters, events, and constraints, that are then instantiated dynamically during interactive playthroughs using an LLM. Other work explores hybrid approaches that integrate LLM-based generation with concepts from classical AI planning. Though earlier planning-based story generation systems ensured long-term coherence through explicit plot graphs but lacked natural language generation (Meehan, 1977; Young et al., 2013), more recent systems combine LLMs with structured representations such as quest trees or plot graphs to generate interactive fiction worlds, scenes, or branching quests (Ammanabrolu et al., 2019; de Lima et al., 2021; Kumaran et al., 2023; Harmon and Rutman, 2023).

Evaluation of these systems has revealed limitations, particularly with respect to maintaining meaningful agency and narrative coherence over extended interactions (Yi et al., 2025). To address these limitations, researchers have turned to memory-augmented approaches by introducing persistent representations of characters, events, and world states that can be retrieved and updated across interactions, enabling more consistent long-term storytelling (Yi et al., 2025; Nonaka and Perry, 2025). On the other hand, empirical evaluations also reveal systematic tendencies toward positive, low-conflict plot developments and reduced diversity, novelty, and surprise compared to human-authored narratives (Tian et al., 2024; Ismayilzada et al., 2024).

Beyond coherence and engagement, existing LLM-based interactive storytelling systems are limited in their broader applicability. Most systems remain predominantly text-based (Walton, 2019; Huang et al., 2024), offering limited multi-modal support that could aid comprehension. Moreover, prior work has focused largely on English narratives (Huang et al., 2024) and has typically targeted adolescent or adult users (Walton, 2019), implicitly assuming higher reading proficiency, cognitive maturity and digital literacy. For instance, OpenTheater by Xu et al. (2025) requires players to configure detailed story settings and character profiles, and to interact through typed chat to drive the narrative forward. Finally, these systems are generally not designed around pedagogical objectives and lack explicit mechanisms for aligning generated language with player proficiency levels or instructional goals (Wang et al., 2025), which limits their suitability for educational contexts.

In light of these limitations, we set out to develop a multi-modal interactive narrative game for young readers in a non-English linguistic context. We focused on children because interactive storytelling plays an important role in their development, while existing AI-based storytelling and reading technologies still often fall short of parents' expectations (Sun et al., 2024). We selected French as the target language because it is widely taught and is comparatively well supported by LLMs as a high-resource language (Li et al., 2025).

3 System Design and Development

In designing the game content, we prioritized two factors: pedagogical relevance and multi-modal story generation. Pedagogically, we defined the story genre, conditioned generation on intended reading levels, and incorporated language scaffolding. For multi-modal story generation, we incorporated illustrations alongside the story text based on the cognitive theory of multimedia learning (Mayer, 2024; Paivio, 1979), which suggests that combining verbal information (e.g., text) with nonverbal information (e.g., visuals) can support encoding and recall. In addition, given the digital literacy of primary schoolers, we designed player inter-

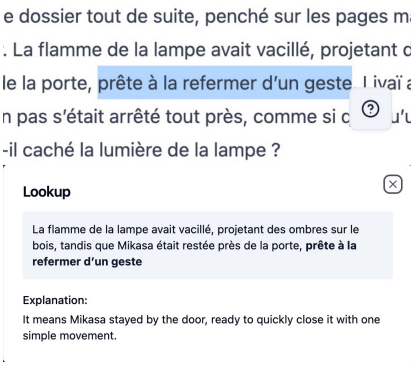


Figure 1: Language query function

action around structured choices rather than free-response chat. For system architecture, we adopted a modular multi-pipeline design (Parnas, 1972) and deployed the system as a web application with a Vue.js frontend and a Django backend.

3.1 Pedagogical Features

Genre selection We focused story generation around the fable genre. Fables are prominent in primary curricula due to their accessible narrative structure, conventional character roles, and explicit moral framing, making them ideal for reading comprehension to young learners. We defined a schema of essential structural elements in fables, which can be found in Appendix A.

Proficiency conditioning We designed five curriculum-informed proficiency bands through consultation with a certified French language teacher. For each band, we curated a dataset of fable texts that were proofread and annotated according to specific curriculum goals. The bands and sample fable texts are used as few-shot examples to condition generation toward the selected reading level (see Appendix B).

Language scaffolding We added a text query feature for players to request explanations of unfamiliar textual elements ranging from single words and phrases to sentences. A player may select the relevant text and click the question mark that subsequently appears. The system then provides a contextualized explanation in French or English (see Figure 1). At the end of a story, queried items are used to generate a recall quiz. The LLM assigns each response a score of 0, 0.5, or 1, and the system calculates a normalized score by dividing the sum of item scores by the number of questions attempted.

3.2 Multi-Modal Story Generation

For story text generation, we selected GPT-5.2[1] as the base LLM, the most advanced model from the GPT series when the study took place. For the structuring of branching narratives, we drew experience from famous commercial interactive narratives such as Detroit: Become Human[2] and Heavy Rain[3]. We base the story generation on a story skeleton, which includes main characters, multiple milestone events and a fixed number of possible endings. The story development to each milestone event progresses in multiple scenes, where players are asked to make decisions towards the end to decide what will happen in the next scene. Each milestone event remains reachable regardless of player choices.

For generating illustrations accompanying each scene, we integrated Gemini-2.5-Flash[4], which excelled in both the quality of images and speed of generation at the time of the study. At each scene, the system passes three types of contextual information to the image generation model: (1) the current scene description, (2) a summary of the story up to the current scene, and (3) the illustration from the previous scene as a reference.

3.3 System Architecture

The system consists of an initialization module, a generation module and a language scaffolding module, as shown in Figure 2. In the initialization module, when a player begins a new game by providing the necessary input, the "Proficiency Benchmarker" pipeline generates five text samples of the same narrative content using the five proficiency-band prompts described in Section 3.1. Figure 5 in Appendix B shows examples of texts at the five bands. The player selects a comfortable level, which is then used as a prompt condition for subsequent text generation. At the same time, the "Story Skeleton Generator" pipeline creates the complete narrative structure (characters, milestones, decision points, and possible endings) based on player input following the fable conventions defined in the LLM prompt.

In the generation module, the "Plot Writer" pipeline iteratively generates narrative scenes based on the story skeleton, selected proficiency level,

[1] https://platform.openai.com/docs/models/gpt-5.2
[2] https://www.quanticdream.com/en/detroit-become-human
[3] https://www.quanticdream.com/en/heavy-rain
[4] https://ai.google.dev/gemini-api/docs/models#gemini-2.5-flash

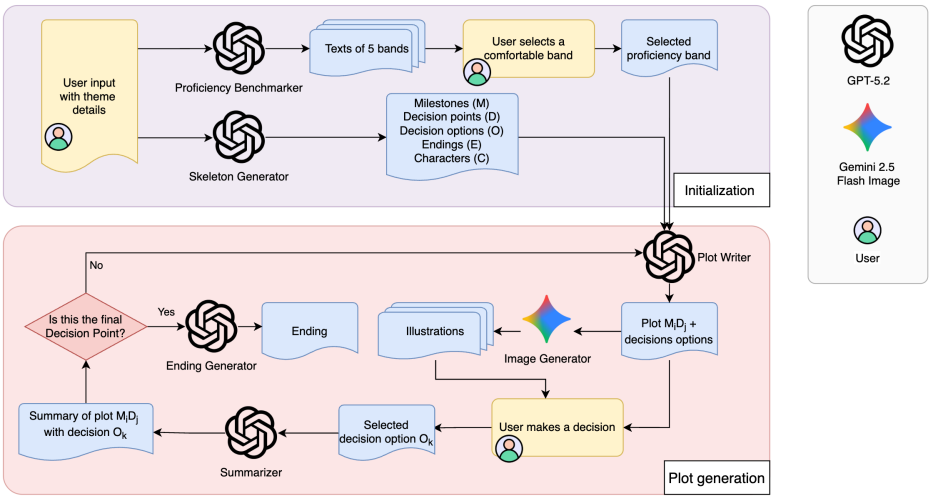


Figure 2: System Architecture

and summary of previous scenes. Subsequently, the "Image Generator" pipeline produces an illustration for each scene using the input mentioned in Section 3.2. After the player makes a decision, the "Story Summarizer" pipeline condenses the scene and choice into a running summary, to be used for next scene generation. When the final decision point is reached, the "Ending Generator" pipeline generates a concluding scene with one of the possible endings.

Independent of story generation, the language-scaffolding module logs each selected textual element and its explanation. If at least one query has been made, the module generates an end-of-story quiz from the queried items and returns the automated normalized score described above.

4 System Validation

We evaluated the system through a user study with 35 primary school students aged 9-10 and two teachers in two intact classes, one at each of two schools in French-speaking Switzerland. The curriculum of the two classes focused on French reading comprehension, which aligned directly with the game's features.

4.1 Participants

The student participants were in their 4th year of primary school, corresponding to level 3 in the five proficiency bands. They were either native French speakers or had been enrolled in French-medium schools since the beginning of their schooling. The study therefore evaluated the system in a French literacy context rather than as a second-language intervention. The teacher participants were experienced teachers with more than 15 years in primary school. Both the teachers and the students are accustomed to utilizing digital tools for learning.

4.2 Experimental Procedures

The validation study lasted for one 50-minute class session in each class. Before the validation study, the researchers provided guidance to the teachers on how to play the game and how to check gameplay logs. At the beginning of each class session, the teacher paired the student participants to reduce the number of simultaneous live-generation requests, distributed shared game accounts, introduced the gameplay mechanics, and asked each pair to test the game by playing a random story. Specifically, the teachers encouraged students to use the customizable "theme" and "scene details" fields to personalize the fable's narrative context, such as "friendship" as the theme and specific locations or characters in the scene details. An example is shown in Figure 3, where a pair requested Luffy, a character from the anime *One Piece*, in the story. Following the initial test, each pair completed one fable game, and the students subsequently completed the player survey. After each session, the corresponding teacher completed the teacher survey based on observations of the class.

4.3 Survey Instruments

The player survey (see Appendix C) was designed based on the Technology Acceptance Model (TAM) (Davis, 1989), focusing on two constructs: Perceived Usefulness (PU) and Perceived Ease of Use



Figure 3: A scene with Luffy from *One Piece*

(PEOU) of the game from the student participants' perspective. Considering the cognitive ability of the students, only three items scaled 1-7 were included under each construct. Open-ended questions were added to allow student participants to freely express their opinions towards the game. The teacher survey (see Appendix D) consisted of five open-ended questions to assess the game's pedagogical integration and its alignment with curriculum standards from the teachers' perspective.

4.4 Data Collection and Analysis

The data collected in the validation study included individual student responses to the player survey, responses from the two teachers, and gameplay logs from the system backend. Because students played in pairs using shared accounts, gameplay logs were analyzed at the pair/account level and cannot be attributed to individual students. For Likert-scale items in the player survey, internal reliability was assessed using Cronbach's alpha for the PU and PEOU scales, followed by descriptive statistics for the individual items. Responses to the open-ended questions in the player and teacher surveys were analyzed using a thematic coding approach by two coders. The gameplay logs were used to complement the survey analysis. For the optional language-scaffolding activity, query records were aggregated by shared account. For each account that completed the follow-up vocabulary quiz, we recorded the number of queries/questions and the automated normalized score produced by the system. Because only six accounts completed the quiz, we report the number of participating accounts, total and per-account query counts, and the mean, median, and range of normalized scores. No inferential tests were conducted for these data.

5 Result

5.1 Player Survey

Likert-scale items Reliability analysis of the two constructs showed good internal consistency for the PU scale (Cronbach's $\alpha = 0.77$) and marginal internal consistency for the three-item PEOU scale (Cronbach's $\alpha = 0.69$; Nunnally and Bernstein, 1994). We therefore interpret PEOU primarily through the individual item statistics. Descriptive statistics shown in Table 1 indicated consistently positive ratings on the PEOU items, with mean scores ranging from 5.66 to 5.80, suggesting that students generally perceived the game as easy to learn, clear to interact with, and easy to understand.

In contrast, the perceived usefulness (PU) items received more moderate ratings, with mean scores ranging from 4.09 to 4.60 and greater variability across participants, indicating more diverse perceptions regarding the learning value of the game.

Open-ended questions Thematic analysis of student comments revealed three important themes. First, regarding the stories, most students expressed positive attitudes (N=24), while several students expressed negative views (N=3), and a small number mentioned technical issues (N=2). One respondent comment stated "only the beginning of a story", indicating network issues in generating scenes after the story was initialized. This comment was consistent with the two observed game freezes during live generation.

Second, regarding the illustrations, the students generally liked the illustrations (N=23), with a small number of neutral responses (N=5). Individual expressions included "beautiful", "pretty", "cool", and "too good". Figure 6 in Appendix E shows a student's drawing of their favorite illustration in the player survey. Meanwhile, three students pointed out occasional inconsistent characters in illustrations (e.g., "one time I asked for a hare it gave me a goat"). Another student also commented that the illustrations looked too similar between scenes. An example is shown in Figure 4. The reason is that most of the story events took place in the same location and thus only minor elements changed between scenes.

Third, regarding language difficulty, 14 students described the game as "easy" or "too easy", while 9 reported fair difficulty. An examination of gameplay logs revealed that the majority of gameplay occurred at level 2, although the participants' expected curriculum band was level 3. This pattern indicates a preference for a lower level of challenge in the self-selection interface.

Construct	Item	Statement	Mean	SD
PU	PU1	Using the game in my French class may enable me to learn to better read and understand texts.	4.60	1.75
PU	PU2	Using the game may improve my French performance.	4.09	1.79
PU	PU3	I find the game useful in learning French.	4.11	2.08
PEOU	PEOU1	Learning to play the game was easy for me.	5.77	1.57
PEOU	PEOU2	My interaction with the game was clear and understandable.	5.66	1.53
PEOU	PEOU3	I found the game clear and understandable.	5.80	1.64

Table 1: Descriptive statistics for the TAM questionnaire items (N = 35).



Figure 4: Similar illustrations between scenes

Pair/account ID	No. of queries/ questions	Automated normalized score
1	4	0.875
2	4	0.875
6	3	0.67
7	1	0.50
9	2	0.50
24	3	0.67

Table 2: Language-query use and automated normalized vocabulary quiz scores for shared accounts that completed the quiz ($n = 6$).

5.2 Vocabulary Quiz Results

Six shared accounts used the language-query function to look up unfamiliar words or phrases and completed the corresponding vocabulary quiz. As shown in Table 2, these accounts made 17 queries in total, with between one and four queries per account. Their automated normalized quiz scores ranged from 0.50 to 0.875 ($M = 0.682$, median $= 0.67$).

5.3 Teacher Survey

The thematic analysis of the teacher survey indicates that the stories are coherent and pedagogically beneficial. The teachers confirmed that the stories maintain strong logical consistency, characterized by clear "cause-consequence" chains. For example, a character's failure is directly linked to specific choices, such as refusing help or acting out of pride. Pedagogically, a notable result is the system's role as an instructional "trigger." One teacher identified the choice-based format as ideal for practicing "anticipation" (predicting outcomes) and "textual justification" (finding evidence for character feelings). Both teachers mentioned that the game text can also be viewed as a "base draft" for writing workshops, where students can move from the digital environment to paper-based activities, such as writing alternative endings or transforming narrative into dialogue. While one teacher described the system's literary style as "flat" and "repetitive", this was framed as a pedagogical benefit as well. For "struggling" readers, the predictable syntax acts as a supportive scaffold. For advanced students, "text improvement" activities may be implemented where pupils may enhance the game's "flat" style with more vivid descriptions and humor.

6 Discussion

Overall, the validation study suggests that the game was well received by the students and teachers, particularly due to its engaging stories and illustrations. One limitation identified by students was the occasional inconsistency and repetitiveness of the generated illustrations. At the time this study was conducted, maintaining character consistency across sequential images remained a well-recognized limitation of visual models (Avrahami et al., 2024; Shen et al., 2025). Although reference images were used to improve character consistency, inconsistencies still occasionally occurred. Moreover, the system's reliance on reference images from previous scenes introduced another trade-off, namely, repetitive illustrations across scenes. Future iterations may introduce location transitions at the skeleton generation stage, for example, by adding a class for locations and marking when the narrative moves

from one place to another or when the environment should change. In this way, the corresponding images can be generated with richer variation.

The classroom deployment also exposed a trade-off between live generation and operational reliability. Although live calls enabled personalization, simultaneous text and image generation exceeded the available school network capacity and caused two games to freeze even after students were paired. A hybrid deployment could pre-generate and teacher-vet common story skeletons, scenes, and illustrations across the five bands, cache frequently used assets, and reserve live generation for constrained personalization and contextual explanations. Such a design would reduce latency and exposure to unreviewed visual inconsistencies while preserving parts of the interactive experience. A local fallback to a cached scene would also allow play to continue when a generation request fails.

Pedagogically, students perceived the game as only moderately useful for French literacy activities. One possible reason is that French was already the students' native language or language of instruction, and many selected proficiency bands below their expected curriculum level. Future iterations could use a brief baseline reading task or a teacher recommendation to establish the starting band, while still allowing students to adjust the level. The system could then recommend text slightly above the established baseline, consistent with the principle of comprehensible challenge expressed in Krashen's $i + 1$ Hypothesis (Krashen, 1977; Loschky, 1994). Whether this design would benefit second-language learners remains an open question requiring evaluation with an L2 population.

The vocabulary quiz logs provide an exploratory account of how the language-scaffolding feature was used during play. Across the six shared accounts that completed the activity, no automated normalized score was below 0.50, and two scores reached 0.875. The scores also followed a descriptive gradient: accounts with four queries scored 0.875, those with three queries scored 0.67, and those with one or two queries scored 0.50. This pattern may reflect differences in engagement, repeated exposure, prior proficiency, or collaboration within pairs. Because the logs are account-level and the LLM grader was not validated against human judgments, the pattern should not be interpreted as evidence of individual vocabulary learning.

The low uptake of the query function is itself informative for system design. Fourteen students described the game as "easy" or "too easy", and gameplay logs showed that many selected a proficiency level below their expected level. Students reading text that contained few unfamiliar words would have little reason to initiate a vocabulary query. Uptake may also have been reduced by the optional, user-initiated design: students may not have noticed the feature, may have preferred to preserve the flow of the story, or may have regarded a separate quiz as additional effort. Future versions could recommend an initial level using a brief proficiency check, adapt difficulty when an account records few queries, and make word support more discoverable through unobtrusive in-text prompts. A one-click definition followed by a short retrieval prompt could integrate practice into the narrative instead of requiring a separate activity. Reintroducing queried words in later scenes and in a delayed quiz would provide multiple opportunities to encounter and recall the words. Finally, finer-grained logs should record whether students saw, opened, and completed each stage of the feature, allowing future studies to distinguish lack of vocabulary need from problems of discoverability, interruption, or quiz completion.

Regarding generalizability, although the current version focuses on a specific genre with French-speaking primary school students, the system provides a flexible, language-agnostic architecture for interactive narratives that may be extended beyond the present use case. The underlying narrative architecture supports alternative narrative schemas, for example, for detective or science-fiction stories, by adjusting the elements that define each genre. Because the architecture is not tied to a particular language, narrative generation and proficiency-conditioning prompts can be adapted for other linguistic contexts. The growing availability of multilingual large language models further facilitates deployment across diverse language populations. The "Proficiency Benchmarker" may also be adapted to other proficiency frameworks, such as the Common European Framework of Reference for Languages (Council of Europe, 2001); however, the resulting outputs would require independent validation in each new context.

7 Conclusion

In this study, we designed, implemented, and conducted a classroom feasibility evaluation of a

multi-modal, LLM-powered interactive narrative game for French-speaking primary school learners. Findings from the classroom study indicate high perceived ease of use, positive engagement with the game system, and potential pedagogical value when the game is combined with other classroom activities. Areas for refinement in reading-level calibration, technical reliability, language scaffolding, and visual diversity were also identified. The system provides a flexible architecture for LLM-powered interactive narratives with curriculum-informed support for different intended reading levels.

Limitations and Ethical Considerations

This study has several limitations that should inform the interpretation of our results. First, data were collected from two intact classes, one at each of two schools, during one 50-minute session per class and with a limited sample size, limiting the generalisability of findings across populations, instructional contexts, and institutional settings. The participants were native French speakers or had received French-medium schooling since Grade 1; consequently, the findings concern French literacy activities in this context and cannot establish usefulness for second-language learners. Future work should replicate the study across additional schools, sessions, and learner populations.

Second, the study evaluated short-term feasibility, usability, and participant perceptions rather than learning, transfer, or retention. It included no pretest, comparison condition, or delayed assessment. In addition, the three-item PEOU scale showed marginal internal consistency ($\alpha = 0.69$), so the ease-of-use findings should be interpreted primarily at the item level rather than as a robust latent construct.

Third, the five curriculum-informed bands were implemented through teacher consultation, annotated examples, and prompting, but generated scenes were not independently audited for lexical, syntactic, or curriculum-band compliance. Nor did the study systematically evaluate the grammaticality, age appropriateness, or cross-scene coherence of generated French text. The bands should therefore be understood as generation targets rather than validated proficiency classifications.

Fourth, the vocabulary-quiz evidence is exploratory. Students played in pairs using shared accounts, so query behavior and quiz scores cannot be attributed to individual students. Only six accounts used the language-query function and completed the optional quiz, producing a small, self-selected subgroup. Quiz length and content varied with the words queried, and the automated 0/0.5/1 grades were not validated against human ratings. No criterion-referenced mastery threshold was specified. The normalized scores therefore cannot establish vocabulary mastery, learning gains, retention, or a causal relationship between query frequency and performance.

Finally, the live-generation design created technical and safety limitations. Simultaneous text and image requests exceeded the available school network capacity and caused two games to freeze. The study also did not systematically evaluate input or output moderation, model bias, or harmful and age-inappropriate content, even though the theme and scene-details fields accepted free text. Teacher supervision reduced immediate classroom risk but does not substitute for technical safeguards. Future deployments should combine age-appropriate input filtering, output moderation, restricted topic options, teacher review controls, audit logging, and cached fallbacks for failed generation requests.

From an ethical standpoint, working with children necessitates strong protections for privacy and autonomy. We adhered to GDPR principles (European Parliament and Council of the European Union, 2016), minimised collection of identifiable information, and secured research data. Informed consent and age-appropriate explanations ensured that participation was voluntary. Teachers were present during all activities, and learners were informed when content was AI-generated to support transparency and agency.

Generative models also exhibit non-deterministic outputs and sensitivity to prompt variation and provider updates. We documented prompts and model configurations to support reproducibility, but exact regeneration of outputs cannot be guaranteed.

Acknowledgments

The authors would like to thank the students and teachers who participated in the user study. Special thanks also go to Mr. Lin Shijiang for providing the computing resources necessary for this research.

References

Prithviraj Ammanabrolu, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara J. Martin, and Mark O. Riedl. 2019. Guided neural language generation for automated storytelling. In *Proceedings of the Second Workshop on Storytelling*, pages 46–55, Florence, Italy. Association for Computational Linguistics.

Omri Avrahami, Amir Hertz, Yael Vinker, Moab Arar, Shlomi Fruchter, Ohad Fried, Daniel Cohen-Or, and Dani Lischinski. 2024. The chosen one: Consistent characters in text-to-image diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12. ACM.

Council of Europe. 2001. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Cambridge University Press, Cambridge, UK.

Fred D. Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340.

Edirlei Soares de Lima, Bruno Feijó, and Antonio L. Furtado. 2021. Adaptive branching quests based on automated planning and story arcs. In *Proceedings of the 20th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, pages 9–18.

European Parliament and Council of the European Union. 2016. Regulation (EU) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (General Data Protection Regulation). Official Journal of the European Union.

Sarah Harmon and Sophia Rutman. 2023. Prompt engineering for narrative choice generation. In *Interactive Storytelling*, volume 14383 of *Lecture Notes in Computer Science*, pages 208–225. Springer Nature Switzerland.

Runsheng Huang, Lara J. Martin, and Chris Callison-Burch. 2024. WHAT-IF: Exploring branching narratives by meta-prompting large language models. *Preprint*, arXiv:2412.10582.

Mete Ismayilzada, Claire Stevenson, and Lonneke van der Plas. 2024. Evaluating creative short story generation in humans and large language models. *Preprint*, arXiv:2411.02316.

Stephen D. Krashen. 1977. The input hypothesis. In *On TESOL '77: Teaching and Learning English as a Second Language: Trends in Research and Practice*.

Vikram Kumaran, Jonathan Rowe, Bradford Mott, and James Lester. 2023. SceneCraft: Automating interactive narrative scene generation in digital games with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 19, pages 86–96. Association for the Advancement of Artificial Intelligence.

Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ali Payani, Ninghao Liu, and Mengnan Du. 2025. Language ranker: a metric for quantifying llm performance across high and low-resource languages. AAAI'25/IAAI'25/EAAI'25. AAAI Press.

Lester Loschky. 1994. Comprehensible input and second language acquisition. *Studies in Second Language Acquisition*, 16(3):303–323.

Richard E. Mayer. 2024. The past, present, and future of the cognitive theory of multimedia learning. *Educational Psychology Review*, 36(1):8.

James R. Meehan. 1977. TALE-SPIN, an interactive program that writes stories. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, volume 1, pages 91–98.

Hiroshi Nonaka and K. E. Perry. 2025. Evaluating LLM story generation through large-scale network analysis of social structures. *Preprint*, arXiv:2510.18932.

Jum C. Nunnally and Ira H. Bernstein. 1994. *Psychometric Theory*, 3rd edition. McGraw-Hill, New York.

Allan Paivio. 1979. *Imagery and Verbal Processes*. Psychology Press, New York.

David L. Parnas. 1972. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12):1053–1058.

Fei Shen, Hu Ye, Sibo Liu, Jun Zhang, Cong Wang, Xiao Han, and Yang Wei. 2025. Boosting consistency in story visualization with rich-contextual conditional diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 6785–6794. Association for the Advancement of Artificial Intelligence.

Yuling Sun, Jiaju Chen, Bingsheng Yao, Jiali Liu, Dakuo Wang, Xiaojuan Ma, Yuxuan Lu, Ying Xu, and Liang He. 2024. Exploring parent's needs for children-centered ai to support preschoolers' interactive storytelling and reading activities. 8(CSCW2).

Yufei Tian, Tenghao Huang, Miri Liu, Derek Jiang, Alexander Spangher, Muhao Chen, Jonathan May, and Nanyun Peng. 2024. Are large language models capable of generating human-level narratives? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17659–17681, Miami, Florida, USA. Association for Computational Linguistics.

Nick Walton. 2019. AI Dungeon. Developed by Latitude.

Qiao Wang, Adnan Labib, Robert Swier, Michael Hofmeyr, and Zheng Yuan. 2025. GenQuest: An LLM-based text adventure game for language learners. *Preprint*, arXiv:2510.04498.

Tiffany Wang, Yuqian Sun, Yi Wang, Melissa Roemmele, John Joon Young Chung, and Max Kreminski. 2026. Design techniques for LLM-powered interactive storytelling: A case study of the Dramamancer system. *Preprint*, arXiv:2601.18785.

Weiqi Wu, Hongqiu Wu, Lai Jiang, Xingyuan Liu, Jiale Hong, Hai Zhao, and Min Zhang. 2024. From role-play to drama-interaction: An LLM solution. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3271–3290. Association for Computational Linguistics.

Tianyang Xu, Hongqiu Wu, Weiqi Wu, and Hai Zhao. 2025. OPEN-THEATRE: An open-source toolkit for LLM-based interactive drama. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 453–460. Association for Computational Linguistics.

Qiang Yi, Yangfan He, Jianhui Wang, Xinyuan Song, ShiYao Qian, Xinhang Yuan, Yi Xin, Yijin Wang, Jingqun Tang, Yuchen Li, Junjiang Lin, Hongyang He, Zhen Tian, Tianxiang Xu, Keqin Li, Kuan Lu, Menghao Huo, Jiaqi Chen, Miao Zhang, and 2 others. 2025. SCORE: Story coherence and retrieval enhancement for AI narratives. *Preprint*, arXiv:2503.23512.

R. Michael Young, Stephen G. Ware, Bradly A. Cassell, and Justus Robertson. 2013. Plans and planning in narrative generation: A review of plan-based approaches to the generation of story, discourse and interactivity in narratives. *Sprache und Datenverarbeitung, Special Issue on Formal and Computational Models of Narrative*, 37(1-2):41–64.

A Story elements designed for fables

```
# Story Elements Reference

The story should incorporate the following classic
narrative elements:

1. Stable beginning
e.g., Known environment (maison, forêt, village)

2. Desire / problem
e.g., Wanting something
(go to the ball, eat, explore, help)

3. Obstacle
e.g., Villain, rule, fatigue, danger, limitation

4. Repeated attempts
e.g., Often 2 or more times with increasing intensity

5. Resolution
e.g., Help, cleverness, cooperation

6. Explicit moral
e.g., Stated directly, not inferred

# Character Design Patterns

- Protagonist: Childlike, curious, imperfect
- Helper: Animal, fairy, friend, object
- Obstacle: Not morally complex
- Collective (optional): Group solves problem together
```

B Five proficiency bands

```
# Level Guidelines

Level 1
Description:
Very short sentences. One concrete action per sentence.
Only high-frequency, concrete vocabulary.
Simple connectors (et, puis, mais, alors).

Example:
Le petit garçon entre dans la forêt. Il voit une
maison. Il a faim alors il frappe à la porte.

Level 2
Description:
Simple narration with basic subordination
(quand, parce que, comme).
Limited description and explicit cause-effect.

Example:
Quand le petit garçon entre dans la forêt, il voit
une petite maison. Comme il a faim, il frappe à
la porte.

Level 3
Description:
More compact language with participial
structures and varied vocabulary.
Do not introduce new events.

Example:
En entrant dans la forêt, le petit garçon aperçoit
une petite maison isolée. Après un moment,
il décide d'entrer.

Level 4
Description:
More fluent narration with light abstract nouns
(e.g., silence, hésitation), while keeping the
same events.

Example:
En entrant dans la forêt, le petit garçon aperçoit
une maison isolée. Après un moment, il décide
d'entrer.

Level 5
Description:
Condensed literary language with richer
vocabulary. Use light metaphor without adding
new narrative content.

Example:
Au cœur de la forêt, le petit garçon distingue
une maison solitaire. La faim, plus forte que la
prudence, l'incite à frapper.
```

C Player Survey (administered in French)

Instructions: Please read each statement carefully and indicate how much you agree or disagree.

Response scale:

7-point Likert scale:

1 = Strongly disagree;
2 = Disagree;
3 = Slightly disagree;
4 = Neither agree nor disagree;
5 = Slightly agree;
6 = Agree;
7 = Strongly agree.

Perceived Usefulness (PU)

1. Using the game in my French class may enable me to learn to better read and understand texts.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

2. Using the game may improve my French performance.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

3. I find the game useful in learning French.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

Perceived Ease of Use (PEOU)

4. Learning to play the game was easy for me.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

5. My interaction with the game was clear and understandable.

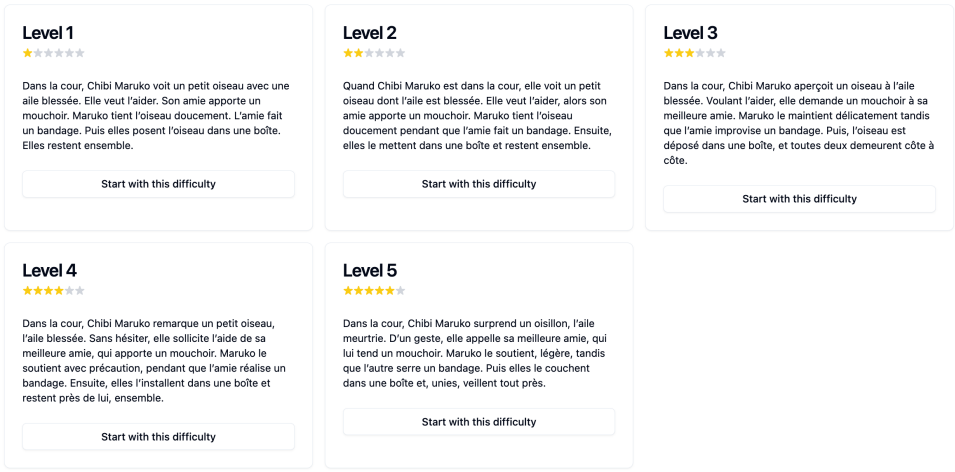


Figure 5: Text samples of the five proficiency bands

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

6. I found the game clear and understandable.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7

Open-Ended Questions

7. Draw what you liked about the game in the space below.

Drawing space

8. Did you like the story of the game?

_____

9. What do you think about the illustrations in the game?

_____

10. Were the stories too difficult, too easy, or just right for you?

_____

11. Would you recommend this game to your classmates or to your friends at school?

_____

12. Do you have any other suggestions or comments about the game?

_____

D Teacher Survey

1. **Pedagogical Coherence:** Do you think that the generated narratives maintain sufficient logical flow for regular classroom use? How well did the levels of difficulty adapt to your students' needs?
2. **Written Production:** Can you imagine a way to use the game to help mastery of written expression and textual genres?
3. **Social-Emotional Learning:** Could the game be used to promote the development of empathy and social competencies through narrative engagement? If so, in what ways?
4. **Oral Communication:** Does the game provide opportunities for verbal interaction and storytelling?
5. **Creativity:** To what degree does the game foster original thought and narrative divergent thinking?

E Example of students' favorite illustrations



Figure 6: A student's favorite illustration

A Training-Free Single-Pass Vision–Language Pipeline for Sentence-Level Visual Chinese Grammatical Error Correction

Lexuan Zhang
KTH Royal Institute of Technology

Abstract

Handwritten text by language learners couples recognition errors with grammatical errors, which makes automatic correction of such text harder than either subproblem alone. Visual Chinese grammatical error correction asks a system to read a handwritten sentence image and to emit both a fully corrected sentence and a list of localized visual edits. We describe a training-free pipeline that makes one call to a vision–language model (VLM) per image and jointly requests a verbatim transcription, a minimal-edit correction, and a typed bounding-box edit list in a single JSON object; a deterministic packer enforces schema and coverage, and a second engine fills only the test images for which the first produces no valid record. A paired 19-image development comparison under a proxy of the official $F_{0.5}$ metric finds that two VLMs obtain the same edit true-positive count while one emits nearly twice as many edits; the more conservative engine has higher edit precision on this slice. The system obtains an official score of 0.5905 on the hidden sentence-level test set of the VisCGEC-v2.0 shared task. The result suggests that a single-pass training-free VLM pipeline is viable for this task, and that edit precision is a useful engine-selection signal on the measured development slice.

1 Introduction

Grammatical error correction (GEC) is usually defined over already-tokenized text (Ng et al., 2014), and Chinese GEC has an established shared-task lineage over learner text (Zhao et al., 2018; Rao et al., 2020; Chang et al., 2023). Visual Chinese GEC (Wang et al., 2025) additionally requires the system to read a real handwritten image produced by a learner of Chinese as a foreign language, so recognition errors, malformed characters, and grammatical errors are entangled. The VisCGEC-v2.0 shared task of Multi-LLL 2026 (Multi-LLL Organizers, 2026) makes that setting operational: the official input is the image; the public test file contains only an identifier and a path; and each prediction must contain a corrected string `predict_text` together with a list of localized edits.

Each edit is a pixel box, a corrected span, and one of four operation types (`Insert`, `Delete`, `Substitute`, `WordOrder`). The official score is the mean of a per-sample combination of text-level $F_{0.5}$ and edit-level $F_{0.5}$, so a fluent rewrite that does not match the annotators' boxes is not enough.

This paper describes the sentence-level system we submitted, a real system that obtained an official score of 0.5905 on the hidden test set (Section 5). Our contributions are:

- a training-free, single-pass pipeline in which one VLM call per image jointly produces a transcription, a correction, and typed bounding-box edits, backed by a deterministic packer that hard-fails on any schema or coverage violation (Section 3);
- an offline, self-validated proxy of the official metric with explicit fidelity limits, used for engine selection on the 245-image development split (Section 5);
- a paired development observation: two engines tie on edit true positives while one emits nearly twice as many edits, making edit precision a useful engine-selection signal on this slice (Section 6).

We did not train or fine-tune any model. We release the prompt template, the offline scorer and submission packer, and both submitted prediction files in an anonymized repository.[1]

[1] `https://anonymous.4open.science/r/viscgec-sentence-pipeline-F466`

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 77–83
November 9, 2026 ©2026 Association for Computational Linguistics

2 Data

Released splits. Sentence-level VisCGEC-v2.0 provides 1,960 training images, 245 development images, and 246 hidden-test images. Training and development records contain `img_id`, `img_path`, a gold corrected string `target_text`, and a gold `edit_list`. They do *not* contain the learner's original string. The test file contains only `img_id` and `img_path`. Systems must therefore recognize text from the image; they cannot assume an OCR transcript.

Annotation shape. On the 1,960 training sentences the gold `edit_list` has mean length 1.80 (median 1, max 14); six sentences have an empty list. Edit types occur in the ratio Substitute 1,693 : Insert 1,079 : Delete 761. `WordOrder` occurs *zero* times in the released training annotations, and also zero times in the 245 development annotations (Substitute 212, Insert 134, Delete 89; mean 1.78 edits). Mean gold sentence length is 24.6 characters (development 25.5). Images are single-line handwriting strips (training width p50 1,494 px, height p50 85 px; test width p50 1,531 px, height p50 93 px).

Why we do not fine-tune. A text GEC trainer wants pairs $(\textit{source}, \textit{target})$. The public files give *target* and a list of *corrected* spans with boxes, but not the original spans, so *source* cannot be reconstructed from the JSON. Obtaining N training pairs would require first OCRing N training images and then trusting that OCR. We did not pay that cost, and we used the training split only (a) to compute the edit-type and edit-count statistics above and (b) as documentation for an optional convention-aware prompt variant. The development split is the only split with gold that we scored.

Scope. The shared task also defines a paragraph-level track over the same schema; we did not submit to it, and the rest of this paper is sentence-level only.

External resources. The only task data we used are the official public packs. Inference uses closed VLMs accessed through an API (Section 4); we did not add an external GEC corpus.

3 Methodology

Figure 1 shows the primary decoding path and the fallback condition. The pipeline operates as follows:

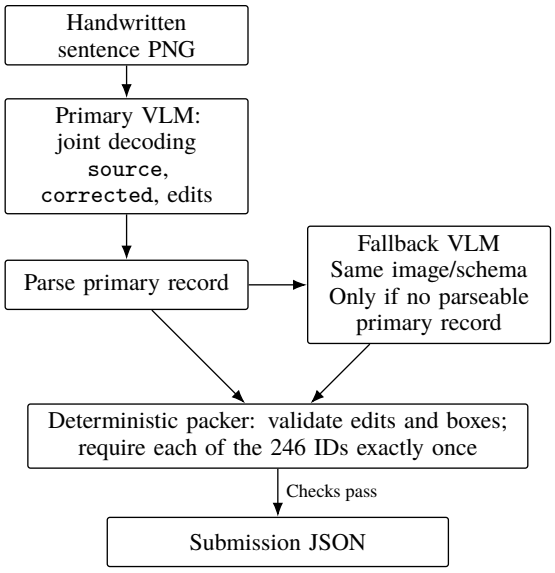


Figure 1: The primary record is retained when parseable. The fallback fills only missing records. The packer refuses to emit a submission when coverage or schema checks fail.

1. Pass one handwritten PNG to a VLM as visual input at original resolution. Invocation goes through a CLI harness: the prompt names the image file and its pixel dimensions, and the harness reads the file and attaches it to the API request.

2. Ask a VLM for a single JSON object with three fields: `source` (verbatim transcription, errors kept), `corrected` (minimal-edit fluent Chinese), and `edit_list` (the visual difference between the two).

3. Parse the last JSON object in the response that contains `source` and `corrected`. Drop edits whose box is non-integer or degenerate, whose type is outside the four-way inventory, or whose `correct_text` is not a string. Deletion text must be the empty string.

4. Set `predict_text` = `corrected`. Round boxes to integers and clip them to the image. Require exact coverage of the 246 test IDs (no missing, extra, or duplicate IDs).

5. If an image has no parseable record from the first engine, fill it from a second engine; a record that already parsed is never overwritten.

6. Write the submission file under the official sample filename

(`sample_submission.json`), packaged as the platform requires.

Prompt. The production template used for both submissions is a short instruction that (i) forbids reading any annotation file, (ii) states the four edit types, (iii) requires Insert boxes to cover a full neighbouring character cell rather than a zero-width line, and (iv) demands one JSON line and nothing else. We also iterated on a longer convention-aware variant of the same template, which tells the model that the reference makes 1.8 edits per sentence, that `WordOrder` is unused in training, and that style improvements and extra connectives are not rewarded, and which quotes eight gold corrected sentences. All 246 archived test prompts instantiate the short template, so we make no claims for the variant. The schema—joint transcription, correction, and boxes—is the same in both templates and is what the packer consumes.

Why joint decoding. Text-level $F_{0.5}$ in the Chinese setting is a comparison of character edits *source* → *gold* against *source* → *prediction*, in the style of ChERRANT (Zhang et al., 2022), itself an adaptation of ERRANT (Bryant et al., 2017). If a system copied its own OCR into `predict_text` and emitted an empty edit list, the text-level term would be zero whenever the OCR differed from the hidden source, and the edit-level term would be zero by construction; the sample score would then be at most 0.5 in the empty-edit case and typically near 0. The same VLM call therefore has to both read and correct, and it has to emit boxes that line up with the annotators.

Deterministic shell. All coverage and schema checks run in a packer that refuses to write the submission file if any ID is missing or illegal. That shell is the only component we treat as bit-stable across engines.

4 Model

We do not train weights; all correction capability comes from commercial vision–language models accessed through an API. A submission in this task is a single file with one record per test image, and the platform allows each team only a small fixed number of scored submissions. We submitted twice: a *contrastive* configuration built first under time pressure, and a *primary* configuration built after development measurement; the platform keeps each team's best-scoring submission. Three engines appear across the two submissions, identified by the router names our access channel recorded at inference time (these are channel identifiers, not necessarily the vendors' public model names; where a channel exposed two aliases, we give both):

- **Qwen** (`qwen3.8-max`), the main engine of the contrastive submission;
- **GPT** (`gpt-5.6-sol / gpt:sol-medium`), the fallback engine in both submissions;
- **Claude** (`claude-opus-5 / claude:high`), the main engine of the primary submission.

A fourth engine (GLM 5.3) was probed on development images and rejected: an 18-image preview scored below the contrastive configuration, and that engine sometimes returned a tool trace instead of the required JSON.

Primary submission. 224 of the 246 records come from Claude. The remaining 22 records are the images for which Claude's output was missing (14), empty (4), or unparseable (4), and are filled by GPT. This fallback is a reliability backfill, not a learned router: a record that already parsed is never overwritten.

Contrastive submission. The earlier submission ran Qwen as the main engine over all 246 test images and applied the same fallback rule, yielding 119 Qwen records and 127 GPT records. Section 5 compares the two configurations on development measurements.

Implementation details that we can state. Each call passes one image (attached by the harness as visual input) and one prompt file; at least one engine's logs show an explicit image-analysis step, so we do not claim to know how each engine consumes the pixels internally. Batch inference used a per-call timeout of 900 seconds and concurrency between 4 and 8. We did not record sampling parameters or exact token limits, and we do not invent them. There is no reranking, no second-pass box refiner, and no lexical GEC model on top of OCR.

5 Evaluation

Official metric. The task defines

$$F_{0.5} = \frac{1.25\,PR}{0.25\,P + R},$$

$$\textit{sample} = 0.5\,\textit{text-}F_{0.5} + 0.5\,\textit{edit-}F_{0.5},$$

and takes the mean over samples. An edit is a true positive only if its box has IoU ≥ 0.5 with a gold box, its `correct_text` matches exactly, and its `edit_type` matches. The task description names ChERRANT as the text-level metric; its implementation, tokenization, and empty-gold convention were not released.

Local proxy. We reimplemented the published formulae as a standalone offline scorer, used only at development time for engine selection and mix control. Text-level $F_{0.5}$ compares character-level Levenshtein edits rather than ChERRANT, so we treat the number as a trend rather than a leaderboard forecast. The scorer is checked against hand-computed cases and negative controls: gold-as-prediction scores 1.0, an empty edit list caps the sample score at 0.5, and shifting every box by 30 px degrades edit $F_{0.5}$ to 0.25. Because we do not have the hidden source string, we compute two text-level branches on development images: an *optimistic* branch that uses the model's own `source` as the ChERRANT source, and a *pessimistic* branch that adds extra false positives for estimated OCR errors, $\max(0,\ d(\texttt{source}, \texttt{target}) - g)$, where d is the character edit distance between the model's transcription and the gold corrected text and g is the number of characters the gold edit list changes. Edit matching is greedy one-to-one, sorted by IoU; empty gold and empty prediction score 1.0. TIER2 is the mean sample score using both terms; TIER1 zeroes the edit term and is not used for selection.

Official result. The primary submission obtains an official overall score of 0.5905 on the hidden test set under the metric above. The platform reports only each team's best-scoring submission, so the contrastive submission has no separately visible official score, and itemized text-level and edit-level sub-scores are not exposed for any team; the remainder of this section therefore analyses the system on the development split. The local proxy numbers below are not directly comparable with the official score: the proxy uses character-level Levenshtein edits instead of ChERRANT, a different split, and a source-side approximation.

Development engine faces. Table 1 reports TIER2 on a 50-image development slice (Claude n=49; one item timed out twice at 600 s). All development figures are single runs. Claude is strongest; GPT is weakest; Qwen sits in between. A 19-image paired slice that holds the image set fixed (Table 2) shows a precision difference: both Claude and Qwen obtain 6 edit true positives against 29 gold edits, but Qwen emits 58 predicted edits against Claude's 31. The TIER2 gap is almost entirely edit-level precision.

Engine	n	TIER2 opt.	TIER2 pess.
Qwen	50	0.230	0.204
GPT	50	0.171	0.160
Claude	49	0.268	0.242

Table 1: Development-set TIER2 by engine. Optimistic/pessimistic refer to the source proxy for text-level $F_{0.5}$. Not hidden-test scores.

	TIER2 pess.	edit $F_{0.5}$	pred.	TP
Claude	0.213	0.232	31	6
Qwen	0.159	0.108	58	6

Table 2: Paired 19-image development slice (same gold, 29 gold edits). Equal edit true-positive counts accompany different predicted edit counts and precision. Unmatched edits may reflect correction, box, or type mismatches.

Mix-weighted control line. Applying the contrastive submission's engine weights (119 Qwen +127 GPT) to the 50-image faces gives a TIER2 of 0.199/0.181. The primary submission's weights (224 Claude +22 GPT) give 0.259/0.235 on the same faces (Table 3). All six recorded axes move up, including edit $F_{0.5}$ ($0.145 \rightarrow 0.184$) and predicted character accuracy ($0.773 \rightarrow 0.869$). A pure-GPT replacement (0.171/0.160) is worse than the contrastive mix and was not used. This arithmetic is a development-face linear mix; it is *not* a score of the 246 test predictions (the test gold is hidden), nor a measured evaluation of either complete fallback pipeline on the same development images. Because fallback is used on specific failures of the primary engine, its average development score need not describe those selected images.

6 Analysis

Edit precision as an engine-selection signal. The paired comparison supports selecting an engine by edit precision on the measured development slice. It does not establish over-editing as the dominant cause of failure: an unmatched edit can also reflect a bounding-box or edit-type mismatch. On a 20-image development sample, character disagreement between the model's transcrip-

	Contrastive	Primary	Δ
TIER2 opt.	0.199	0.259	+0.060
TIER2 pess.	0.181	0.235	+0.053
text $F_{0.5}$ opt.	0.254	0.334	+0.080
text $F_{0.5}$ pess.	0.218	0.285	+0.067
edit $F_{0.5}$	0.145	0.184	+0.040
pred. char acc.	0.773	0.869	+0.096

Table 3: Mix-weighted development faces. Contrastive $= (119/246)$ Qwen $+(127/246)$ GPT; Primary $= (224/246)$ Claude $+(22/246)$ GPT. Development estimates, not hidden-test scores.

tion and the gold corrected text, net of the gold edits themselves, is around 3%. This is a source-side proxy, not a manually measured transcription error rate, and it cannot rule out recognition errors as a major factor. One recorded example illustrates disagreement with the reference: on development item `1974_1`, the gold correction inserts the connective "但是随着" at the start of the strip, whereas the model inserts a sentence-final "了". This example motivates checking unnecessary edits, but it does not quantify their prevalence. Manually checked transcriptions and an analysis of unmatched edits would be needed to separate recognition, correction, box, and type errors.

Optimization trajectory. The two submissions are one measured iteration, not two systems. The contrastive submission was assembled first, under the platform's fixed submission quota. The paired development measurement favoured the engine with higher edit precision—engines tied on true positives yet differed nearly two-fold in emitted edits—so the primary submission re-ran the test set with the more conservative engine and refilled only its 22 missing records. The corresponding mix-weighted estimates are higher on every recorded development axis (Table 3); they do not establish the gain of the actual fallback pipeline. Two cheaper levers were probed and not used in the final calls: a convention-aware prompt variant that encodes training-split statistics (Section 3), and a fourth engine dropped after an 18-image preview scored below the contrastive configuration (Section 4). Outside the sentence track, a pure-geometry feasibility check—in the released paragraph-level training annotations, all 2,237 gold Insert boxes are identically 4 px wide, against which full-cell boxes cannot reach IoU ≥ 0.5—concentrated the entire submission budget on the sentence track.

Submitted edit inventories. On the 246 test predictions themselves (no gold) the contrastive submission contains 584 edits (mean 2.37 per sentence) and the primary submission 435 (mean 1.77). The training mean is 1.80. Type counts: contrastive Substitute 234 / Insert 155 / Delete 114 / `WordOrder` 81; primary Substitute 208 / Insert 111 / Delete 89 / `WordOrder` 27. `WordOrder` is unused in the released training and development gold, although it is permitted by the schema. The hidden-test gold is unavailable, so these counts alone do not establish that every such test prediction is incorrect. Reducing that count from 81 to 27 is a side-effect of preferring Claude, not a dedicated type filter—we did not post-hoc drop `WordOrder`. Five test items in each submission have an empty edit list (two IDs overlap).

Coverage mix versus fusion. The 22 GPT records in the primary submission exist only because a valid submission must cover all 246 test images. They are a reliability backfill, not an ensemble vote. GPT's average development TIER2 is the weakest of the three engines, but that does not determine its performance on the images where Claude produces no parseable record. The recorded linear mix is above the contrastive line on every local axis; the effect of fallback requires measuring complete pipelines on the same development images.

Boxes. Insert boxes on the sentence-level gold are real character cells (width p50 52 px), which is what the prompt asks for. We did not add a geometric refiner: illegal and degenerate boxes are dropped, and the primary submission dropped none.

What we did not try. We did not OCR the 1,960 training images to build GEC pairs; we did not fine-tune; we did not run a two-stage OCR-then-GEC pipeline as a submitted system; we did not enter the paragraph-level track.

7 Conclusion

We have described a training-free, single-pass vision–language pipeline for sentence-level visual Chinese grammatical error correction that obtains an official score of 0.5905 on the hidden test set. The design is minimal by choice: one model call per image, a rigid output contract, and a deterministic packer. On the paired 19-image development slice, the engine that emits fewer edits has the same

edit true-positive count and higher edit precision. This makes edit precision a useful selection signal for that comparison; it does not isolate recognition errors or establish the benefit of the complete fallback pipeline.

Limitations

The local scorer is not ChERRANT and cannot be claimed to match the official script. Development faces have n=50 (Claude n=49), and the mix table linearly combines those faces; it does not evaluate the 246 test predictions against hidden gold. We do not have official itemized scores. Decoding hyperparameters were not logged. Closed VLMs are opaque: we cannot report architecture, context length actually used, or pretraining data. The system is sentence-level only. API engines are a moving target; the identifiers above are those recorded at submission time.

Declaration on Generative AI

The system described in this paper *is* a generative AI system: each test image is corrected by a commercial vision–language model (Section 4) under a fixed prompt and a deterministic packer. No generative model was used to invent evaluation numbers; every figure in Section 5 is copied from run logs or recomputed from the submitted JSON. This manuscript was drafted with assistance from a coding agent that was instructed to quote only logged facts and to mark gaps rather than fill them. The authors take responsibility for the claims that remain.

References

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Hongxiang Chang, Yang Liu, Meng Xu, Yingying Wang, Cunliang Kong, Liner Yang, Yang Erhong, Maosong Sun, Gaoqi Rao, Renfen Hu, and Zhenghao Liu. 2023. CCL23-Eval 任务 7 总结报告: 汉语学习者文本纠错 (Overview of CCL23-Eval Task: Chinese Learner Text Correction). In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 3: Evaluations)*, pages 239–249, Harbin, China. Chinese Information Processing Society of China.

Multi-LLL Organizers. 2026. The workshop on multimodality and multi-lingual language learning (Multi-LLL 2026). https://xiaoaugenstern.github.io/Multi-LLL-2026/. Shared task: VisCGEC-v2.0. Accessed 13 September 2026.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Gaoqi Rao, Erhong Yang, and Baolin Zhang. 2020. Overview of NLPTEA-2020 shared task for Chinese grammatical error diagnosis. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 25–35, Suzhou, China. Association for Computational Linguistics.

Xiaoman Wang, Dan Yuan, Xin Liu, Yike Zhao, Xiaoxiao Zhang, Xizhi Chen, and Yunshi Lan. 2025. VisCGEC: Benchmarking the visual Chinese grammatical error correction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5054–5068, Albuquerque, New Mexico. Association for Computational Linguistics.

Yue Zhang, Zhenghua Li, Zuyi Bao, Jiacheng Li, Bo Zhang, Chen Li, Fei Huang, and Min Zhang. 2022. MuCGEC: a multi-reference multi-source evaluation dataset for Chinese grammatical error correction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3128–3143, Seattle, United States. Association for Computational Linguistics.

Yuanyuan Zhao, Nan Jiang, Weiwei Sun, and Xiaojun Wan. 2018. Overview of the NLPCC 2018 shared task: Grammatical error correction. In *Natural Language Processing and Chinese Computing*, pages 439–445. Springer.

A Production prompt and output contract

The production prompt instantiates the following template once per image, with `<PATH>`, `<W>`, and `<H>` filled by the image path and its pixel dimensions:

> Read the image file at `<PATH>` with your Read tool. The image is exactly `<W>` pixels wide and `<H>` pixels tall. It is a cropped strip of handwritten Chinese, one sentence, written by a Chinese-as-a-second-language learner. The sentence contains grammatical errors (wrong character, missing character, redundant character, wrong word order). Produce three things:

(1) `source`: verbatim transcription of what the student wrote, keeping every error exactly as written; (2) `corrected`: the same sentence with all grammatical errors fixed into fluent standard Chinese, changing as little as possible; (3) `edit_list`: the localized corrections, each with `bbox` (integer pixel coordinates `start_x`, `start_y`, `end_x`, `end_y` in this image's original coordinate system, covering the character cell(s) involved; for an `Insert` the box covers a full character-cell-sized region at the insertion position, not a zero-width line), `correct_text` (the corrected text for that spot; the empty string for `Delete`), and `edit_type` (one of `Insert`, `Delete`, `Substitute`, `WordOrder`). Hard constraints: do not search the filesystem for any answer key, annotation file, dataset JSON, or ground truth—the only permitted input is the image itself; coordinates must be measured off this image; output exactly one line of JSON and nothing else.

The file-reading phrasing reflects the invocation harness described in Section 4: it reads the named file and attaches it to the API call as visual input.

The packer accepts the last JSON object in the response that contains `source` and `corrected`; drops edits with non-integer or degenerate boxes, out-of-inventory types, or non-string `correct_text`; requires an empty `correct_text` for `Delete`; rounds and clips boxes to the image; and refuses to emit the submission file unless the 246 test IDs are covered exactly once.

Outlining an Experiment Design of Gamifying Language Learning in Low-Resource Contexts: Kurdish Learning in University, Language Courses, and on Web-Based Applications

Quánzī Mǎ[1]*, Christian Schuler[1,2]*, Xueqing Zhang[3], Meiyi Chen[1], Mahîr Dogan[4], Raman Ahmad, Tomohiro Nagashima[1], Timo Baumann[5], Josef van Genabith[1,2]

[1]Saarland University, [2]DFKI, [3]Beijing Foreign Studies University,
[4]University of Bamberg, [5]OTH Regensburg

Correspondence: quma00001@stud.uni-saarland.de & christianschuler8989@gmail.com

Abstract

Low-resourced and marginalized languages suffer from generational attrition, with dwindling incentives for younger speakers to learn a language to which they may be linked culturally but which offers limited opportunities for use in education, employment, and other domains of everyday life. Gamification techniques and the integration of digital support structures may hold considerable potential for language revitalization and digitization efforts. This work explores a research design that explores approaches targeting both extrinsic and intrinsic motivation for learners of Kurdish varieties across three contrasting learning environments. By running comparable conditions in environments that differ sharply in institutional pressure, cost, and voluntariness, we aim to produce a multifaceted account of how gameful design and behavior-oriented guidance interact with learner contexts, and thereby inform future initiatives in low-resource language learning.

1 Introduction

Language technology support differs radically across the world's languages. Only a very small number of languages are represented in language technologies and applications (Joshi et al., 2020). Kurdish, in contrast, is considered a low-resource language (Kamal and Hassani, 2020) with the lack of corpora as one of the main obstacles in Kurdish language processing (Abdulrahman et al., 2019). These resource limitations also make it more difficult to develop digital language-learning resources, since the tooling that supports gameful design in well-resourced settings, such as automatic assessment and generated feedback, is largely unavailable. Zhang (2026) highlights that availability of high-quality didactic materials is one of the reportedly prominent problems shared in minority language teaching (van Dongera et al., 2017)

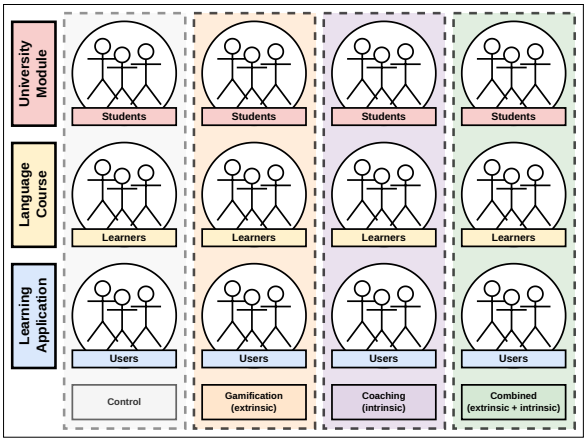


Figure 1: The condition matrix of the planned study in which rows correspond to the three environments and columns to the four condition families.

and that literature on Kurdish language teaching is still very limited (Muhammadamin, 2025; Resul and Muhammadamin, 2015). According to Zhang (2026), one of the proposed solutions is the integration of traditional and modern pedagogical methods (Ali and Saeed, 2025), something our planned work directly acts upon.

Gamification is commonly defined as "the use of game design elements in non-game contexts" (Deterding et al., 2011). Previous research has reported potential benefits of gamification for motivation, engagement, and learning-related outcomes, but reported effects remain inconsistent and appear to depend on the specific game elements employed, the learning context, learner characteristics, and the evaluation methods used (Dichev and Dicheva, 2017; Sailer and Homner, 2020; Luo, 2023). Much of this evidence base is drawn from well-resourced languages and from settings in which learners have a clear instrumental motive for study.

This assumption may not hold uniformly. For instance, Chinese student's learning Kurdish may reflect a mix of factors beyond institutional or ca-

* Contributed Equally

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 84–93
November 9, 2026 ©2026 Association for Computational Linguistics

reer incentive, including geopolitical engagement tied to the Belt and Road Initiative (Chaziza, 2024; World Bank, 2019), longstanding cultural and literary ties between the two peoples (Dadparvar and Shams, 2024), and identity-based motivations such as the Kurdish concept of *"beganeperestî"*, a documented tendency to value and support outsiders as an expression of cultural or identity-preserving duty (Jasim, 2024). If extrinsic incentives alone do not fully account for learner motivation in these contexts, gamification approaches that lean primarily on extrinsic feedback may be insufficient on their own, motivating our inclusion of an agent-based intervention targeting intrinsic, self-regulation-driven motivation.

Currently, there is a lack of research on how learners perceive the learning activities and gameful design in low-resource language-learning contexts. Neither is it clear whether findings obtained in one kind of environment transfer to another, such as a student facing examination pressure and a user of a free application who faces very different opportunity costs for disengagement. We therefore propose a study that holds the intervention families comparable while varying the environment, for which we formulate three research questions:

RQ1 How does gameful redesign of existing learning activities affect learning outcomes and self-reported motivation in low-resource language learning?

RQ2 Does the addition of an agent that targets self-regulation and learning behavior yield effects beyond those of gameful design alone?

RQ3 How do these effects differ across environments that vary in institutional pressure, financial investment, and voluntariness of participation?

2 Related Work

Kurdish Language Varieties Kurdish is part of the Indo-European language family and presents a diverse spectrum of varieties which use Latin, Perso-Arabic, and slightly modified writing systems (Tavadze, 2019). The most prominent variety-groups and their language codes according to ISO-639 are Northern Kurdish (kmr), also known as Kurmanjî (Haig and Öpengin, 2018), Central Kurdish (ckb), known as Soranî (Sheyholislami, 2021), Southern Kurdish (sdh) (Ahmadi et al., 2023), Zazaki (zza) (Dogan, 2024), Gorani (hac), often called Hewramî (Mohammadirad, 2024), and Laki (lki) (Belelli, 2021).

Kurdish' Low-Resource Status Ahmadi (2020) describes Kurdish as a less-resourced language and points out the lack of basic language processing tools. For Central Kurdish, Ahmadi and Masoud (2020) report scarce parallel data for Kurdish-English machine translation and more recent work finds that language technology resources are unevenly available across dialectal varieties (Ahmadi et al., 2024). Even though recent years have seen exciting progress in Kurdish natural language processing (NLP) (Hassani, 2018; Morad et al., 2024; Ahmadi et al., 2024), low-resource circumstances of language technology extend to the Kurdish learning context studied here.

Digital Language Learning Resources Digital language learning resources are used in many formats and learning contexts. In Computer-Assisted Language Learning (CALL), software and learning activities can be evaluated using criteria such as language learning potential, learner fit, meaning focus, authenticity, positive impact, and practicality (Chapelle, 2001). Research on mobile-assisted and gamified language learning also covers different types of applications, educational levels, learning settings, and evaluation measures (Ishaq et al., 2021). However, research on digital game-based second language (L2) learning has shown that most studies were conducted in formal learning environments and that English was usually the target language (Acquah and Katz, 2020).

Gamification in Language Learning Gamificaton is frequently used in language learning by integrating game elements such as points, feedback, quizzes, badges, leaderboards, rewards, and progress indicators into learning activities (Zhang and Hasim, 2023) These elements help provide feedback, and make learners' progress more visible. Such benefits do not always translate into better learning outcomes. A systematic review of foreign language learning research indicates that findings range from positive and negative effects to no differences (Luo, 2023). Existing evidence remains weak evidence for long-term effects and insufficient understanding of how gamification should be adapted to specific educational contexts (Dichev and Dicheva, 2017).

Self-Regulation & Behavioral Changes Self-regulated learning (SRL) describes learning as an

active process in which learners plan, monitor, and adjust their own learning. This includes setting goals, selecting and applying strategies, evaluating progress, and responding to feedback (Jivet et al., 2018). Major SRL frameworks conceptualize these activities as cyclical. Zimmerman (2000) distinguishes three phases: forethought, performance, and self-reflection; whereas Winne and Hadwin (1998) provide a more detailed account involving task definition, goal setting and planning, strategy use, and adaptation. Pintrich (2000) similarly organizes SRL around forethought, monitoring, control, and reflection, while emphasizing regulation across cognitive, motivational, behavioral, and contextual dimensions.

Behavior change theories offer a broader perspective on why and how behaviors change. Some focus on factors that shape behavioral intentions and actions. For example, the Theory of Planned Behavior (Ajzen, 1991) proposes that behavior is influenced by intentions arising from attitudes, perceived social norms, and perceived behavioral control. Related attitude-based accounts also emphasize the combined role of cognitive and affective factors (Zanna and Rempel, 2008). Other theories describe change as a progression over time. The Stages of Change Model (Prochaska et al., 1992), originally developed in health research, characterizes behavior change through a sequence of stages from precontemplation and contemplation to preparation, action, maintenance, and termination. These perspectives complement SRL by explaining not only how learners regulate ongoing learning, but also how changes in behavior may be initiated and sustained.

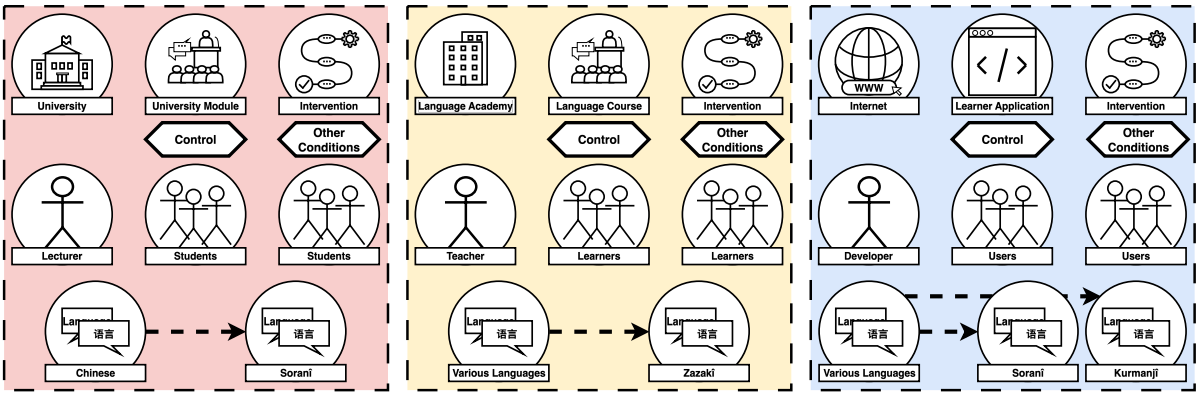
Evaluation of Deployed and Multimodal Language-Learning Systems When evaluating language learning systems, evaluation can consider both learning outcomes and learners' experiences with the technology. Evaluation frameworks in the field of CALL have incorporated learner-related factors when assessing technology-assisted language learning tasks (Aljohani, 2025). This is because the use of game elements may alter the way learners engage in activities, yet it does not necessarily lead to corresponding differences in learning outcomes. Zwitserlood et al. (2022) compared a tablet game vocabulary intervention with a non-game intervention using real objects. They found that learners' vocabulary improved under both conditions, but with no significant difference between the two intervention conditions. Sun et al. (2025) provides further evidence from low-resource language technology. They developed Dia-Lingle, a gamified interface to gather dialectal data with gamified levels of difficulty and active learning. In their usability evaluation, they reported a high level of overall user satisfaction. Qualitative feedback further suggested that varying difficulty levels could support continued participation.

Multimodal technologies have been explored to language learning in endangered and low-resource contexts, as illustrated by a multimodal corpus toolkit developed for Hawaiian language learning (Winkie et al., 2026). Montero-Pérez (2025) points out that research on multimodal language learning tools focuses on learning outcomes, such as vocabulary acquisition and language comprehension, and how learners process different types of input, including images, text, and audio. For example, Guichon and McLornan (2008) found that adding visual information and captions to audio input improved French university students' second language comprehension of authentic audiovisual materials.

3 Experiment Environments

We intend to study three environments, referred to throughout as *E1*, *E2*, and *E3*. They were selected because they place learners under markedly different incentive structures while sharing the common goal of learning a Kurdish language, which allows comparable intervention families to be instantiated in each. Yet, the environments differ in the source languages learner bring, in the Kurdish variety targeted, and in the degree to which learning activity in its entirety can be observed.

E1: University module. Chinese students take an elementary Kurdish module taught by a lecturer as part of a degree programme and learn Central Kurdish. This is the most constrained of the three environments, in that the stakes are high since failure carries academic consequences. Additionally, across multiple sessions per week, the lecturer gives instructions and explanations to then assign tasks to be completed both in class and at home. Traditionally, this type of environment is primarily analogue, with all participants meeting in the same room. We will introduce a digital platform that mediates (at least a selected subset of) the lecturer's assignments, and it is this platform that carries the



(a) *E1*, university module. (b) *E2*, language course. (c) *E3*, learning application.

Figure 2: Instantiation of the interventions across the three environments *E1*, *E2*, and *E3*. In each panel the responsible actor (lecturer, teacher, developer) supplies the learning activities both in a conventional form, which we call control condition, and in a redesigned form which is how our interventions are made.

experimental manipulation. Importantly, all of the students receive the same platform throughout, so the platform itself is held constant across different manipulations. For the pilot study, the class starts in the control condition, in which assignments are presented in a conventional, digitized form. The whole class then moves together into the gamified condition, in which the same types of assignments are presented in a gameful form, and subsequently into the coaching condition, which will make use of the same conventional assignments but with certain modifications surrounding them (see Figure 4). Eventually, the class will work with a combined condition we consider gamified coaching. The platform itself is therefore held constant and is not confounded with the manipulation.

E2: Language course. Learners from various backgrounds enroll in a paid course at a language academy, taught by a teacher in order to learn a Zazakî dialect. Investment here is financial rather than academic, which we expect to produce a different commitment profile from *E1*. This environment already provides a small companion application supporting quiz-based practice alongside the course, which offers a natural site for our planned intervention.

E3: Learning application. The least constrained of the three environments in which users from various backgrounds can work towards a better proficiency of Northern or Central Kurdish. The application is free of charge, users choose which language to study and which components to interact with at any given moment, and participation can be abandoned at any point without consequences. This environment is entirely digital, which makes it easier to introduce our planned interventions, while we have to expect to see the highest attrition due to its unconstrained nature.

In collaboration with the developer, we are currently implemtenting the technical basis for users to get different roles and permissions assigned in order to use the application in different modes. Each mode will hold one of the planned experiment conditions for the users to use during language learning. The application has several gamification features, including experience points (XP), gems, hearts, daily goals, leaderboards, duels, streaks, and a shop. Which parts of the already existing gamification elements we keep and which we need to suppress in order to create a comparable control condition across environments is still object of active discussion.

4 Experimental Design

Across all three environments the study follows a common skeleton which includes one control group, one gamified treatment group, an agent-based treatment group, and a combination of gamified agent-based treatment group. Each group is measured with the same instruments before and after the intervention period.

In each environment, and in cooperation with the teaching or developing party, we introduce alternative conditions that vary the modality of the learning experience while holding the underlying learning content constant. We aim at four conditions per environment which we intend to create as comparable as possible considering the varying nature of each context. The first is the un-

Env.	Setting	Institutional pressure	Financial investment	Voluntariness	Content provider (Learning actor)	Delivery mode
E1	University module	High	Low	Low	Lecturer (Student)	Primarily in-person
E2	Language course	Medium	High / Paid	Medium	Teacher (Learner)	Mixed
E3	Learning application	Low	None / Free	High	Developer (User)	Fully digital

Table 1: Contextual differences across the three learning environments.

modified but digitalized activities, serving as control. The second is a gameful redesign of the same activities, targeting primarily extrinsic motivation through game design elements. The third introduces coaching, via an agent that accompanies learners over time and prompts changes in learning behavior, targeting self-regulation and thus primarily intrinsic motivation. The fourth and last condition combines the agent-based intervention with those gamification elements applied as part of the second condition. This design provides a way to disentangle the observed effects of gamification and agent-based interventions more clearly via a cross-comparison among all four settings.

The resulting design crosses three environments with four condition families, as depicted in Figure 1. Since the environments differ in ways we cannot control, we do not treat them as levels of a factor to be tested against one another but rather perceive each environment to support a within-environment comparison, and the cross-environment contrast to be interpreted qualitatively.

Considering the limited number of students in the first year, we intend to slightly adjust the execution during our initial pilot study. As described in section 3, we will have all students progress through a staggered setup in which everyone experiences the same condition at the same time and in which the described measurements will be done again at each transition between stages.

5 Procedure & Instruments

All groups are measured before and after the intervention period using the same instrument families, as well as when switching from one condition to another. We plan a language proficiency assessment at the start and at the end of the intervention period, and a smaller such assessment between condition intervals. In parallel, we aim to run initial and final questionnaires covering self-regulation, and a

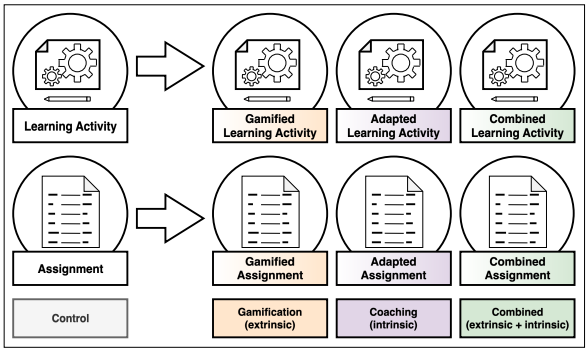


Figure 3: Interventions overview.

self-assessment survey addressing perceptions of gameful design. Since we will introduce digital platforms for serving the different conditions, we can log actual usage such as session count, time spent on task, and feature interactions. By pairing this telemetry with the self-report instruments we will be able to distinguish learners' reported engagement from their real engagement. Taken together, these assessments should allow us to characterise how different modalities affect language learning outcomes and learner experience, and to identify which combinations warrant further investigation.

Coaching Coaching utilizes a stage-based learning agent that supports students throughout the behavior change process to develop productive learning habits and metacognitive routines. It provides timely learning analytics, personalized guidance, and interactive prompts for goal setting, action, reflection, adaptation, and maintenance. In each behavior change stage, it will first provide a primary learning analytics, and then provide students' agency to choose other supporting analytics. The learning analytics could be descriptive, diagnostic, predictive, and prescriptive. During practice, it can also provide on-demand tutor support while keeping the learner focused on their longer-term learning routine. Its name, avatar, and parts of its visual

Gamification element	Role in learning activities	Example implementation	Taxonomy branch
Objectives	Define clear, achievable learning goals for each activity.	Missions, side-quests, milestones	Personal
Point	Provide immediate, quantifiable feedback for completing activities or performing successfully.	Earn experience points (XP)/points for correct answers or completed tasks	Performance
Progression	Make learners' progress toward goals visible over time.	Progress bars, steps, maps; completed/remaining activities	Performance
Acknowledgement	Recognize accumulated achievement and completion.	Badges, achievement markers, trophies	Performance

Table 2: Elements of the planned core gamification package, their roles in learning activities, and how they relate to Toda et al. (2019)'s gamification taxonomy.

styles can be customized by the students to accompany the students through their learning journey.

Interventions on Learner Activities The intervention families are instantiated separately in each environment, since the actors, the delivery channel, and the degree of control over learner activities all differ. In each case the responsible party, that is the lecturer in *E1*, the teacher in *E2*, and the developer in *E3*, retains authorship of the learning content, while the redesign concerns how that content is presented. The redesigns and intervening modifications of the learning material are derived in a collaborative co-creation process together with the researchers. Figure 3 provides an initial overview of the planned interventions, while Figure 2 contrasts the three instantiations.

For the study, each environment will be redesigned around five core learner activities: Vocabulary, Grammar, Alphabet, Listening, and Dialogue. These cover the main forms of language learning across the three environments. The original content and goals will be kept, but the way activities are presented will be reorganized. A shared Core Gamification Package (see Table 2) runs through all three environments. Objectives provide learners with clear goals and a purpose to perform the required tasks. Points provide an extrinsic feedback to the user' actions. Progression provides an extrinsic guidance to the uses of their advance through the activities. Acknowledgement is a kind of extrinsic feedback that praises the players' specific set of actions (Toda et al., 2019). The four elements will stay the same across environments; only their implementation will be adapted to each environment.

Ethics and consent. Aligning with work from participatory research involving native speakers of low-resource languages (Millour, 2019), we base our work on voluntary crowd-sourcing for ethical reasons as well as to ensure quality (Fort et al., 2011), and also for practical reasons of gaining access to the participants (Callison-Burch and Dredze, 2010) at all.

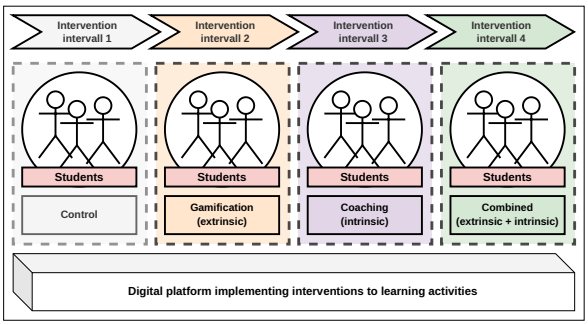


Figure 4: Staged condition sequence where students progress in the shown order during initial pilot study.

6 Conclusion & Outlook

We are currently setting up the experiment start of the first learning environment *E1* (university module) which will run until the end of the year Figure 4. Alongside, we aim to initiate the experiments in the other two environments (language course and learning application) which are currently being designed in more detail.

At the time of writing, we are working in close collaboration with the stakeholders (teaching and developing parties) of the three environments described in this project outline. We are looking for potential sources of funding to support development and execution of the planned later stages of this research work. The prototype which we will use during the earlier pilot study is close to deployment. Together with the Chinese lecturer, we are in the process of first digitizing the assignments for

the first group of students who just entered university and then to introduce our different sets of interventions into their coursework one after another. Thereby, this pilot study allows us to run the first iteration of experiments in the fixed sequence described in section 3 and shown in Figure 4.

Depending on the pace at which we achieve progress, we will then shift our efforts targeting the other two environments (first *E3*, followed by *E2*) from design to implementation and finally execution in order to initiate new experiment iterations in sequence or even in parallel to *E1*. We hope to gain many valuable insights from the initial pilot study as well as from discussing this work at the Multi-LLL workshop to best inform our work on and during the later stages of this research experiment.

7 Limitations

The following describes the identified limitations of this work which should be considered carefully and handled responsibly.

Lack of funding. The same economic challenges which language learners often face, cause research to struggle when securing funding for this kind of research. We rely on voluntary crowdsourcing and have to accept the resulting small and self-selected samples sizes.

Few participants. As a consequence, the group of each condition is bound to be small and potential attrition (e.g. a student dropping out of university) may severely affect the comparability of the final results. While the expected dropout differs across environments and might be a limitation as well as insightful outcomes too, we opt to run a slightly modified pilot study to face this risk.

Cross-sectional self-reported perceptions Self-reporting on motivation and self-regulation is vulnerable to social desirability bias, which is most likely to affect *E1* where the lecturer who delivers the intervention is also the one who assigns grades to the students in the end of a term. In the scope of our intended pilot study, this limitation slightly changes into "repeated self-reported perception", where the same learners now report on each of the four stages in sequence, which may affect their later ratings due to how they rated earlier stages, rather than reflecting each condition independently.

Only few pre- or post-test Only relying on single measurements may not be enough the detect short-lived novelty excitement about a new platform or modality. For this, a delayed follow-up could detect how earlier observed motivational gains slowly evaporate after the novelty of something like a gamified interface wears off. This plays into our somewhat ambitious intention of setting the experiments up such that they can run multiple years in succession.

No causal claims Since we do not intend to treat the three environments as levels of one factor due to how different they are, research question 3 can only be answered qualitatively. Within the environment during our pilot study, the fixed and uncounterbalanced order of the four stages means that differences between stages are also confounded with the passage of time. As learners mature and gain language proficiency which they carry over from earlier stages, any observed stage-to-stage differences cannot clearly be attributed to a given condition in isolation.

Possible recall bias and short or uneven exposure The timing and extent of intervention actually received may vary among participants. In environment *E3*, the learning application, participants can freely choose their learning content and start or stop at any time. Even when paticipants are assigned to the same conditon, the degree to which they are actually exposed to the intervention may differ.

Limited generalizability across Kurdish varieties and interface-language groups Findings from one variety-learner combination may not transfer well to another, which is something that has to be carefully considered when reporting results in later stages of the experiment.

Instrument validity across cultural and linguistic contexts The self-regulation and motivation instruments we intend to apply are drawn largely from Western, well-resourced educational research. Their validity for Kurdish-variety learners, or for constructs that are translated into Kurdish, Chinese, or other interface languages, is not established at this point and may not transfer very well.

Declaration on Generative AI

During the preparation of this work, Asta[1] was used to assist with literature search and to identify potentially relevant publications. Grammarly[2] was used to check grammar and improve the clarity of sentences. The first technical prototype of the digital platform for delivering the different conditions to the students is developed with coding-support by Claude[3]. For resolving LaTeX-related layout issues during the creation of this document, Grok[4] provided useful code-snippets.

Acknowledgements

Christian gratefully acknowledges financial support from GraduSaar, University of Saarland.

References

Roshna Omer Abdulrahman, Hossein Hassani, and Sina Ahmadi. 2019. Developing a fine-grained corpus for a less-resourced language: The case of kurdish. *Preprint*, arXiv:1909.11467.

Emmanuel O. Acquah and Heidi T. Katz. 2020. Digital game-based l2 learning outcomes for primary through high-school students: A systematic literature review. *Computers & Education*, 143:103667.

Sina Ahmadi. 2020. Klpt – kurdish language processing toolkit. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 72–84, Online. Association for Computational Linguistics.

Sina Ahmadi, Zahra Azin, Sara Belelli, and Antonios Anastasopoulos. 2023. Approaches to corpus creation for low-resource language technology: The case of southern kurdish and laki. In *Proceedings of the Second Workshop on NLP Applications to Field Linguistics*, pages 52–63, Dubrovnik, Croatia. Association for Computational Linguistics.

Sina Ahmadi, Daban Jaff, Md Mahfuz Ibn Alam, and Antonios Anastasopoulos. 2024. Language and speech technology for central kurdish varieties. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10034–10045, Torino, Italia. ELRA and ICCL.

Sina Ahmadi and Mariam Masoud. 2020. Towards machine translation for the kurdish language. *Preprint*, arXiv:2010.06041.

[1] https://asta.allen.ai
[2] https://www.grammarly.com/
[3] https://claude.ai/
[4] https://grok.com

Icek Ajzen. 1991. The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, 50(2):179–211.

Sdiq Rahman Ali and Naz Ahmad Saeed. 2025. Integrating literary topics in textbooks with e-learning. *Sulaimani Journal for Humanities*, 1(1):209–231.

Nouf J. Aljohani. 2025. Technology-mediated task-based language teaching: A call evaluation framework and its pedagogical implications. *Computer Assisted Language Learning*, 38(3):638–667.

Sara Belelli. 2021. *The Laki Variety of Harsin: Grammar, Texts, Lexicon*. Number 2 in Bamberg Studies in Kurdish Linguistics. University of Bamberg Press, Bamberg.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 1–12, Los Angeles. Association for Computational Linguistics.

Carol A. Chapelle. 2001. *Computer Applications in Second Language Acquisition*. Cambridge Applied Linguistics. Cambridge University Press.

Mordechai Chaziza. 2024. The kurdish question in china's middle east foreign policy - proquest.

Shabnam Dadparvar and Ismael Shams. 2024. Oriental whispers: Unveiling chinese cultural influences in classical kurdish literature. *Kurdish Studies*, 12(1).

Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart E. Nacke. 2011. From game design elements to gamefulness: defining "gamification". In *International Conference on Entertainment and Media in the Ubiquitous Era*.

Christo Dichev and D. Dicheva. 2017. Gamifying education: what is known, what is believed and what remains uncertain: a critical review. *International Journal of Educational Technology in Higher Education*, 14.

Mahîr Dogan. 2024. 8 problems in zazakî nomenclature. In *Gorani in Its Historical and Linguistic Context*, pages 249–276. De Gruyter Mouton.

Karën Fort, Gilles Adda, and K. Bretonnel Cohen. 2011. Last words: Amazon mechanical turk: Gold mine or coal mine? *Computational Linguistics*, 37(2):413–420.

Nicolas Guichon and Sinead McLornan. 2008. The effects of multimodality on l2 learners: Implications for call resource design. *System*, 36(1):85–93.

Geoffrey Haig and Ergin Öpengin. 2018. Kurmanji kurdish in turkey: Structure, varieties, and status. *Linguistic minorities in Turkey and Turkic-speaking minorities of the periphery*, pages 157–229.

Hossein Hassani. 2018. Blark for multi-dialect languages: Towards the kurdish blark. *Language Resources and Evaluation*, 52(2):625–644.

Kashif Ishaq, Nor Azan Mat Zin, Fadhilah Rosdi, Muhammad Jehanghir, Samia Ishaq, and Adnan Abid. 2021. Mobile-assisted and gamification-based language learning: A systematic literature review. *PeerJ. Computer Science*, 7:e496.

Dastan Jasim. 2024. Fixers, fieldwork, and precarity: The postcoloniality of western fieldwork on isis in kurdistan. *South Atlantic Quarterly*, 123(4):755–777.

Ioana Jivet, Maren Scheffel, Marcus Specht, and Hendrik Drachsler. 2018. License to evaluate: Preparing learning analytics dashboards for educational practice. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, LAK '18, pages 31–40, New York, NY, USA. Association for Computing Machinery.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the nlp world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Zina Kamal and Hossein Hassani. 2020. Towards kurdish text to sign translation. In *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*, pages 117–122, Marseille, France. European Language Resources Association (ELRA).

Zhanni Luo. 2023. The effectiveness of gamified tools for foreign language learning (fll): A systematic review. *Behavioral Sciences*, 13.

Alice Millour. 2019. Getting to know the speakers: A survey of a non-standardized language digital use. In *9th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Poznań, Poland.

Masoud Mohammadirad. 2024. Zagros region: The kurdish-gorani continuum. *Post-predicate elements in the Western Asian Transition Zone*, pages 245–279.

Maribel Montero-Pérez. 2025. Chapter 6. the impact of multimodal input tools in instructed second language acquisition (and beyond). In Shawn Loewen, Frederick J. Poole, Hyun-Bin Hwang, and Matthew D. Coss, editors, *Technology and Instructed Second Language Acquisition: Connecting Research and Pedagogy*, Language Learning & Language Teaching, pages 139–162. John Benjamins Publishing Company.

Peshmerge Morad, Sina Ahmadi, and Lorenzo Gatti. 2024. Part-of-speech tagging for northern kurdish. In *Proceedings of the Joint Workshop on Multiword Expressions and Universal Dependencies (MWE-UD) @ LREC-COLING 2024*, pages 70–80, Torino, Italia. ELRA and ICCL.

Amir Muhammad Muhammadamin. 2025. The role of educational linguistics in kurdish language learning: The basic stage as a sample. *Zanco Journal of Human Sciences*, 29(5):211–229.

Paul R. Pintrich. 2000. Chapter 14 - the role of goal orientation in self-regulated learning. In Monique Boekaerts, Paul R. Pintrich, and Moshe Zeidner, editors, *Handbook of Self-Regulation*, pages 451–502. Academic Press, San Diego.

James O. Prochaska, Carlo C. DiClemente, and John C. Norcross. 1992. In search of how people change: Applications to addictive behaviors. *American Psychologist*, 47(9):1102–1114.

Baba Resul Nurî Resul and Amir Muhammad Muhammadamin. 2015. 'se wazî gutnewey waney jman u edebî kurdî' (teaching methods of kurdish language and literature).

Michael Sailer and Lisa Homner. 2020. The gamification of learning: A meta-analysis. *Educational Psychology Review*, 32(1):77–112.

Jaffer Sheyholislami. 2021. The history and development of literary central kurdish. In Hamit Bozarslan, Cengiz Gunes, and Veli Yadirgi, editors, *The Cambridge History of the Kurds*, pages 633–662. Cambridge University Press, Cambridge.

Jiugeng Sun, Rita Sevastjanova, Sina Ahmadi, Rico Sennrich, and Mennatallah El-Assady. 2025. Dialingle: A gamified interface for dialectal data collection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 148–158, Vienna, Austria. Association for Computational Linguistics.

Givi Tavadze. 2019. Spreading of the kurdish language dialects and writing systems used in the middle east. *Bulletin of the Georgian National Academy of Sciences*, 13(1):170–174.

Armando M. Toda, Ana C. T. Klock, Wilk Oliveira, Paula T. Palomino, Luiz Rodrigues, Lei Shi, Ig Bittencourt, Isabela Gasparini, Seiji Isotani, and Alexandra I. Cristea. 2019. Analysing gamification elements in educational environments using an existing gamification taxonomy. *Smart Learning Environments*, 6(1):16.

Rixt van Dongera, Cor van der Meer, and Richt Sterk. 2017. Research for cult committee – minority languages and education: Best practices and pitfalls. Technical report, European Parliament, Policy Department for Structural and Cohesion Policies, Brussels.

Joseph Winkie, Michol Miller, and Winston Wu. 2026. Developing a hawaiian corpus toolkit for data-driven language learning. In *Proceedings of the Ninth Workshop on the Use of Computational Methods in the Study of Endangered Languages (ComputEL-9)*, pages 167–176, San Diego, California, USA. Association for Computational Linguistics.

Philip H. Winne and Allyson F. Hadwin. 1998. Studying as self-regulated learning. In *Metacognition in Educational Theory and Practice*. Routledge.

World Bank. 2019. *Belt and Road Economics: Opportunities and Risks of Transport Corridors*. Washington, DC: World Bank.

Mark Zanna and John Rempel. 2008. Attitudes: A new look at an old concept. *Attitudes: Their structure, function, and consequences*.

Songcun Zhang and Zuwati Hasim. 2023. Gamification in efl/esl instruction: A systematic review of empirical research. *Frontiers in Psychology*, 13:1030790.

Xueqing Zhang. 2026. Is the future of the kurdish language secured in the kurdistan region of iraq? insights from kurdish teaching in erbil. *Asian Affairs*, 57(1):122–141.

Barry J. Zimmerman. 2000. Chapter 2 - attaining self-regulation: A social cognitive perspective. In Monique Boekaerts, Paul R. Pintrich, and Moshe Zeidner, editors, *Handbook of Self-Regulation*, pages 13–39. Academic Press, San Diego.

Rob Zwitserlood, Marjan ter Harmsel, Johanna Schulting, Karin Wiefferink, and Ellen Gerrits. 2022. To game or not to game? efficacy of using tablet games in vocabulary intervention for children with dld. *Applied Sciences*, 12(3):1643.

Clean Evaluations on Contaminated Visual Language Models

Hongyuan Lu[♡*], **Shujie Miao**[◇*], **Wai Lam**[♠]
♡FaceMind
♠The Chinese University of Hong Kong
◇National University of Singapore
hongyuanlu@outlook.com

Abstract

How to evaluate large language models (LLMs) cleanly has been established as an important research era to genuinely report the performance of possibly contaminated LLMs. Yet, how to cleanly evaluate the visual language models (VLMs) is an under-studied problem. We propose a novel approach to achieve such goals through data augmentation methods on the visual input information. We then craft a new visual clean evaluation benchmark with thousands of data instances. Through extensive experiments, we found that the traditional visual data augmentation methods are useful, but they are at risk of being used as a part of the training data as a workaround. We further propose using BGR augmentation to switch the colour channel of the visual information. We found that it is a simple yet effective method for reducing the effect of data contamination and fortunately, it is also harmful to be used as a data augmentation method during training. It means that it is hard to integrate such data augmentation into training by malicious trainers and it could be a promising technique to cleanly evaluate visual LLMs. Our code, data, and model weights will be released upon publication.

1 Introduction

With the rapid advancement of LLMs (Hou et al., 2025; Yang et al., 2025; Lu et al., 2026), VLMs represent a critical milestone in the journey towards artificial intelligence (Fan et al., 2023; Zhao et al., 2023). VLMs extend the capabilities of textual LLMs by integrating cross-modal architectures such as CLIP (Radford et al., 2021), allowing for the interpretation and generation of multi-modal content across both text and images (Cao et al., 2024; Huang et al., 2024). Moreover, prior research has established numerous benchmarks to evaluate the capabilities of VLMs from various dimensions (Fan et al., 2024; Fu et al., 2023a,b).

However, the reliability of these VLM benchmarks is at risk of being undermined by a widely recognized issue in the LLM evaluation: data contamination. Data contamination occurs when the benchmark data overlaps with a model's training data, causing the model's performance metrics to be artificially inflated and not truly representative of its generalization ability (Magar and Schwartz, 2022; Dong et al., 2024). Researchers have developed various techniques for LLMs to mitigate these issues, including advanced detection methods (Dong et al., 2024; Zhang et al., 2024), proactive prevention strategies (Jacovi et al., 2023; Zhu et al., 2024a; Fan et al., 2024), and genuinely evaluating the capabilities of LLMs via input textual rephrasing (Zhu et al., 2024b).

While much attention has been given to the problem of data contamination for LLMs, the ones for VLMs remains under-explored. We propose a new clean evaluation benchmark for VLMs and a novel method to genuinely evaluate VLMs' capabilities by operation on the visual input.

Our benchmark comprises thousands of carefully curated data which are newly released and collected on the internet to ensure that the evaluation process remains free from data contamination.

We found that traditional data augmentation such as flipping and rotation on the image can help with the problem of data augmentation, making the performance closer to the uncontaminated model, they are yet at risk of being used as a part of the training techniques. We propose a new method, BGR channel swapping to the visual input, which we fortunately found could not be used as a training technique and can degrade the performance.
We make the following three key contributions:

- We establish a new visual clean evaluation benchmark for VLMs.
- We propose to use data augmentation methods to reveal the true capabilities of VLMs and

[*] Equal Contribution.

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 94–99
November 9, 2026 ©2026 Association for Computational Linguistics



Q: What is the central element in the background of the image?
A: Statue.
Q: How many columns are visible in the background of the image?
A: Four.
Q: What is the direction of the sunlight in the image?
A: Diagonal.

Q: What type of setting is shown in the image?
A: Garden.
Q: What is the main object being held in the image?
A: Gun.
Q: Are there any flowers visible in the image?
A: Yes.

Figure 1: Two examples from the dataset. Pairs of images with corresponding questions and answers.

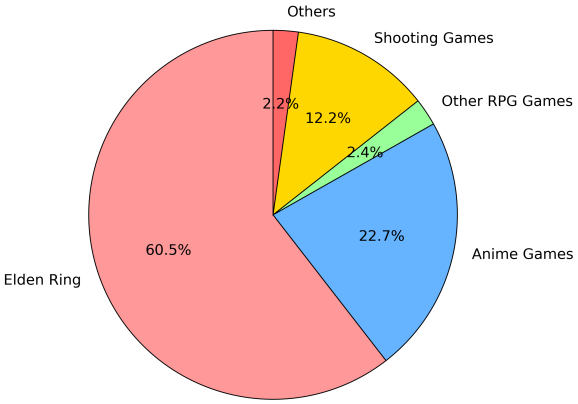


Figure 2: Distribution of the game genres in our collected dataset. 'Elden Ring' (605 instances), 'Anime Games' (227 instances, including Genshin Impact and Honkai: Star Rail), 'Other RPG Games' (24 instances, with titles like Dungeon I& Fighter), 'Shooting Games' (122 instances, featuring GTA V, Valorant, and Delta Force), and 'Others' (22 instances, including Palworld and League of Legends).

reduce the impact of data contamination.

- We identify BGR channel swapping as a robust method for clean evaluation and preventing exploitation. Fortunately, further analysis reveals that BGR data augmentation is harmful and cannot be used during training.

2 Dataset

The dataset collected for this study originates from the well-known gaming guide website Gamersky,[1] from which we collected 1,000 high-quality images. These images capture key in-game scenes and contain complex visual information, including objects, text, and scene elements. Using GPT-4o's multi-modal understanding capabilities [2], we generated 1 to 3 question-answer (QA) pairs for each image (see Figure 1). All generated QA pairs underwent careful manual review and correction to ensure accuracy and relevance. The final dataset comprises 1,000 images and 2,561 rounds of dialogue. Furthermore, the distribution of game types represented in our dataset is illustrated in Figure 2.

Importantly, the selected image data, collected from June 20 to June 25, 2024, was published after the release of the models used in this study. This approach mitigates the risk of data contamination by preventing the premature inclusion of our benchmark in pre-training datasets. Consequently, our dataset provides an uncontaminated benchmark for the evaluation of VLMs.

We partition our dataset into a training set (90%) and a test set (10%). We use Low-Rank Adaptation (LoRA) for fine-tuning VLMs (Hu et al., 2021).

[1] https://www.gamersky.com/handbook/game/gl/
[2] https://openai.com/research/gpt-4o

3 Methodology

We train a baseline model, denoted as M_1, on the training set only. To simulate data contamination, we construct a contaminated training set by replacing a subset of the original training samples with the test set (this keeps the total size of the training set the same). This contaminated set is then used to train a second model, M_2. Both M_1 and M_2 are trained for multiple epochs to simulate different levels of generalization and data contamination. We expect higher performance on both models with more epochs, and M_2 is contaminated, so it should report higher scores consistently than M_1.

To evaluate them fairly, we apply data augmentation techniques such as multi-angle rotations, and horizontal and vertical flips to transform them while persevering their semantics. For each test sample $(x, z) \in D_{\text{test}}$, where x is the text input and z is the image, we generate augmented samples:

$$(x, z') = (x, t(z)), \quad t \in T$$

Here, t represents the functions of transformations. x remains unchanged while z undergoes transformation, producing z'.

This augmentation process enables an assessment of the model's visual robustness and adaptability by testing its performance on the transformed images while keeping the textual input constant. We formalize the prediction process on these

augmented inputs as:

$$P(y|x, z') = M(x, z'), \quad z' = t(z), \quad t \in T \tag{1}$$

In Equation 1, M represents the model (either M_1 or M_2), t is a transformation function from the set T, z is the original image input, z' is the transformed image, and y is the model's prediction.

We then use y as the calibrated output which fairly represents the models' performance.

4 Experimental Setup

VLMs In our experimental setup, we evaluated two state-of-the-art VLMs: bunny-4B and internvl2.0-2B. These models were selected based on their strong performance in MME (multi-modal Evaluation) benchmarks (Fu et al., 2023a), particularly excelling in tasks involving existence and position perception, OCR, common reasoning, and numerical calculation.

Baselines We use classical data augmentation methods such as multi-angle rotations, and horizontal and vertical flips to evaluate VLMs fairly.

Metrics We employed ROUGE-1, ROUGE-2 (Lin, 2004), and BLEU (Papineni et al., 2002) metrics, which collectively measure linguistic overlap and semantic similarities between generated and reference texts, providing a comprehensive evaluation of the models' outputs.

Machine Environment Both models were fine-tuned using LoRA on 2 RTX 4090 GPUs, each with 24GB of memory. For bunny-4B, we used the phi-2 architecture with LoRA (rank 128, alpha 256), while internvl2.0-2B employed parameter freezing and LoRA (rank 16) for the language model component. We utilized mixed-precision training (bfloat16) and gradient checkpointing for memory efficiency. Both models used an effective batch size of 16, achieved through gradient accumulation, and employed cosine learning rate schedules with warmup. To simulate different contamination scenarios, we trained each model for both 5 and 10 epochs, using learning rates of 5e-4 for bunny-4B and 4e-5 for internvl2.0-2B.

5 Results

5.1 Main Results

Table 1 illustrates the impact of simulated data contamination on model performance. For both Bunny and InternVL, we observed a substantial increase in evaluation metrics when trained on contaminated data. For instance, Bunny's BLEU score increased from 0.0816 to 0.1366 at 5 epochs, while ROUGE-1 and ROUGE-2 scores also showed marked improvements. InternVL exhibited a similar pattern, with contamination raising its BLEU score from 0.1047 to 0.1499. These inflated results suggest that data contamination can lead to an overestimation of a model's true performance by allowing it to access evaluation data during training. This highlights the need for a reliable method to counteract this inflation and offer a more accurate assessment.

Model	Data	BLEU	ROUGE-1	ROUGE-2
Bunny (Epoch 5)	Original	0.0816	0.5319	0.0352
	Conta	0.1366 ↑	0.7482 ↑	0.1140 ↑
Bunny (Epoch 10)	Original	0.0792	0.5248	0.0341
	Conta	0.1413 ↑	0.7506 ↑	0.1174 ↑
InternVL (Epoch 5)	Original	0.1047	0.6250	0.0670
	Conta	0.1499 ↑	0.7867 ↑	0.1238 ↑
InternVL (Epoch 10)	Original	0.1001	0.6114	0.0739
	Conta	0.1748 ↑	0.8973 ↑	0.1771 ↑

Table 1: Performance metrics for Bunny and InternVL models on original and Conta (contaminated) data

To address this challenge, we test various data augmentation techniques as part of our clean evaluation process, as shown in Table 2. Applying augmentations like rotations, flips, and our proposed BGR channel swaps to the test data helped reveal the true performance of contaminated models. For example, the BLEU score of the contaminated Bunny model dropped from 0.1366 to 0.1030 with horizontal flipping and further to 0.0796 with 150-degree rotation. These drops in performance illustrate the model's vulnerability when faced with even slight modifications, further underscoring the harmful effects of contamination. Importantly, the performance of contaminated models under these augmentations consistently fell between that of the original uncontaminated models and the fully contaminated versions, validating the effectiveness of our clean evaluation method in restoring a more accurate reflection of model capabilities.

As the severity of augmentation increased, the model's robustness weakened, with larger rotations causing greater performance degradation. While small rotations showed only minor declines, extreme transformations like 150-degree or 180-degree rotations led to substantial drops, exposing the contaminated model's fragile generalization.

Models	Original Model			Contaminated Model		
	BLEU	**ROUGE-1**	**ROUGE-2**	**BLEU**	**ROUGE-1**	**ROUGE-2**
Target Performance	0.1047	0.6250	0.0670	0.1047	0.6250	0.0670
w/o Data Aug.	-	-	-	0.1499	0.7867	0.1238
Vertical flip	0.0817	0.4982	0.0212	0.0974	0.5674	0.0439
Horizontal flip	0.0800	0.5169	0.0057	**0.1030**	**0.5983**	0.0341
Rotate 30°	0.0960	**0.5825**	0.0398	0.1144	0.6506	**0.0663**
Rotate 60°	0.0822	0.4929	0.0246	0.0927	0.5469	0.0417
Rotate 90°	0.0810	0.4996	0.0307	0.0977	0.5704	0.0534
Rotate 120°	0.0724	0.4437	0.0170	0.0808	0.4788	0.0227
Rotate 150°	0.0659	0.4275	0.0057	0.0796	0.4881	0.0170
Rotate 180°	0.0716	0.4456	0.0057	0.0844	0.5086	0.0360
BGR	**0.1012**	0.5750	**0.0783**	0.1368	0.7081	0.1108

Table 2: Performance comparison of InternVL model (Epoch 5) under various data augmentation techniques. The table presents BLEU, ROUGE-1, and ROUGE-2 scores for both the original and contaminated models. Data augmentation methods include vertical and horizontal flips, rotations (30°, 60°, 90°, 120°, 150°, 180°), and our proposed BGR colour space conversion. The closer the models are to the target performance, the better they are.

Training Condition	BLEU	ROUGE-1	ROUGE-2
Original	0.1047	0.6250	0.0670
Mixed BGR Data	0.1041	0.6082	0.0670

Table 3: Comparison of InternVL model performance with and without mixed data augmentation (5 epochs)

5.2 BGR swapping

Notably, we found that using our proposed BGR augmentation apparently restores the performance, where it scores the most of the metrics among all.

Also, during our experiments, we discovered that the BGR channel-swapping method exhibited particularly strong resistance to potential manipulation. As shown in Table 3, incorporating BGR augmentation into the training data does not lead to a significant increase in model performance for InternVL at 5 epochs. The BLEU scores for the original model (0.1047) and the model trained with mixed data augmentation (0.1041) are nearly identical, with similar trends observed for ROUGE-1 and ROUGE-2 scores. This result is crucial as it indicates that BGR augmentation is particularly useful and can effectively reveal a contaminated model's true capabilities while being difficult to exploit through training data manipulation.

6 Conclusions and Related Work

Recent advancements in LLMs have highlighted the critical issue of data contamination in natural language processing. While significant progress has been made in addressing this challenge for text-based LLMs, the problem remains understudied for VLMs. Notable contributions include CDD and TED (Dong et al., 2024), which detect contamination through output distribution analysis and mitigate its impact on evaluation, and Clean-Eval (Zhu et al., 2024b), which employs neural-based paraphrasing to generate semantically equivalent but surface-level different expressions of potentially contaminated data.

Our research extends these clean evaluation techniques to the visual domain, introducing a novel approach to mitigate data contamination in VLMs through visual data augmentation methods. We collect and present a new clean evaluation benchmark for VLMs and propose various data augmentation techniques, with a novel method called BGR channel swapping emerging as a particularly robust method for clean evaluation. This method demonstrates resistance to exploitation during training, effectively reducing the performance gap between contaminated and uncontaminated models.

This work significantly advances the field of VLMs evaluation, enhancing transparency and reliability in assessing model capabilities. As multi-

modal AI continues to evolve, such clean evaluation methods will play a crucial role in ensuring the integrity of model development and deployment. Future research may focus on extending these approaches to other modalities and investigating their potential to improve the robustness and generalization capabilities of VLMs beyond clean evaluation. Our resources will be released upon publication.

Limitations

This paper has studied visual data contamination on question answering. Further extending the scope of tasks can enhance the usefulness of the method.

Ethics Statement

We honour and support the ARR Code of Ethics. We spot no obvious ethical issues in this paper.

References

Lele Cao, Valentin Buchner, Zineb Senane, and Fangkai Yang. 2024. Introducing GenCeption for multimodal LLM benchmarking: You may bypass annotations. In *Proceedings of the 4th Workshop on Trustworthy Natural Language Processing (TrustNLP 2024)*, pages 196–201, Mexico City, Mexico. Association for Computational Linguistics.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 12039–12050, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Lizhou Fan, Wenyue Hua, and Xiang Li. 2024. Nphardeval4v: A dynamic reasoning benchmark of multimodal large language models. https://synthical.com/article/8cc2ab7b-0527-4a90-912c-9bab53ecc10a. *Preprint*, arXiv:2403.01777.

Lizhou Fan, Lingyao Li, Zihui Ma, Sanggyu Lee, Huizi Yu, and Libby Hemphill. 2023. A bibliometric review of large language models research from 2017 to 2023. *ACM Transactions on Intelligent Systems and Technology*.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. 2023a. Mme: A comprehensive evaluation benchmark for multimodal large language models. *ArXiv*, abs/2306.13394.

Chaoyou Fu, Renrui Zhang, Zihan Wang, Yubo Huang, Zhengye Zhang, Longtian Qiu, Gaoxiang Ye, Yunhang Shen, Mengdan Zhang, Peixian Chen, Sirui Zhao, Shaohui Lin, Deqiang Jiang, Di Yin, Peng Gao, Ke Li, Hongsheng Li, and Xing Sun. 2023b. A challenger to gpt-4v? early explorations of gemini in visual expertise. *ArXiv*, abs/2312.12436.

Ruijie Hou, Yueyang Jiao, Hanxu Hu, Yingming Li, Wai Lam, Huajian Zhang, and Hongyuan Lu. 2025. LNE-blocking: An efficient framework for contamination mitigation evaluation on large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 3512–3528, Suzhou, China. Association for Computational Linguistics.

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

Dawei Huang, Chuan Yan, Qing Li, and Xiaojiang Peng. 2024. From large language models to large multimodal models: A literature review. *Applied Sciences*, 14(12).

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. 2023. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5084, Singapore. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hongyuan Lu, Zixuan Li, Zefan Zhang, Bowen Cao, and Wai Lam. 2026. Adam's law: Textual frequency law on large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4079–4105, San Diego, California, United States. Association for Computational Linguistics.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 157–165, Dublin, Ireland. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Hao Yang, Hongyuan Lu, Xinhua Zeng, Yang Liu, Xiang Zhang, Haoran Yang, Yumeng Zhang, Shan Huang, Yiran Wei, and Wai Lam. 2025. Stephanie: Step-by-step dialogues for mimicking human interactions in social conversations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 153–166, Albuquerque, New Mexico. Association for Computational Linguistics.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Yang, and Hai Li. 2024. Min-k%++: Improved baseline for detecting pre-training data from large language models. *arXiv preprint arXiv:2404.02936*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Z. Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jianyun Nie, and Ji rong Wen. 2023. A survey of large language models. *ArXiv*, abs/2303.18223.

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2024a. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations*.

Wenhong Zhu, Hongkun Hao, Zhiwei He, Yun-Ze Song, Jiao Yueyang, Yumeng Zhang, Hanxu Hu, Yiran Wei, Rui Wang, and Hongyuan Lu. 2024b. CLEAN–EVAL: Clean evaluation on contaminated large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 835–847, Mexico City, Mexico. Association for Computational Linguistics.

Audio Representation Learning and Downstream Task Modeling: A Survey

Kaihong Wang[1], Liang He[1,2†],
[1]Department of Electronic Engineering, and Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China
[2]School of Computer Science and Technology, School of Intelligence Science and Technology, Xinjiang University, Urumqi 830017, China
{wkh25, heliang}@mails.tsinghua.edu.cn

Abstract

Audio representation learning has revolutionized acoustic intelligence, enabling audio foundation models to generalize and perform well across various downstream tasks. To better characterize the paradigm shift in general representation extraction and modeling, this survey presents a unified decoupled framework separating audio processing into audio-to-feature representation learning and feature-to-target downstream modeling. We systematically review modeling domains, representation forms, architectures, and pre-training paradigms, while categorizing five integration mechanisms and downstream modeling paradigms. Furthermore, we summarize representative systems and discuss key trade-offs, providing practical guidelines for designing next-generation audio foundation models.

1 Introduction

Raw audio signals entangle complex, dynamic, and diverse acoustic factors. While historical research bottlenecked representational capacity by relying on rigid hand-crafted features like MFCCs (Davis and Mermelstein, 1980; Reynolds and Rose, 1995) and Log-Mel spectrograms (Cotton and Ellis, 2011; Rakotomamonjy and Gasso, 2015) or heuristic mixtures (Scheirer and Slaney, 1997; Lu et al., 2002), data-driven deep learning (Hinton et al., 2012; Abdel-Hamid et al., 2014) and self-supervised learning (SSL) (Baevski et al., 2020; Hsu et al., 2021; Chen et al., 2023) have redefined this domain. By learning universal representations directly from unlabeled data, pre-trained models now support diverse downstream tasks, ranging from speech language modeling (Borsos et al., 2023a; Chen et al., 2024a) and low-latency dialogue (Défossez et al., 2024; Roy et al., 2026) to unified generation (Chen et al., 2025b; Tian et al., 2026d) and diffusion-based understanding (Zhou et al., 2026a,b).

Current audio representation extraction paradigm has shifted from task-specific networks to large-scale pre-trained models capturing general representations. These robust representations diversify mapping mechanisms and inspire unified architectures, such as discrete representations with autoregressive prediction for audio language modeling (Borsos et al., 2023a; Chen et al., 2024a) and low-latency voice interaction (Défossez et al., 2024; Roy et al., 2026), unified understanding and generation via joint prediction (Chen et al., 2025b; Tian et al., 2026d), and diffusion language models for audio understanding (Zhou et al., 2026a,b). However, existing task-centric surveys (Xie et al., 2025b; Yeow et al., 2025; Guo et al., 2026b; Zhu et al., 2025; Retkowski et al., 2025) or those focusing solely on autoregressive speech-language models (Yang et al., 2025; Peng et al., 2025a) neglect representation injection and downstream modeling, failing to capture this paradigm shift. To address this, we propose a decoupled perspective of representation and modeling that first delineates representation space construction and evaluation before discussing downstream adaptation strategies across tasks.

Our main contributions are threefold: (a) we establish a decoupled taxonomy that untangles audio foundation model research into "audio-to-feature" representation extraction (Section 2) and "feature-to-target" downstream modeling (Section 3); (b) we systematically categorize five representation injection mechanisms (Section 3.1) and analyze their downstream task suitability, providing clear architectural guidelines for next-generation systems; and (c) we provide a comprehensive, multi-dimensional taxonomy of frontier models across three appendices: representation models (Appendix D), pre-trained foundation models (Appendix F.1), and fine-tuned downstream models (Appendix F.2). This decoupled lens offers actionable blueprints for modular audio architectures.

Proceedings of the 1st Workshop on Multi-modality and Multi-lingual Language Learning, pages 100–130
November 9, 2026 ©2026 Association for Computational Linguistics

2 Audio Representation Learning

An ideal universal audio representation efficiently generalizes across diverse downstream tasks under limited supervision (Turian et al., 2021). To systematically deconstruct this field, this chapter analyzes: (1) **audio modeling domains** of raw signals (Section 2.1), (2) **representation form** of audio features (Section 2.2), (3) **architectural designs** of mainstream audio representation models (Section 2.3), and (4) data-driven **pre-training paradigms** (Section 2.4). Additionally, the appendix provides: (5) **evaluation methods** (Section B) and (6) a summary of mainstream **audio representation models** (Section D).

2.1 Audio Modeling Domain

2.1.1 Time Domain

Time-domain modeling refers to directly ingesting sampled raw acoustic waveforms into neural networks without any transform-domain operations. By preserving complete physical signals, such as phase and transient details, this approach is highly suited for tasks requiring sample-level fidelity and phase-sensitive reconstruction such as speech separation (Bahmaninezhad et al., 2019; Luo and Mesgarani, 2019; Luo et al., 2020), high-fidelity audio generation (van den Oord et al., 2016), and universal representation extraction (Yuksel et al., 2025). Eliminating transform-related latency also natively supports ultra-low latency applications (Défossez et al., 2023). However, raw waveforms result in extremely long sequences, posing severe computational challenges for global attention mechanisms. Furthermore, lacking explicit harmonic inductive biases, pure time-domain networks often face optimization difficulties in signal-to-signal tasks (Nossier et al., 2020; Lin et al., 2024). To address this, feature extraction front-ends commonly leverage deep strided 1D-CNNs (Défossez et al., 2023) or parameterized physical filterbanks (Ravanelli and Bengio, 2018) to reduce dimensionality and capture local temporal continuity.

2.1.2 Time-Frequency Domain

To capture temporal evolution, mainstream audio tasks rely on time-frequency (T-F) modeling, where the Short-Time Fourier Transform (STFT) projects 1D waveforms into 2D spectrograms. T-F representations compress the temporal dimension, reducing computational complexity (Radford et al., 2023; Chen et al., 2023), and allow the seamless inheritance of computer vision architectures using the "audio-as-image" paradigm (Gong et al., 2021; Huang et al., 2022; Tuncay et al., 2025). Backed by strong frequency inductive biases, T-F modeling is well-suited for spectrum-dependent tasks like Audio Event Detection (Hertel et al., 2016; Kong et al., 2020b), Speech Emotion Recognition (Penumajji, 2025; Cao et al., 2025), and Audio Deepfake Detection (Pachon and Ballesteros, 2026; Hamza et al., 2022). Nevertheless, T-F modeling faces physical and mathematical limitations. First, STFT suffers from a rigid time-frequency resolution trade-off, typically mitigated by multi-resolution STFT losses (Kong et al., 2020a; Yuan et al., 2025) or learnable front-ends (Zeghidour et al., 2021; Leiber et al., 2025). Second, phase wrapping within $(-\pi, \pi]$ disrupts gradient continuity (Engel et al., 2019; Ai and Ling, 2023). Consequently, standard models abandon phase modeling (Radford et al., 2023; Huang et al., 2022), or overcome this bottleneck through adversarial training with time-domain discriminators (Zeghidour et al., 2022; Défossez et al., 2023; Zhang et al., 2024) or direct prediction of the complex spectrum's real and imaginary parts (Feng et al., 2025; Wang et al., 2023).

2.2 Audio Representation Form

2.2.1 Continuous Representation

Continuous representations consist of high-dimensional dense float vector sequences extracted by an audio encoder at each time step. By preserving fine-grained acoustic details, they systematically excel on speech understanding benchmarks (Wang et al., 2025a). Consequently, mainstream autoregressive understanding models (e.g., Qwen2-Audio (Chu et al., 2024), SALMONN (Tang et al., 2024), Audio Flamingo 3 (Ghosh et al., 2025)) and diffusion language models (e.g., the DIFFA series (Zhou et al., 2026a,b)) widely adopt them. Integrating these continuous sequences into modern large language models (LLMs) typically requires custom adapter modules and alignment training for feature mapping and downsampling. On the generation side, continuous representations naturally align with continuous-space diffusion and flow matching algorithms (Le et al., 2023; Hung et al., 2026; Tian et al., 2026d). For autoregressive generation, systems can leverage regression-based objectives to predict continuous target vectors directly, which subsequently condition non-autoregressive models (Liu et al., 2024b; Rouard et al., 2026). Alterna-

tively, methods can map generated discrete tokens into continuous vectors to guide subsequent acoustic reconstruction (Du et al., 2024c, 2025).

2.2.2 Discrete Representation

Discrete representations compress high-dimensional continuous signals into sequences of discrete symbols. These are primarily categorized into acoustic tokens, generated by neural audio codecs (van den Oord et al., 2017; Kumar et al., 2023; Pepino et al., 2025; Gong et al., 2026a) via residual vector quantization (RVQ) for high-fidelity reconstruction, and semantic tokens, derived from applying K-means to self-supervised representations like HuBERT (Hsu et al., 2021) and wav2vec 2.0 (Baevski et al., 2020) to capture phonetic information (Borsos et al., 2023a,b). For downstream modeling, discrete tokens enable autoregressive language models to perform next-token prediction via vocabulary expansion (Borsos et al., 2023a; Agostinelli et al., 2023; Shi et al., 2025; Arora et al., 2025), or can be modeled via classification-based discrete diffusion (Borsos et al., 2023b; Yang et al., 2023b; Long et al., 2026). This discretization unifies text and audio processing, endowing speech synthesis systems with remarkable zero-shot and in-context capabilities (Chen et al., 2025c, 2024a). To optimize the compression ratio, recent architectures apply semantic distillation during codec training to preserve semantic content and acoustic quality at low frame rates (Défossez et al., 2024).

2.3 Audio Representation Models and Downstream Components

Figure 1 summarizes audio representation architectures and downstream components.

2.3.1 Encoder-only

With no learnable decoder, this architecture focuses on generating dense, decoupled, discriminative latent representations, and its downstream head is typically a lightweight linear classifier. Representative Encoder-only models include AST (Gong et al., 2021), HuBERT (Hsu et al., 2021), BEATs (Chen et al., 2023), MERT (Li et al., 2024c), USAD (Chang et al., 2025), and CAT (Han et al., 2026). Furthermore, this paradigm extends to multimodal scenarios, where models such as CLAP (Elizalde et al., 2023) and SLAP (Siamese) (Guinot et al., 2025) employ a dual-encoder structure to achieve cross-modal alignment.

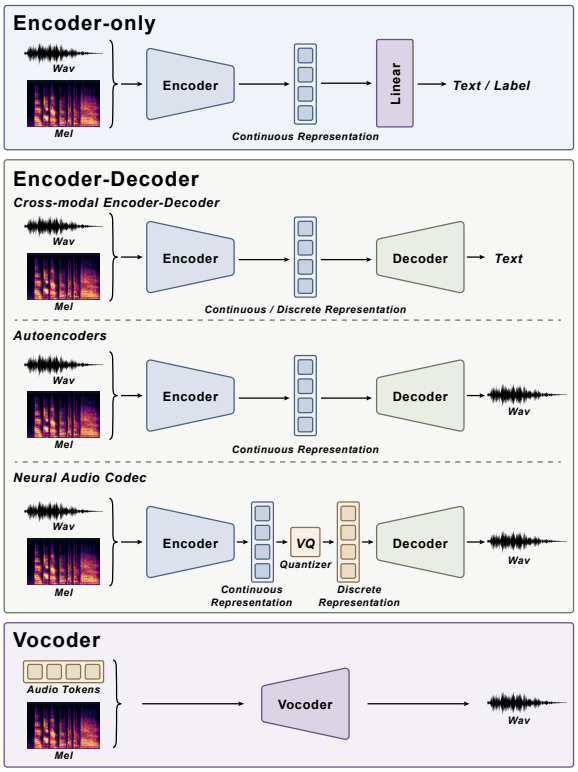


Figure 1: Architectures of Mainstream Audio Representation Models and Vocoder.

2.3.2 Encoder-Decoder

An encoder maps input signals to latent representations, which a decoder converts into waveforms, spectrograms, or text. Depending on the input/output modalities and bottleneck properties, this architecture features three subcategories:

Cross-modal Encoder-Decoders align and translate across heterogeneous modalities. Whisper (Radford et al., 2023) and OWSM v3.1 (Peng et al., 2024) employ native Transformer decoders for tasks like automatic speech recognition. StableToken (Song et al., 2026) extends Whisper with a quantization module for robust speech tokens, while SLAP (Scalable) (Mei et al., 2026) adopts an audio-text dual-encoder combined with a single text decoder for efficient multimodal retrieval and classification.

Autoencoders extract intrinsic representations by mapping an input $\mathbf{x}$ to a low-dimensional latent space $\mathbf{z}$ and minimizing the reconstruction error. Structurally, *Standard Autoencoders* such as DWM (Niizumi et al., 2026), DashengTokenizer (Dinkel et al., 2026a), and CARL (Sridhar and Cartwright, 2025) maintain symmetric frameworks. *Asymmetric Autoencoders*, including EnCodecMAE (Pepino et al., 2025), Audio-JEPA (Tuncay et al., 2025),

MATPAC (Quelennec et al., 2025), and WavJEPA (Yuksel et al., 2025), utilize lightweight decoders for pre-training reconstruction that are discarded during inference to yield highly discriminative features. Additionally, *Diffusion Autoencoders* like LatentFT (Wang and Huang, 2026) incorporate diffusion models into decoding to enhance latent capacity and signal restoration.

Neural Audio Codecs augment traditional autoencoders with quantizers. They impose a stringent information bottleneck to compress high-dimensional raw audio into low-dimensional tokens, followed by high-fidelity waveform restoration via a generative decoder. Representative models include MOSS-Audio-Tokenizer (Gong et al., 2026a), UniCodec (Jiang et al., 2025), SpeechTokenizer (Zhang et al., 2024), DAC (Kumar et al., 2023), EnCodec (Défossez et al., 2023), and SoundStream (Zeghidour et al., 2022).

2.3.3 Vocoders

Vocoders typically map acoustic intermediate representations, such as Mel-spectrograms, back into raw time-domain waveforms. Prominent examples of this paradigm include BemaGANv2 (Park et al., 2025), PeriodWave (Lee et al., 2025), Vocos (Siuzdak, 2024), and WaveNeXt (Okamoto et al., 2023). Furthermore, recent advancements have extended this framework beyond continuous spectrograms; for instance, T-Mimi (Wu et al., 2026b) trains the vocoder to reconstruct high-fidelity audio conditioned entirely on discrete codec tokens generated by the Mimi encoder.

2.4 Audio Representation Model Pre-training Paradigms

2.4.1 Supervised Pre-training

Audio representation models heavily rely on massive training datasets, yet acquiring clean human annotations is both challenging and expensive. Consequently, weakly supervised learning utilizing large-scale noisy data has emerged as one of the dominant paradigms to overcome this bottleneck. In large-scale weakly supervised pre-training, models such as Whisper (Radford et al., 2023), PANNs (Kong et al., 2020b), Contrastive-init/Captioning-scratch (Tseng et al., 2026), OWSM v3.1 (Peng et al., 2024), and StableToken (Song et al., 2026) directly leverage massive web data with weak labels, significantly enhancing their zero-shot generalization capabilities. Additionally, weakly supervised knowledge distillation—adopted by SLAP (Siamese)(Guinot et al., 2025), USAD (Chang et al., 2025), LLM-Codec (Yang et al., 2024), and CED (Dinkel et al., 2024)—utilizes soft labels generated by powerful teacher models to guide the learning process, enabling the efficient transfer of deep semantic features. On the generation side, neural vocoders including WaveNeXt (Okamoto et al., 2023), Vocos (Siuzdak, 2024), BemaGANv2 (Park et al., 2025), and T-Mimi (Wu et al., 2026b) still fundamentally depend on real physical waveforms or spectrograms as strong supervised regression targets to ensure high-fidelity audio reconstruction. In addition, PACE (Li et al., 2026a) proposes a continuous learning framework to address catastrophic forgetting in the pre-training of audio representation models.

2.4.2 Self-Supervised Pre-training (SSL)

Self-Supervised Learning (SSL) has become the dominant training paradigm for audio representation. It learns robust features from unlabeled audio through proxy tasks, deriving supervision directly from the data's structure. SSL objectives are primarily categorized into three types, often complemented by joint pre-training:

Predictive Objectives. These losses compel models to infer masked targets or distill teacher-generated latent features to capture high-level semantics. Specifically, HuBERT (Hsu et al., 2021), BEATs (Chen et al., 2023), and DWM (Niizumi et al., 2026) predict discrete labels, whereas data2vec (Baevski et al., 2022), EAT (Chen et al., 2024b), and Audio-JEPA (Tuncay et al., 2025) leverage Teacher-Student architectures for latent feature distillation. Additionally, LST (Lu et al., 2026b) employs autoregressive sequence prediction.

Contrastive Objectives. By pulling positive samples closer and pushing negatives apart, these losses filter out low-level acoustic noise to extract discriminative, abstract semantics. CLAP (Elizalde et al., 2023) and SLAP (Scalable) (Mei et al., 2026) utilize this mechanism for cross-modal audio-text alignment. Furthermore, wav2vec 2.0 (Baevski et al., 2020) combines contrastive learning with masked prediction over quantized representations, while AaSP (Yamamoto and Okusa, 2026) applies contrastive regularization to enhance representation discriminability.

Reconstructive Objectives. After an information bottleneck, these losses minimize reconstruction errors in signal or latent space to flawlessly

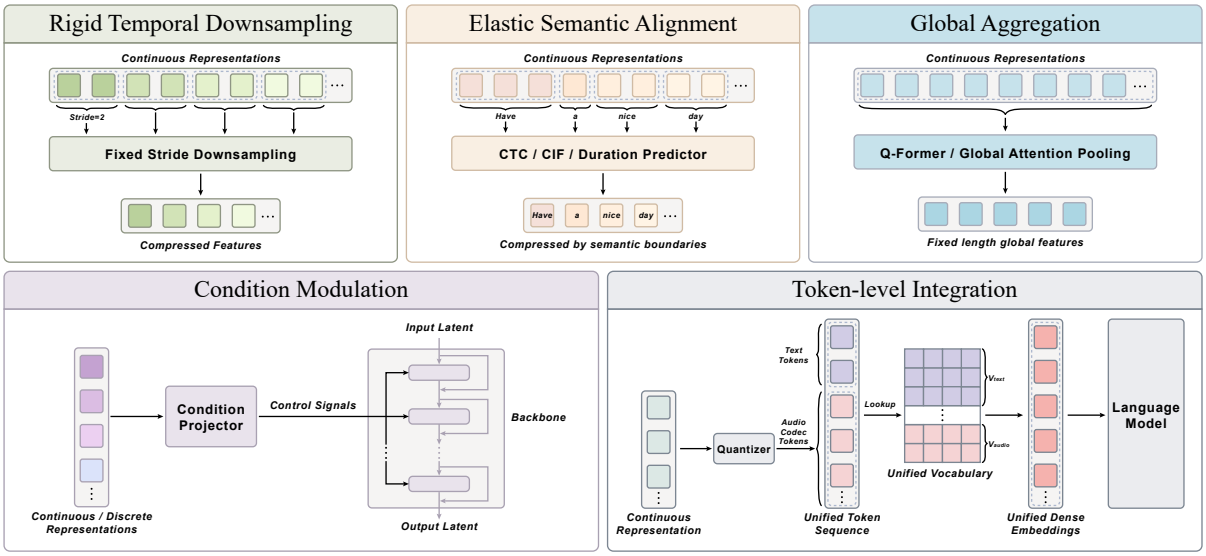


Figure 2: A Comparison of Audio Representation Integration Mechanisms for Downstream Models.

restore acoustic details. For example, Audio-MAE (Huang et al., 2022) and MAE-AST (Baade et al., 2022) directly reconstruct masked spectrograms. Similarly, neural codecs like VQ-VAE (van den Oord et al., 2017), EnCodec (Défossez et al., 2023), and DAC (Kumar et al., 2023) strictly rely on reconstruction objectives to guarantee high-fidelity audio generation.

3 Modeling for Audio Tasks

This chapter explores downstream workflows following audio representation extraction. Specifically, we investigate **integration mechanisms** into downstream models(Section 3.1) and subsequent downstream **modeling paradigms** (Section 3.2). Furthermore, the appendix introduces a comprehensive **taxonomy of audio tasks** (Section C), outlines **training paradigms** (Section E), and summarizes **representative audio models** (Section F).

3.1 Audio Representation Integration Mechanisms for Downstream Models

As illustrated in Figure 2, we systematically categorize these mechanisms into five paradigms: Rigid Temporal Downsampling, Elastic Semantic Alignment, Global Aggregation, Condition Modulation and Token-level Integration.

3.1.1 Rigid Temporal Downsampling

Rigid temporal downsampling applies fixed-ratio compression to extract acoustic envelopes. Though computationally efficient and strictly order-preserving, it risks over-smoothing and losing fine-grained high-frequency details. Common adapters for fixed-stride compression include 1D-CNNs (Xie and Wu, 2024; Chen et al., 2026; Tian et al., 2026b) and strided linear pooling (Xu et al., 2025; StepFun Audio Team, 2025; Tian et al., 2026a; Ghosh et al., 2026a). To mitigate the information loss inherent in uniform compression, window-level Q-Former (Tang et al., 2024; Wang et al., 2025d) and window-level self-attention (Jia et al., 2025) introduce adaptive weighting within fixed temporal windows.

3.1.2 Elastic Semantic Alignment

Addressing uneven audio density, elastic semantic alignment reshapes the time axis along semantic boundaries, mitigating the sequence length mismatch between acoustic frames and language model inputs while preserving semantics. Representative mechanisms include CTC-guided compression (Deng et al., 2026; Zhou et al., 2025), which suppresses blank frames via posterior peaks for variable-ratio compression; Continuous Integrate-and-Fire (CIF)(Gao et al., 2022; Deng et al., 2025a), which accumulates frame weights to emit fused representations upon reaching a threshold; and duration predictors(Le et al., 2023), which explicitly estimate frame counts per phoneme to guide feature pooling.

3.1.3 Global Aggregation

By discarding fine-grained temporal information, global aggregation mechanisms condense core acoustic features from variable-length audio into a fixed-length summary. Representative techniques

Cat.	Task	Benchmark & Metrics	Model	Downstream Inference Data Flow Pipeline	Performance
Und	ASR	**LibriSpeech** test-clean WER↓|test-other WER↓	CJST (Zhou et al., 2025)	$[\mathcal{M}\rightarrow\text{VGG}^{\text{RTD}}\rightarrow\text{Conformer}^{\text{ESA}}\rightarrow H_{\text{A}}]\rightarrow$Dec-only Tr$_{\text{AR}}\rightarrow\mathcal{T}$	2.09% | 4.71%
			MDM-ASR (Yen et al., 2026)	$[\mathcal{M}\rightarrow\text{FastConformer}^{\text{RTD}}\rightarrow H_{\text{A}}^{[\text{M}]}]\ \|\ [Z_{\text{T}}]\rightarrow$Non-causal Tr Dec$_{\text{iNAR}}\rightarrow\mathcal{T}$	1.80% | 3.60%
			Qwen3-Omni-30B-A3B-Instruct (Qwen Team, 2025)	$[\mathcal{M}\rightarrow\text{AuT}^{\text{RTD}}\rightarrow H_{\text{A}}]\oplus[Z_{\text{T}}]\rightarrow\text{Qwen3}_{\text{AR}}\rightarrow\mathcal{T}$	**1.22%** | **2.48%**
	AQA	**MMAR** ACC↑	DiFFA-2 (Zhou et al., 2026b)	$[\mathcal{M}\rightarrow$Whisper Encoder$\rightarrow H_{\text{A}}\rightarrow$[CNN$\rightarrow$Linear$^{\text{RTD}}\rightarrow H_{\text{A}}]\oplus$[Q-Former$^{\text{DA}}\rightarrow H_{\text{A}}]]\ \|\ [Z_{\text{T}}]\rightarrow\text{LLaDA}_{\text{iNAR}}\rightarrow\mathcal{T}$	50.80%
			AF-Next-Think (Ghosh et al., 2026a)	$[\mathcal{M}\rightarrow$AF-Whisper Encoder$\rightarrow H_{\text{A}}\rightarrow$Stride-2 Pool$^{\text{RTD}}\rightarrow$MLP$\rightarrow H_{\text{A}}]\oplus[\mathcal{T}]\rightarrow\text{Qwen2.5}_{\text{AR}}\rightarrow\mathcal{T}$	61.00%
			Audio-DeepThinker (He et al., 2026)	$[\mathcal{M}]\ \|\ [\mathcal{T}]\rightarrow$Qwen3-Omni-30b-A3B$_{\text{AR}}\rightarrow\mathcal{T}$	**74.00%**
Gen	TTS	**Seed-TTS-Eval** test-zh CER↓|test-zh SIM↑|test-en WER↓|test-en SIM↑	OmniVoice (Zhu et al., 2026)	$[\mathcal{W}\rightarrow$Higgs-Audio Tok$\rightarrow Z_{\text{A}}^{[]}]\oplus[\mathcal{T}]\rightarrow\text{Qwen3}_{\text{iNAR}}\rightarrow Z_{\text{A}}\rightarrow$Higgs-Audio Detok$\rightarrow\mathcal{W}$	1.60% | 0.741 | **0.84%** | **0.777**
			CosyVoice 3 (Du et al., 2025)	$[\mathcal{W}\rightarrow\text{FSQ-MinMo}\rightarrow Z_{\text{A}}^{[]}\oplus[\mathcal{T}]\rightarrow\text{Tr}_{\text{AR}}\rightarrow Z_{\text{A}}^{[\text{M}]}\rightarrow\text{DiT}_{\text{iNAR}}\rightarrow\mathcal{M}\rightarrow\text{Vocoder}\rightarrow\mathcal{W}$	**0.71%** | **0.775** | 1.45% | 0.695
			VibeVoice (Peng et al., 2026b)	$[\mathcal{W}\rightarrow$Acoustic Tok$^{\text{RTD}}\rightarrow H_{\text{A}}]\oplus[\mathcal{T}]\rightarrow\text{Qwen2.5}_{\text{AR}}\rightarrow H_{\text{A}}^{\text{LM}}\rightarrow$Diffusion$_{\text{iNAR}}\rightarrow H_{\text{A}}\rightarrow$Acoustic Dec$\rightarrow\mathcal{W}$	1.16% | 0.744 | 3.04% | 0.689
	TTA	**AudioCaps** $\text{FD}_{\text{L3}}\downarrow$| $\text{KL}_{\text{P}}\downarrow$| FD ↓| FAD ↓| KL ↓| CLAP ↑	TangoFlux (Hung et al., 2026)	$[\mathcal{W}\rightarrow$Stable Audio Open VAE$\rightarrow H_{\text{A}}^{[\text{M}]}]\oplus[\mathcal{T}\rightarrow\text{FLAN-T5}\rightarrow H_{\text{T}}^{[\text{M}]}]\rightarrow\text{DiT}_{\text{iNAR}}\rightarrow H_{\text{A}}\rightarrow$Stable Audio Open VAE Dec$\rightarrow\mathcal{W}$	83.58 | 0.95 | 1.57 | 2.34 | 0.52 | **0.385**
			AudioLDM 2 (Liu et al., 2024b)	$\mathcal{T}\rightarrow$[CLAP]$\ |\ $[FLAN-T5]$\rightarrow H_{\text{T}}\rightarrow\text{GPT-2}_{\text{AR}}\rightarrow H_{\text{T}}^{\text{LM}}\rightarrow$ Transformer-UNet$_{\text{iNAR}}\rightarrow H_{\text{A}}\rightarrow$VAE Dec$\rightarrow\mathcal{M}\rightarrow$HiFi-GAN$\rightarrow\mathcal{W}$	87.74 | 1.01 | 1.59 | 2.63 | 0.57 | 0.252
			AudioStory (Guo et al., 2025)	$[\mathcal{M}\rightarrow$Whisper Encoder$^{\text{RTD}}\rightarrow H_{\text{A}}]\oplus[\mathcal{T}_p]\rightarrow\text{Qwen2.5}_{\text{AR}}\rightarrow[\mathcal{T}\ \|\ Z_{\text{A}}^{[\text{M}]}]\rightarrow\text{TangoFlux}_{\text{iNAR}}\rightarrow H_{\text{A}}\rightarrow$VAE Dec$\rightarrow\mathcal{W}$	**83.39** | **0.91** | **1.52** | **2.29** | **0.51** | 0.383
Conv	VC	**Seed-TTS-Eval** test-zh CER↓|test-zh SIM↑|test-en WER↓|test-en SIM↑	MeanVC (Ma et al., 2026a)	$[\mathcal{W}\rightarrow\text{ECAPA-TDNN}^{\text{GA}}\rightarrow H_{\text{A}}^{[\text{M}]}\oplus[[[\mathcal{W}\rightarrow\text{Fast-U2++}\rightarrow H_{\text{A}}]\ \|\ [\mathcal{M}]]\rightarrow$Timbre Enc$\rightarrow H_{\text{A}}^{[\text{M}]}]\rightarrow\text{DiT}_{\text{1-NAR}}\rightarrow\mathcal{M}\rightarrow\text{Vocos}\rightarrow\mathcal{W}$	3.89 | **0.73** | — | —
			X-VC (Zheng et al., 2026)	$[\mathcal{W}\rightarrow\text{SAC Enc}^{\text{RTD}}\rightarrow H_{\text{A}}]\ \|\ [[\mathcal{W}\rightarrow\text{ERes2Net}\rightarrow H_{\text{A}}^{[\text{M}]}]\oplus[\mathcal{M}^{[\text{M}]}]\rightarrow\text{Tr}_{\text{1-NAR}}\rightarrow H_{\text{A}}\rightarrow\text{SAC Dec}\rightarrow\mathcal{W}$	1.99 | **0.73** | 2.83 | **0.63**
			Seed-VC tiny (Liu, 2024)	$[\mathcal{M}\rightarrow$Whisper Encoder$\rightarrow H_{\text{A}}^{[\text{M}]}]\oplus[\mathcal{W}\rightarrow\text{CAM++}\rightarrow H_{\text{A}}^{[\text{M}]}]\oplus[\mathcal{W}\rightarrow\text{RMVPE}\rightarrow H_{\text{A}}^{[\text{M}]}]\rightarrow\text{U-DiT}_{\text{iNAR}}\rightarrow\mathcal{M}\rightarrow\text{BigVGANv2}\rightarrow\mathcal{W}$	**1.79** | 0.60 | **2.24** | 0.41
Int	FSD	**Full-Duplex-Bench** DMOS↓ | SSIM↑ | LAL↓ | GPT4o↑	PersonaPlex (Roy et al., 2026)	$\mathcal{W}\rightarrow\text{Moshi}_{\text{AR}}\rightarrow\mathcal{W}$	3.90 | **0.57** | **0.070** | **4.210**
			Moshi (Défossez et al., 2024)	$[\mathcal{W}\rightarrow\text{Mimi Tok}^{\text{RTD}}\rightarrow Z_{\text{A}}^{[]}]\rightarrow\text{RQ-Tr}_{\text{AR}}\rightarrow[Z_{\text{T}}\ \|\ Z_{\text{A}}]\rightarrow\text{Mimi Detok}\rightarrow\mathcal{W}$	**3.11** | 0.10 | 0.265 | 0.765
			Freeze-Omni (Wang et al., 2025c)	$[\mathcal{M}\rightarrow\text{Speech Enc}\rightarrow\text{Adapter}^{\text{RTD}}\rightarrow H_{\text{A}}]\rightarrow\text{Qwen2-7B}_{\text{AR}}\rightarrow[H_{\text{A}}\ \|\ Z_{\text{T}}]\rightarrow\text{Speech Dec}_{\text{AR}}\rightarrow Z_{\text{A}}\rightarrow\text{TiCodec}\rightarrow\mathcal{W}$	3.51 | 0.05 | 0.953 | 3.615

$\mathcal{W}$: Waveform | $\mathcal{M}$: Mel-spec | $\mathcal{T}$: Text | H_A/H_T: Audio/Text Continuous Representations | Z_A/Z_T: Audio/Text Discrete Tokens
||: Parallel branches | $\oplus$: Feature fusion or concatenation | **Tr**: Transformer | **Enc/Dec**: Encoder/Decoder | **Tok/Detok**: Tokenizer/Detokenizer
RTD: Rigid Temporal Downsampling | CM: Condition Modulation | ESA: Elastic Semantic Alignment | GA: Global Aggregation | TI: Token-level Integration
AR: Autoregressive generation | iNAR: Iterative Non-Autoregressive | 1-NAR: One-shot Non-Autoregressive

Table 1: Overview of representative audio and speech models, their inference pipelines, and performance. Performance for TTA, VC, and FSD tasks are sourced from AudioStory (Guo et al., 2025), X-VC (Zheng et al., 2026), and PersonaPlex (Roy et al., 2026), respectively. For all other models, the performance are obtained directly from their respective original publications. Bold indicates the best result, underline denotes the second-best result.

include Q-Former (Ghosh et al., 2024; Biswas et al., 2026; Lu et al., 2026a), which uses a set of learnable queries to attend to the entire acoustic sequence, yielding compact, length-agnostic representations for downstream LLMs. Global Attention Pooling (Popov et al., 2022; Wang et al., 2026d) derives this macroscopic summary by computing a dynamically weighted sum of all audio frames using learned attention scores.

3.1.4 Condition Modulation

Condition modulation guides generation by dynamically reshaping backbone feature distributions using audio representation as independent variables. Three core mechanisms prevail: FiLM (Liu et al., 2024c; Gupta et al., 2026) predicts channel-wise affine parameters for coarse-grained global injection, offering minimal computational overhead in CNNs; AdaLN (Chen et al., 2025c; Liu, 2024; Dror et al., 2026) maps conditions to scaling and shifting parameters, injecting global styles while stabilizing gradients—now standard in DiTs and flow-matching architectures; and cross-attention (Vyas et al., 2023; Evans et al., 2025; Peng et al., 2026b) leverages dynamic similarity addressing, using backbone features as queries to achieve fine-grained alignment with variable-length condition sequences.

3.1.5 Token-level Integration

Token-level integration quantizes continuous audio into symbol sequences, reducing complex cross-modal modeling to a unified next-token prediction paradigm. Current approaches fall into three categories: (1) training dedicated language models from scratch without pre-trained backbones (Borsos et al., 2023a; Chen et al., 2024a; Anastassiou et al., 2024); (2) expanding the vocabularies of pre-trained large language models (Rubenstein et al., 2023; Liu et al., 2026a) or discrete diffusion models (Long et al., 2026); and (3) employing independent audio and text vocabularies for decoupled prediction (Défossez et al., 2024; Chien et al., 2026).

3.2 Downstream Task Modeling Paradigm

3.2.1 Discriminative

Discriminative modeling directly learns mappings from audio representations to specific targets, outputting classification or regression results via lightweight neural networks. Discrete targets aim to map inputs into a finite, mutually exclusive label space, widely used in audio classification (Gupta et al., 2026; Li et al., 2026a). Conversely, continuous targets project features into a continuous real-valued space, exemplified by predicting continuous time-frequency masks for speech separation and enhancement (Wang et al., 2023; Liu et al., 2024c), and similarity-based anomalous sound detection (Wilkinghoff et al., 2026).

3.2.2 Autoregressive

Autoregressive modeling generates causal sequences via the chain rule, constrained by continuous or discrete representations. Continuous representations map to LLM latent spaces via

lightweight adapters for audio understanding (Chu et al., 2024; Xu et al., 2025; Qwen Team, 2025; Ghosh et al., 2025, 2026a). However, predicting continuous matrices is intractable, restricting them to inputs; furthermore, the lack of shared symbols creates a modality gap that impedes chain-of-thought integration (Wu et al., 2026a). Conversely, discrete tokens seamlessly align with AR paradigms, sharing embedding spaces and cross-entropy loss with text for audio generation (Borsos et al., 2023a; Chen et al., 2025c; Gong et al., 2026b). Nevertheless, discretization incurs information loss, and inherently lengthy audio sequences necessitate mitigation via dynamic token compression (Lu et al., 2026b).

3.2.3 Iterative NAR Generation

Iterative generation performs parallel, synchronous sequence updates using global context. In discrete spaces, discrete diffusion recovers real tokens from uniform noise, enabling audio generation via vocabulary expansion (Borsos et al., 2023b; Long et al., 2026), or audio understanding by predicting text tokens from concatenated continuous audio and text prompts (Zhou et al., 2026a,b). In continuous spaces, traditional diffusion models denoise Gaussian noise on latent (Evans et al., 2025) or Mel-spectrogram (Popov et al., 2022) manifolds. To optimize evolution trajectories, flow matching directly regresses ODE vector fields, significantly easing network fitting and permitting large inference steps to balance fidelity and efficiency (Le et al., 2023; Liu, 2024).

3.2.4 One-shot NAR Generation

Assuming conditional independence, one-shot non-autoregressive (NAR) generation yields complete sequences in a single forward pass with $O(1)$ latency. To bridge cross-modal length mismatches, it typically employs elastic semantic alignment via CTC (Altinok, 2025), CIF (Gao et al., 2022), or duration predictors (Ren et al., 2021; Kim et al., 2021a). For strictly time-aligned scenarios, it employs deterministic feature mapping in continuous latent spaces (Zheng et al., 2026) or parallel slot-filling label prediction in discrete spaces (Mu et al., 2026). Consistency Models (Ye et al., 2023; Li et al., 2024b) distill iterative diffusion trajectories into single-step mappings for high-fidelity continuous generation.

3.2.5 Hybrid Modeling Paradigms

Hybrid paradigms balance sequence coherence and generation efficiency through three evolutionary paths. First, integrating AR and Iterative NAR: Seed-TTS (Anastassiou et al., 2024) and CosyVoice3 (Du et al., 2025) cascade AR planning with NAR acoustic refinement, whereas DiTAR (Jia et al., 2025) interleaves global AR with local intra-patch diffusion. Second, synergizing AR with One-shot NAR: VALL-E2(Chen et al., 2024a) and CALM (Rouard et al., 2026) leverage AR for long-range logic while employing ultra-fast single-step NAR to decode multi-layer or continuous concurrent features. Third, cascading discriminative and generative models: Speech World Model (Zhou et al., 2026c) uses discriminative graph networks to predict audio tags, serving as prompts to guide controlled AR generation.

3.3 Representative Systems and Data Flow Pipelines

Table 1 summarizes over 20 representative downstream systems across four core domains (understanding, generation, conversion, and interaction) evaluated on standard benchmarks (e.g., LibriSpeech (Panayotov et al., 2015), MMAR (Ma et al., 2026b), Seed-TTS-Eval (Anastassiou et al., 2024), AudioCaps (Kim et al., 2019), and Full-Duplex-Bench (Lin et al., 2025a)). Our notation describes workflows through input modalities, integration adapters, and downstream modeling paradigms. Reported performance is not directly comparable because model scales, pre-training data, and evaluation settings differ. The table illustrates how pipeline choices—continuous versus discrete representations and autoregressive versus non-autoregressive architectures—shape downstream capabilities.

4 Comparative Discussion and Future Directions

4.1 Time and Time-Frequency Trade-off

Selecting between time-domain (T) and time-frequency (T-F) representations involves a trade-off between signal preservation and computational efficiency. Time-domain modeling preserves raw transients and phase, favoring tasks requiring waveform reconstruction or low-latency execution, such as speech separation (Luo and Mesgarani, 2019), telecommunication (Défossez et al., 2023), and super-resolution (Yuan et al., 2025). Conversely,

T-F modeling compresses sequence length and introduces harmonic priors, making it suitable for semantic comprehension and classification tasks like event detection (Kong et al., 2020b) and emotion recognition (Penumajji, 2025). For downstream tasks demanding both frequency inductive biases and phase details, hybrid modeling is adopted. For example, OWL (Biswas et al., 2026) extracts STFT and phase features simultaneously to support spatial audio understanding.

4.2 Continuous and Discrete Representation Trade-off

Selecting between continuous and discrete representations fundamentally trades off acoustic fidelity and text-audio modality alignment. Continuous representations preserve dense acoustic details, making them suited for discriminative speech understanding (Wang et al., 2025a) and high-fidelity continuous diffusion (Hung et al., 2026). Conversely, discrete representations introduce quantization loss but successfully unify audio and text tokens within a single vocabulary. Thus, discrete tokens are preferred for interactive speech-to-speech agents and in-context voice synthesis (Chen et al., 2025c; Défossez et al., 2024), which require unified autoregressive generation. Recent systems combine both representation types hierarchically (Du et al., 2024c, 2025), or use continuous representations for high-quality, end-to-end reasoning and generation (Rouard et al., 2026).

4.3 Comparative Discussion on Integration Mechanisms

Selecting an integration adapter depends heavily on the downstream task's demand for temporal resolution versus global stylization. For sequence-preserving speech understanding and translation, rigid downsampling (Xie and Wu, 2024; Tian et al., 2026a; Tang et al., 2024) and elastic semantic alignment (Deng et al., 2026; Gao et al., 2022; Le et al., 2023) are commonly adopted to align temporal resolution while preserving local ordering. Conversely, global classification and retrieval tasks prefer global aggregation (Biswas et al., 2026; Popov et al., 2022) to extract length-agnostic semantic summaries. For high-fidelity generation, condition modulation mechanisms like AdaLN (Chen et al., 2025c) and cross-attention (Vyas et al., 2023) are standard for injecting stylistic constraints, while token-level integration (Borsos et al., 2023a; Défossez et al., 2024; Roy et al., 2026) is well-suited for unified, conversational speech-to-speech modeling.

4.4 Downstream Modeling Paradigm Selection and Evolving Trends

Downstream modeling depends on representations and adapters. Discriminative modeling, paired with rigid downsampling or global aggregation, is suited for classification and mask prediction (Wang et al., 2023; Gupta et al., 2026). Conversely, target-centric generation utilizes iterative or one-shot NAR paradigms, paired with condition modulation and elastic alignment, to achieve low-latency synthesis (Ren et al., 2021; Le et al., 2023). For conversational and reasoning tasks, AR modeling remains preferred to unify text-audio sequence learning (Borsos et al., 2023a). The current trend emphasizes hybrid cascades combining AR semantic planning with NAR acoustic refinement (Du et al., 2025), while pushing toward continuous autoregressive architectures (Rouard et al., 2026) and audio-interleaved reasoning (Lin et al., 2025b; Wu et al., 2026a) to eliminate quantization bottlenecks.

5 Conclusion

This survey presents a decoupled framework separating the audio processing pipeline into representation extraction and downstream modeling. We systematically review audio domains, representations, and architectures, while categorizing five integration mechanisms and downstream modeling paradigms based on task suitability and trade-offs. The appendix catalogs representative models. These taxonomies offer guidelines for designing next-generation audio foundation models.

Limitations

This survey covers literature up to early 2026 and may omit recent developments. Differences in datasets, hardware, and evaluation protocols limit direct comparisons, especially for subjective generation tasks. Our analysis remains primarily qualitative because controlled comparisons across design choices are scarce and incomplete public documentation restricts assessment of training costs and efficiency.

Declaration on Generative AI

Generative AI was used to assist with language editing and text refinement, including rephrasing sentences and removing redundant expressions.

References

Ossama Abdel-Hamid, Abdel rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. 2014. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10):1533–1545.

Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, and 1 others. 2023. MusicLM: Generating music from text. *arXiv preprint arXiv:2301.11325*.

Yang Ai and Zhen-Hua Ling. 2023. Neural speech phase prediction based on parallel estimation architecture and anti-wrapping losses. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Tony Alex, Sara Ahmed, Armin Mustafa, Muhammad Awais, and Philip J. B. Jackson. 2024. DTF-AT: Decoupled time-frequency audio transformer for event classification. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 17647–17655.

Duygu Altinok. 2025. Boosting ctc-based asr using llm-based intermediate loss regularization. In *International Conference on Text, Speech, and Dialogue*, pages 275–288. Springer.

Alexander Amini, Anna Banaszak, Harold Benoit, Arthur Böök, Tarek Dakhran, Song Duong, Alfred Eng, Fernando Fernandes, Marc Härkönen, Anne Harrington, Ramin M. Hasani, Saniya Karwa, Yuri Khrustalev, Maxime Labonne, Mathias Lechner, Valentine Lechner, Simon Lee, Zetian Li, Noel Loo, and 14 others. 2025. LFM2 technical report. *arXiv preprint arXiv:2511.23404*.

Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, Mingqing Gong, Peisong Huang, Qingqing Huang, Zhiying Huang, Yuanyuan Huo, Dongya Jia, Chumin Li, Feiya Li, Hui Li, and 27 others. 2024. Seed-tts: A family of high-quality versatile speech generation models. *Preprint*, arXiv:2406.02430.

Junyi Ao, Yuancheng Wang, Xiaohai Tian, Dekun Chen, Jun Zhang, Lu Lu, Yuxuan Wang, Haizhou Li, and Zhizheng Wu. 2024. SD-Eval: A benchmark dataset for spoken dialogue understanding beyond words. *Advances in Neural Information Processing Systems*, 37:56898–56918.

Hagai Aronowitz, Zvi Kons, Avihu Dekel, George Saon, and Ron Hoory. 2026. Speaker attributed automatic speech recognition using speech aware LLMs. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 19077–19081.

Siddhant Arora, Jinchuan Tian, Hayato Futami, Jeeweon Jung, Jiatong Shi, Yosuke Kashiwagi, Emiru Tsunoo, and Shinji Watanabe. 2025. Chain-of-thought training for open E2E spoken dialogue systems. In *Proceedings of the 26th Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA.

Alan Baade, Puyuan Peng, and David Harwath. 2022. MAE-AST: Masked autoencoding audio spectrogram transformer. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2438–2442.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. data2vec: A general framework for self-supervised learning in speech, vision and language. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1298–1312.

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 (NeurIPS)*, volume 33, pages 12449–12460.

Fahimeh Bahmaninezhad, Jian Wu, Rongzhi Gu, Shi-Xiong Zhang, Yong Xu, Meng Yu, and Dong Yu. 2019. A comprehensive study of speech separation: Spectrogram vs waveform separation. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 4574–4578.

Jisheng Bai, Haohe Liu, Mou Wang, Dongyuan Shi, Wenwu Wang, Mark D. Plumbley, Woon-Seng Gan, and Jianfeng Chen. 2025. AudioSetCaps: An enriched audio-caption dataset using automated generation pipeline with large audio and language models. *IEEE Transactions on Audio, Speech and Language Processing*, 33:2817–2829.

Nenad Banfic, David Fan, Kunal Vaishnavi, Sam Kemp, Sunghoon Choi, Rui Ren, Sayan Shaw, and Meng Tang. 2026. Pushing the limits of on-device streaming ASR: A compact, high-accuracy english model for low-latency inference. *arXiv preprint arXiv:2604.14493*.

Subrata Biswas, Mohammad Nur Hossain Khan, and Bashima Islam. 2026. OWL: Geometry-aware spatial reasoning for audio large language models. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matthew Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. 2023a. AudioLM: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2523–2533.

Zalán Borsos, Matt Sharifi, Damien Vincent, Eugene Kharitonov, Neil Zeghidour, and Marco Tagliasacchi. 2023b. SoundStorm: Efficient parallel audio generation. *arXiv preprint arXiv:2305.09636*.

Ronghe Cao, Yunxing Wang, Xiaolong Wu, Shuang Jin, and Huiling Niu. 2025. A lightweight tri-stream feature fusion network for speech emotion recognition. *IEEE Access*.

Heng-Jui Chang, Saurabhchand Bhati, James R. Glass, and Alexander H. Liu. 2025. USAD: Universal speech and audio representation via distillation. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8.

Gongyu Chen, Xiaoyu Zhang, Zhenqiang Weng, Junjie Zheng, Da Shen, Chaofan Ding, Wei-Qiang Zhang, and Zihao Chen. 2025a. YingMusic-SVC: Real-world robust zero-shot singing voice conversion with flow-grpo and singing-specific inductive biases. *arXiv preprint arXiv:2512.04793*.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. 2021a. GigaSpeech: An evolving, multi-domain ASR corpus with 10,000 hours of transcribed audio. In *Proceedings of Interspeech 2021*, pages 3670–3674. ISCA.

Hongjie Chen, Zehan Li, Yaodong Song, Wenming Deng, Yitong Yao, Yuxin Zhang, Hang Lv, Xuechao Zhu, Jian Kang, Jie Lian, Jie Li, Chao Wang, Shuangyong Song, Yongxiang Li, Zhongjiang He, and Xuelong Li. 2025b. GOAT-SLM: A spoken language model with paralinguistic and speaker characteristic awareness. *arXiv preprint arXiv:2507.18119*.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. 2020. VGGSound: A large-scale audio-visual dataset. In *Proceedings of the ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725.

Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. 2021b. WaveGrad: Estimating gradients for waveform generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Sanyuan Chen, Shujie Liu, Long Zhou, Yanqing Liu, Xu Tan, Jinyu Li, Sheng Zhao, Yao Qian, and Furu Wei. 2024a. VALL-E 2: Neural codec language models are human parity zero-shot text to speech synthesizers. *arXiv preprint arXiv:2406.05370*.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. 2022. WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Sanyuan Chen, Chengyi Wang, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. 2025c. Neural codec language models are zero-shot text to speech synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, 33:705–718.

Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, Wanxiang Che, Xiangzhan Yu, and Furu Wei. 2023. BEATs: Audio pre-training with acoustic tokenizers. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 5178–5193.

Wenxi Chen, Yuzhe Liang, Ziyang Ma, Zhisheng Zheng, and Xie Chen. 2024b. EAT: Self-supervised pre-training with efficient audio transformer. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3807–3815.

William Chen, Prem Seetharaman, Rithesh Kumar, Oriol Nieto, Shinji Watanabe, Justin Salamon, and Zeyu Jin. 2026. AudioChat: Unified audio storytelling, editing, and understanding with transfusion forcing. *arXiv preprint arXiv:2602.17097*.

Cheng-Han Chiang, Xiaofei Wang, Linjie Li, Chung-Ching Lin, Kevin Lin, Shujie Liu, Zhendong Wang, Zhengyuan Yang, Hung yi Lee, and Lijuan Wang. 2026. STITCH: Simultaneous thinking and talking with chunked reasoning for spoken language models. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Chung-Ming Chien, Manu Orsini, Eugene Kharitonov, Neil Zeghidour, Karen Livescu, and Alexandre Défossez. 2026. MoshiRAG: Asynchronous knowledge retrieval for full-duplex speech language models. *arXiv preprint arXiv:2604.12928*.

Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. 2022. Self-supervised learning with random-projection quantizer for speech recognition. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 3915–3924.

Iwona Christop, Tomasz Kuczyński, and Marek Kubis. 2025. ClonEval: An open voice cloning benchmark. *Preprint*, arXiv:2504.20581.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. 2024. Qwen2-Audio technical report. *arXiv preprint arXiv:2407.10759*.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-Audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*.

Luca Comanducci, Paolo Bestagini, and Stefano Tubaro. 2025. FakeMusicCaps: A dataset for detection and attribution of synthetic music generated via text-to-music models. *Journal of Imaging*, 11(7):242.

Seamless Communication, Loic Barrault, Yu-An Chung, Mariano Cora Meglioli, David Dale, Ning Dong, Paul-Ambroise Duquenne, Hady Elsahar, Hongyu Gong, Kevin Heffernan, John Hoffman, Christopher Klaiber, Pengwei Li, Daniel Licht, Jean Maillard, Alice Rakotoarison, Kaushik Ram Sadagopan, Guillaume Wenzek, Ethan Ye, and 49 others. 2025. Joint speech and text machine translation for up to 100 languages. *Nature*, 637(8046):587–593.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth ... Dalmia, Clara Rivera, and Ankur Bapna. 2023. FLEURS: Few-shot learning evaluation of universal representations of speech. In *Proceedings of the 2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and controllable music generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent. 2020. LibriMix: An open-source dataset for generalizable speech separation. *Preprint*, arXiv:2005.11262.

Courtenay V. Cotton and Daniel P. W. Ellis. 2011. Spectral vs. spectro-temporal features for acoustic event detection. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 69–72.

Steven Davis and Paul Mermelstein. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366.

Maureen de Seyssel, Antony D'Avirro, Adina Williams, and Emmanuel Dupoux. 2024. Emphassess: A prosodic benchmark on assessing emphasis transfer in speech-to-speech models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 495–507.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2023. High fidelity neural audio compression. *Transactions on Machine Learning Research*, 2023.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: A speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.

Keqi Deng, Wenxi Chen, Xie Chen, and Phil Woodland. 2025a. SimulS2S-LLM: Unlocking simultaneous inference of speech LLMs for speech-to-speech translation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16718–16734, Vienna, Austria. Association for Computational Linguistics.

Keqi Deng, Ruchao Fan, Bo Ren, Yiming Wang, and Jinyu Li. 2026. Speech llms are contextual reasoning transcribers. *Preprint*, arXiv:2604.00610.

Keqi Deng, Guangzhi Sun, and Phil Woodland. 2025b. Wav2Prompt: End-to-end speech prompt learning and task-based fine-tuning for text-based LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6940–6956.

Soham Deshmukh, Satvik Dixit, Rita Singh, and Bhiksha Raj. 2026. Mellow: A small audio language model for reasoning. In *Proceedings of the Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. MuST-C: A multilingual speech translation corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2012–2017, Minneapolis, Minnesota. Association for Computational Linguistics.

Xingjian Diao, Chunhui Zhang, Keyi Kong, Weiyi Wu, Chiyu Ma, Zhongyu Ouyang, Peijun Qing, Soroush Vosoughi, and Jiang Gui. 2025. SoundMind: RL-incentivized logic reasoning for audio-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 528–540. Association for Computational Linguistics.

Heinrich Dinkel, Xingwei Sun, Gang Li, Jiahao Mei, Yadong Niu, Jizhong Liu, Xiyang Li, Yifan Liao, Jiahao Zhou, Junbo Zhang, and Jian Luan. 2026a. DashengTokenizer: One layer is enough for unified audio understanding and generation. *CoRR*, abs/2602.23765.

Heinrich Dinkel, Yongqing Wang, Zhiyong Yan, Junbo Zhang, and Yujun Wang. 2024. CED: Consistent ensemble distillation for audio tagging. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 291–295.

Heinrich Dinkel, Zhiyong Yan, Tianzi Wang, Yongqing Wang, Xingwei Sun, Yadong Niu, Jizhong Liu, Gang Li, Junbo Zhang, and Jian Luan. 2026b. Glap: General contrastive audio-text pretraining across domains and languages. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 14737–14741. IEEE.

Tali Dror, Iftach Shoham, Moshe Buchris, Oren Gal, Haim H. Permuter, Gilad Katz, and Eliya Nachmani. 2026. Token-based audio inpainting via discrete

diffusion. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. 2020. Clotho: An audio captioning dataset. In *Proceedings of the ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 736–740.

Hui-Peng Du, Ye-Xin Lu, Yang Ai, and Zhen-Hua Ling. 2024a. APNet2: High-quality and high-efficiency neural vocoder with direct prediction of amplitude and phase spectra. In *Proceedings of the Man-Machine Speech Communication Conference*, pages 66–80. Springer.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, and 1 others. 2024b. CosyVoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Chongjia Ni, Xian Shi, Keyu An, Guanrou Yang, Yabin Li, Yanni Chen, Zhifu Gao, Qian Chen, Yue Gu, Mengzhe Chen, Yafeng Chen, and 3 others. 2025. CosyVoice 3: Towards in-the-wild speech generation via scaling-up and post-training. *arXiv preprint arXiv:2505.17589*.

Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, Fan Yu, Huadai Liu, Zhengyan Sheng, Yue Gu, Chong Deng, Wen Wang, Shiliang Zhang, Zhijie Yan, and Jingren Zhou. 2024c. CosyVoice 2: Scalable streaming speech synthesis with large language models. *arXiv preprint arXiv:2412.10117*.

Harishchandra Dubey, Ashkan Aazami, Vishak Gopal, Babak Naderi, Sebastian Braun, Ross Cutler, Alex Ju, Mehdi Zohourian, Min Tang, Mehrsa Golestaneh, and Robert Aichner. 2024. ICASSP 2023 Deep Noise Suppression Challenge. *IEEE Open Journal of Signal Processing*, 5:725–737.

Berkin Durmus, Blaise Munyampirwa, Eduardo Pacheco, Atila Orhon, and Andrey Leonov. 2025. SDBench: A comprehensive benchmark suite for speaker diarization. In *Proceedings of Interspeech 2025*, pages 1598–1602.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023. CLAP learning audio concepts from natural language supervision. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Jesse H. Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. 2019. GANSynth: Adversarial neural audio synthesis. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*.

Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. 2025. Stable audio open. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Jiajun Fan, Roger Ren, Jingyuan Li, Rahul Pandey, Prashanth Gurunath Shivakumar, Ivan Bulyko, Ankur Gandhe, Ge Liu, and Yile Gu. 2026. Incentivizing consistent, effective and scalable reasoning capability in audio LLMs via reasoning process rewards. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Ruchao Fan, Bo Ren, Yuxuan Hu, Rui Zhao, Shujie Liu, and Jinyu Li. 2025. Alignformer: Modality matching can achieve better zero-shot instruction-following speech-llm. *IEEE Journal of Selected Topics in Signal Processing*, 19(7):1329–1337.

Tao Feng, Zhiyuan Zhao, Yifan Xie, Yuqi Ye, Xiangyang Luo, Xun Guan, and Yu Li. 2025. STFT-Codec: High-fidelity audio compression through time-frequency domain representation. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.

Li Fu, Yu Xin, Sunlu Zeng, Lu Fan, Youzheng Wu, and Xiaodong He. 2026. PAC: Pronunciation-aware contextualized large language model-based automatic speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16377–16381.

Zhifu Gao, Shiliang Zhang, Ian McLoughlin, and Zhijie Yan. 2022. Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition. In *Proc. Interspeech 2022*, pages 2063–2067.

Gemini Team. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio Set: An ontology and human-labeled dataset for audio events. In *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780.

Sreyan Ghosh, Arushi Goel, Kaousheik Jayakumar, Lasha Koroshinadze, Nishit Anand, Zhifeng Kong, Siddharth Gururani, Sang gil Lee, Jaehyeon Kim, Aya Aljafari, Chao-Han Huck Yang, Sungwon Kim, Ramani Duraiswami, Dinesh Manocha, Mohammad Shoeybi, Bryan Catanzaro, Ming-Yu Liu, and Wei Ping. 2026a. Audio Flamingo Next: Next-generation open audio-language models for speech, sound, and music. *arXiv preprint arXiv:2604.10905*.

Sreyan Ghosh, Arushi Goel, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael

Valle, and Bryan Catanzaro. 2025. Audio Flamingo 3: Advancing audio intelligence with fully open large audio language models. In *Proceedings of the Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, and Bryan Catanzaro. 2026b. Music flamingo: Scaling music understanding in audio language models. In *International Conference on Learning Representations (ICLR)*.

Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, and Bryan Catanzaro. 2026c. Music Flamingo: Scaling music understanding in audio language models. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Sreyan Ghosh, Sonal Kumar, Ashish Seth, Chandra Kiran Reddy Evuru, Utkarsh Tyagi, S. Sakshi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. 2024. GAMA: A Large Audio-Language Model with Advanced Audio Understanding and Complex Reasoning Abilities. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6288–6313, Miami, Florida, USA. Association for Computational Linguistics.

Yitian Gong, Kuangwei Chen, Zhaoye Fei, Xiaogui Yang, Ke Chen, Yang Wang, Kexin Huang, Mingshu Chen, Ruixiao Li, Qingyuan Cheng, Shimin Li, and Xipeng Qiu. 2026a. MOSS-Audio-Tokenizer: Scaling audio tokenizers for future audio foundation models. *CoRR*, abs/2602.10934.

Yitian Gong, Botian Jiang, Yiwei Zhao, Yucheng Yuan, Kuangwei Chen, Yaozhou Jiang, Cheng Chang, Dong Hong, Mingshu Chen, Ruixiao Li, Yiyang Zhang, Yang Gao, Hanfu Chen, Ke Chen, Songlin Wang, Xiaogui Yang, Yuqian Zhang, Kexin Huang, ZhengYuan Lin, and 7 others. 2026b. MOSS-TTS technical report. *arXiv preprint arXiv:2603.18090*.

Yuan Gong, Yu-An Chung, and James R. Glass. 2021. AST: Audio spectrogram transformer. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 571–575.

Yuan Gong, Hongyin Luo, Alexander H. Liu, Leonid Karlinsky, and James R. Glass. 2024. Listen, think, and understand. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*.

Petr Grinberg and Hassan Shahmohammadi. 2026. ALARM: Audio-language alignment for reasoning models. *arXiv preprint arXiv:2603.09556*.

Julien Guinot, Alain Riou, Elio Quinton, and George Fazekas. 2025. SLAP: Siamese language-audio pretraining without negative samples for music understanding. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 382–390.

Tianyu Guo, Hongyu Chen, Hao Liang, Meiyi Qiang, Bohan Zeng, Linzhuang Sun, Bin Cui, and Wentao Zhang. 2026a. Brace: A benchmark for robust audio caption quality evaluation. *Advances in Neural Information Processing Systems*, 38.

Yuxin Guo, Teng Wang, Yuying Ge, Shijie Ma, Yixiao Ge, Wei Zou, and Ying Shan. 2025. AudioStory: Generating long-form narrative audio with large language models. *arXiv preprint arXiv:2508.20088*.

Zhihan Guo, Wenqian Cui, Guan-Ting Lin, Daxin Tan, Jingyao Li, Qiyong Zheng, Dingdong Wang, Jing Xiong, Han Shi, Jiaya Jia, and Irwin King. 2026b. A survey of Audio Reasoning in Multimodal Foundation Models. *Preprint*, arXiv:2605.21008.

Shubham Gupta, Adarsh Arigala, B. R. Dilleswari, and Sri Rama Murty Kodukula. 2026. Shared representation learning for reference-guided targeted sound detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 14547–14551.

Alexandros Haliassos, Rodrigo Mira, and Stavros Petridis. 2026. Pay attention to CTC: Fast and robust pseudo-labelling for unified speech recognition. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Ameer Hamza, Abdul Rehman Rehman Javed, Farkhund Iqbal, Natalia Kryvinska, Ahmad S. Almadhor, Zunera Jalil, and Rouba Borghol. 2022. Deepfake audio detection via MFCC features using machine learning. *IEEE Access*, 10:134018–134028.

Bing Han, Chushu Zhou, Yifan Yang, Wei Wang, Chenda Li, Wangyou Zhang, and Yanmin Qian. 2026. Representation-regularized convolutional audio transformer for audio understanding. *Preprint*, arXiv:2601.21612.

Xiang He, Chenxing Li, Jinting Wang, Yan Rong, Tianxin Xie, Wenfu Wang, Li Liu, and Dong Yu. 2026. Audio-DeepThinker: Progressive reasoning-aware reinforcement learning for high-quality chain-of-thought emergence in audio language models. *arXiv preprint arXiv:2604.18187*.

John R. Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. 2016. Deep clustering: Discriminative embeddings for segmentation and separation. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 31–35.

Lars Hertel, Huy Phan, and Alfred Mertins. 2016. Comparing time and frequency domain for audio event recognition using deep learning. In *Proceedings of*

the 2016 International Joint Conference on Neural Networks (IJCNN), pages 3407–3411. IEEE.

Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. 2012. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29(6):82–97.

Yueran Hou, Peilei Jia, Zihan Sun, Qihang Lu, Wenbing Yang, Yingming Gao, Ya Li, and Jun Gao. 2026. Hello-Chat: Towards realistic social audio interactions. *arXiv preprint arXiv:2602.23387*.

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.

Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. 2022. Masked autoencoders that listen. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Tiansheng Huang, Virat Shejwalkar, Oscar Chang, Milad Nasr, and Ling Liu. 2025. Rebellion: Noise-robust reasoning training for audio reasoning models. *arXiv preprint arXiv:2511.09682*.

Wen-Chin Huang, Lester Phillip Violeta, Songxiang Liu, Jiatong Shi, and Tomoki Toda. 2023. The singing voice conversion challenge 2023. In *Proceedings of the 2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8.

Chia-Yu Hung, Navonil Majumder, Zhifeng Kong, Ambuj Mehrish, Amir Zadeh, Chuan Li, Rafael Valle, Bryan Catanzaro, and Soujanya Poria. 2026. TangoFlux: Super fast and faithful text to audio generation with flow matching and clap-ranked preference optimization. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Won Jang, Dan Lim, Jaesam Yoon, Bongwan Kim, and Juntae Kim. 2021. UnivNet: A neural vocoder with multi-resolution spectrogram discriminators for high-fidelity waveform generation. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2207–2211.

Dongya Jia, Zhuo Chen, Jiawei Chen, Chenpeng Du, Jian Wu, Jian Cong, Xiaobin Zhuang, Chumin Li, Zhen Wei, Yuping Wang, and Yuxuan Wang. 2025. DiTAR: Diffusion transformer autoregressive modeling for speech generation. In *Forty-second International Conference on Machine Learning*.

Ye Jia, Michelle Tadmor Ramanovich, Quan Wang, and Heiga Zen. 2022. CVSS corpus and massively multilingual speech-to-speech translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6691–6703, Marseille, France. European Language Resources Association.

Yidi Jiang, Qian Chen, Shengpeng Ji, Yu Xi, Wen Wang, Chong Zhang, Xianghu Yue, Shiliang Zhang, and Haizhou Li. 2025. UniCodec: Unified audio codec with single domain-adaptive codebook. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 19112–19124.

Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aäron van den Oord, Sander Dieleman, and Koray Kavukcuoglu. 2018. Efficient neural audio synthesis. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2415–2424.

Takuhiro Kaneko, Kou Tanaka, Hirokazu Kameoka, and Shogo Seki. 2022. ISTFTNET: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time fourier transform. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6207–6211.

Harunori Kawano and Takeshi Sasaki. 2026. A human-inspired decoupled architecture for efficient audio representation learning. *Preprint*, arXiv:2603.26098.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. 2019. AudioCaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 119–132, Minneapolis, Minnesota. Association for Computational Linguistics.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021a. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International conference on machine learning*, pages 5530–5540. PMLR.

Ji-Hoon Kim, Sang-Hoon Lee, Ji-Hyun Lee, and Seong-Whan Lee. 2021b. Fre-GAN: Adversarial frequency-consistent audio synthesis. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2197–2201.

KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, Zhengtao Wang, Chu Wei, Yifei Xin, Xinran Xu, Jianwei Yu, Yutao Zhang, Xinyu Zhou, Y. Charles, and 21 others. 2025. Kimi-Audio technical report. *arXiv preprint arXiv:2504.18425*.

Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna. 2023. LibriTTS-R: A restored multi-speaker text-to-speech corpus. In *Proceedings of Interspeech 2023*, pages 5496–5500. ISCA.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020a. HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 17022–17033.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley. 2020b. PANNs: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2021. DiffWave: A versatile diffusion model for audio synthesis. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Khaled Koutini, Jan Schlüter, Hamid Eghbal-zadeh, and Gerhard Widmer. 2022. Efficient training of audio transformers with patchout. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2753–2757.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. 2023. High-fidelity audio compression with improved RVQGAN. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Sonal Kumar, Šimon Sedláček, Vaibhavi Lokegaonkar, Fernando López, Wenyi Yu, Nishit Anand, Hyeonggon Ryu, Lichang Chen, Maxim Plička, Miroslav Hlaváček, and 1 others. 2026a. MMAU-Pro: A challenging and comprehensive benchmark for holistic evaluation of audio general intelligence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 22688–22697.

Sonal Kumar, Prem Seetharaman, Ke Chen, Oriol Nieto, Jiaqi Su, Zhepei Wang, Rithesh Kumar, Dinesh Manocha, Nicholas J. Bryan, Zeyu Jin, and Justin Salamon. 2026b. TAC: Timestamped audio captioning. *arXiv preprint arXiv:2602.15766*.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and 1 others. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

Ryan Langman, Ante Jukic, Kunal Dhawan, Nithin Rao Koluguri, and Jason Li. 2025. Spectral codecs: Improving non-autoregressive speech synthesis with spectrogram-based audio codecs. *Preprint*, arXiv:2406.05298.

Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. 2023. Voicebox: Text-guided multilingual universal speech generation at scale. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2023. BigVGAN: A universal neural vocoder with large-scale training. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Sang-Hoon Lee, Ha-Yeong Choi, and Seong-Whan Lee. 2025. PeriodWave: Multi-period flow matching for high-fidelity waveform generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Maxime Leiber, Yosra Marnissi, Axel Barrau, Sylvain Meignen, and Laurent Massoulié. 2025. Optimal adaptive time-frequency representation via differentiable short-time fourier transform. *IEEE Transactions on Signal Processing*, 73:5047–5059.

Chang Li, Kanglei Zhou, and Liyuan Wang. 2026a. PACE: Pretrained audio continual learning. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Hanzhao Li, Liumeng Xue, Haohan Guo, Xinfa Zhu, Yuanjun Lv, Lei Xie, Yunlin Chen, Hao Yin, and Zhifei Li. 2024a. Single-Codec: Single-codebook speech codec towards high-performance speech generation. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Jiaqi Li, Xiaolong Lin, Zhekai Li, Shixi Huang, Yuancheng Wang, Chaoren Wang, Zhenpeng Zhan, and Zhizheng Wu. 2025a. DualCodec: A low-frame-rate, semantically-enhanced neural audio codec for speech generation. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Kang Li, Yan Song, Li-Rong Dai, Ian McLoughlin, Xin Fang, and Lin Liu. 2023a. AST-SED: An effective sound event detection method based on Audio Spectrogram Transformer. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Tianhua Li, Chenda Li, Wei Wang, Xin Zhou, Xihui Chen, Jianqing Gao, and Yanmin Qian. 2026b. SLM-SS: Speech language model for generative speech separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 15432–15436.

Tianpeng Li, Jun Liu, Tao Zhang, Yuanbo Fang, Da Pan, Mingrui Wang, Zheng Liang, Zehuan Li, Mingan Lin, Guosheng Dong, Jianhua Xu, Haoze Sun, Zenan Zhou, and Weipeng Chen. 2025b. Baichuan-Audio: A unified framework for end-to-end speech interaction. *arXiv preprint arXiv:2502.17239*.

Xiang Li, FanBu FanBu, Ambuj Mehrish, Yingting Li, Jiale Han, Bo Cheng, and Soujanya Poria. 2024b. Cm-tts: Enhancing real time text-to-speech synthesis efficiency through weighted samplers and consistency models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3777–3794.

Yadong Li, Guoxin Wu, Haiping Hou, and Biye Li. 2026c. UAF: A unified audio front-end LLM for full-duplex speech interaction. *arXiv preprint arXiv:2604.19221*.

Yinghao Aaron Li, Cong Han, Xilin Jiang, and Nima Mesgarani. 2023b. HiFTNet: A fast high-quality neural vocoder with harmonic-plus-noise filter and inverse short time fourier transform. *Preprint*, arXiv:2309.09493.

Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xingran Chen, Hanzhi Yin, Chenghao Xiao, Chenghua Lin, Anton Ragni, Emmanouil Benetos, Norbert Gyenge, Roger B. Dannenberg, Ruibo Liu, Wenhu Chen, Gus Xia, Yemin Shi, Wenhao Huang, Zili Wang, Yike Guo, and Jie Fu. 2024c. MERT: Acoustic music understanding model with large-scale self-supervised training. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ziwei Li, Lukuang Dong, Saierdaer Yusuyin, Xianyu Zhao, and Zhijian Ou. 2026d. Phonemes vs. projectors: An investigation of speech-language interfaces for LLM-based ASR. *arXiv preprint arXiv:2604.09332*.

Guan-Ting Lin, Chen Chen, Zhehuai Chen, and Hung-yi Lee. 2026a. Full-Duplex-Bench-v3: Benchmarking tool use for full-duplex voice agents under real-world disfluency. *arXiv preprint arXiv:2604.04847*.

Guan-Ting Lin, Shih-Yun Shan Kuan, Jiatong Shi, Kai-Wei Chang, Siddhant Arora, Shinji Watanabe, and Hung yi Lee. 2026b. Full-Duplex-Bench-v2: A multi-turn evaluation framework for duplex dialogue systems with an automated examiner. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 27–36.

Guan-Ting Lin, Jiachen Lian, Tingle Li, Qirui Wang, Gopala Anumanchipalli, Alexander H. Liu, and Hung-yi Lee. 2025a. Full-Duplex-Bench: A benchmark to evaluate full-duplex spoken dialogue models on turn-taking capabilities. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Xin Lin, Yang Zhang, and Shiyuan Wang. 2024. Mixed T-domain and TF-domain magnitude and phase representations for GAN-based speech enhancement. *Scientific Reports*, 14(1).

Yueqian Lin, Zhengmian Hu, Jayakumar Subramanian, Qinsi Wang, Nikos Vlassis, Hai Li, and Yiran Chen. 2025b. AsyncVoice agent: Real-time explanation for LLM planning and reasoning. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–4. IEEE.

Haohe Liu, Xuenan Xu, Yi Yuan, Mengyue Wu, Wenwu Wang, and Mark D. Plumbley. 2024a. SemantiCodec: An ultra low bitrate semantic audio codec for general sound. *IEEE Journal of Selected Topics in Signal Processing*, 18(8):1448–1461.

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. 2024b. AudioLDM 2: Learning holistic audio generation with self-supervised pretraining. *IEEE ACM Transactions on Audio, Speech, and Language Processing*, 32:2871–2883.

Songting Liu. 2024. Zero-shot voice conversion with diffusion transformers. *arXiv preprint arXiv:2411.09943*.

Tianqiao Liu, Xueyi Li, Hao Wang, Haoxuan Li, Zhichao Chen, Weiqi Luo, and Zitao Liu. 2026a. From text to talk: Audio-language model needs non-autoregressive joint training. *arXiv preprint arXiv:2509.20072*.

Xubo Liu, Qiuqiang Kong, Yan Zhao, Haohe Liu, Yi Yuan, Yuzhuo Liu, Rui Xia, Yuxuan Wang, Mark D. Plumbley, and Wenwu Wang. 2024c. Separate anything you describe. *IEEE Transactions on Audio, Speech, and Language Processing*, 33:458–471.

Yisi Liu, Nicholas Lee, and Gopala Anumanchipalli. 2026b. StyleStream: Real-time zero-shot voice style conversion. *arXiv preprint arXiv:2602.20113*.

Zefang Liu, Chenyang Zhu, Sangwoo Cho, and Shi-Xiong Zhang. 2026c. ReHear: Iterative pseudo-label refinement for semi-supervised speech recognition via audio large language models. *arXiv preprint arXiv:2602.18721*.

Zhijun Liu, Shuai Wang, Sho Inoue, Qibing Bai, and Haizhou Li. 2024d. Autoregressive diffusion transformer for text-to-speech synthesis. *arXiv preprint arXiv:2406.05551*.

LLM-Core Xiaomi. 2025. MiMo-Audio: Audio language models are few-shot learners. *arXiv preprint arXiv:2512.23808*.

Zuwei Long, Yunhang Shen, Heting Gao, Haoyu Cao, Xing Sun, Caifeng Shan, Ran He, Chaoyou Fu, and 1 others. 2026. Omni-Diffusion: Unified multimodal understanding and generation with masked discrete diffusion. In *International Conference on Machine Learning (ICML)*.

Ke-Han Lu, Zhehuai Chen, Szu-Wei Fu, Chao-Han Huck Yang, Sung-Feng Huang, Chih-Kai Yang, Chee-En Yu, Chun-Wei Chen, Wei-Chih Chen, Chien-yu Huang, and 1 others. 2026a. DeSTA2.5-Audio: Toward General-Purpose Large Audio Language Model with Self-Generated Cross-Modal

Alignment. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Lie Lu, Hong-Jiang Zhang, and Hao Jiang. 2002. Content analysis for audio classification and segmentation. *IEEE Transactions on Speech and Audio Processing*, 10(7):504–516.

Yen-Ju Lu, Yashesh Gaur, Wei Zhou, Benjamin Muller, Jesus Villalba, Najim Dehak, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Srini Iyer, and Duc Le. 2026b. Latent speech-text transformer. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yi Luo, Zhuo Chen, and Takuya Yoshioka. 2020. Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 46–50.

Yi Luo and Nima Mesgarani. 2019. Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(8):1256–1266.

Guobin Ma, Jixun Yao, Ziqian Ning, Yuepeng Jiang, Lingxin Xiong, Lei Xie, and Pengcheng Zhu. 2026a. Meanvc: Lightweight and streaming zero-shot voice conversion via mean flows. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 17347–17351.

Ziyang Ma, Zhuo Chen, Yuping Wang, Eng Siong Chng, and Xie Chen. 2025. Audio-CoT: Exploring chain-of-thought reasoning in large audio language model. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–6. IEEE.

Ziyang Ma, Yinghao Ma, Yanqiao Ext Zhu, Chen Yang, Yi-Wen Chao, Ruiyang Xu, Wenxi Chen, Yuanzhe Chen, Zhuo Chen, Jian Cong, and 1 others. 2026b. MMAR: A challenging benchmark for deep reasoning in speech, audio, music, and their mix. *Advances in Neural Information Processing Systems*, 38.

Navonil Majumder, Chia-Yu Hung, Deepanway Ghosal, Wei-Ning Hsu, Rada Mihalcea, and Soujanya Poria. 2024. Tango 2: Aligning diffusion-based text-to-audio generations through direct preference optimization. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pages 564–572.

Potsawee Manakul, Woody Haosheng Gan, Martijn Bartelds, Guangzhi Sun, William Barr Held, and Diyi Yang. 2026. Scaling open discrete audio foundation models with interleaved semantic, acoustic, and text tokens. *CoRR*, abs/2602.16687.

Ruskin Raj Manku, Yuzhi Tang, Xingjian Shi, Mu Li, and Alexander Smola. 2026. EmergentTTS-eval: Evaluating TTS models on complex prosodic, expressiveness, and linguistic challenges using model-as-a-judge. *Advances in Neural Information Processing Systems*, 38.

Xinhao Mei, Gaël Le Lan, Haohe Liu, Zhaoheng Ni, Varun Nagaraja, Yang Liu, Yangyang Shi, and Vikas Chandra. 2026. SLAP: Scalable language-audio pretraining with variable-duration audio and multi-objective training. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 14517–14521.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2018. A multi-device dataset for urban acoustic scene classification. In *Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, pages 9–13.

Christoph Minixhofer, Ondřej Klejch, and Peter Bell. 2026. TTSDS2: Resources and benchmark for evaluating human-quality text to speech systems. In *International Conference on Learning Representations (ICLR)*.

Bingshen Mu, Xian Shi, Xiong Wang, Hexin Liu, Jin Xu, and Lei Xie. 2026. LLM-ForcedAligner: A non-autoregressive and accurate LLM-based forced aligner for multilingual and long-form speech. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25627–25638.

Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. 2023. Masked modeling duo: Learning representations by encouraging both networks to model the input. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Daisuke Niizumi, Daiki Takeuchi, Masahiro Yasuda, Binh Thien Nguyen, Noboru Harada, and Nobutaka Ono. 2026. Rethinking masking strategies for masked prediction-based audio self-supervised learning. *Preprint*, arXiv:2603.23810.

Soha A. Nossier, Julie A. Wall, Mansour Moniri, Cornelius Glackin, and Nigel Cannings. 2020. A comparative study of time and frequency domain approaches to deep learning based speech enhancement. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Takuma Okamoto, Haruki Yamashita, Yamato Ohtani, Tomoki Toda, and Hisashi Kawai. 2023. WaveNeXt: ConvNeXt-based fast neural vocoder without ISTFT layer. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8.

Cesar Pachon and Dora Ballesteros. 2026. FakeVoiceFinder: An open-source framework for synthetic and deepfake audio detection. *Big Data and Cognitive Computing*, 10(1):25.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Taesoo Park, Mungwi Jeong, Mingyu Park, Narae Kim, Junyoung Kim, Dohyun Park, Mujung Kim, Jisang Yoo, Hoyun Lee, Sanghoon Kim, and 1 others. 2025. BemaGANv2: A vocoder with superior periodicity capture for long-term audio generation. In *Proceedings of the International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 0830–0834.

Jing Peng, Chenghao Wang, Yi Yang, Lirong Qian, Junjie Li, Yu Xi, Shuai Wang, and Kai Yu. 2026a. TASU2: Controllable CTC simulation for alignment and low-resource adaptation of speech LLMs. *arXiv preprint arXiv:2604.08384*.

Jing Peng, Yucheng Wang, Bohan Li, Yiwei Guo, Hankun Wang, Yangui Fang, Yu Xi, Haoyu Li, Xu Li, Ke Zhang, and 1 others. 2025a. A survey on speech large language models for understanding. *IEEE Journal of Selected Topics in Signal Processing*.

Yifan Peng, Jinchuan Tian, William Chen, Siddhant Arora, Brian Yan, Yui Sudo, Muhammad Shakeel, Kwanghee Choi, Jiatong Shi, Xuankai Chang, Jeeweon Jung, and Shinji Watanabe. 2024. OWSM v3.1: Better and faster open whisper-style speech models based on E-Branchformer. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Yizhou Peng, Yi-Wen Chao, Dianwen Ng, Yukun Ma, Chongjia Ni, Bin Ma, and Eng Siong Chng. 2025b. FD-Bench: A full-duplex benchmarking pipeline designed for full duplex spoken dialogue systems. In *Proceedings of Interspeech 2025*. ISCA.

Zhiliang Peng, Jianwei Yu, Wenhui Wang, Yaoyao Chang, Yutao Sun, Li Dong, Yi Zhu, Weijiang Xu, Hangbo Bao, Zehua Wang, Shaohan Huang, Yan Xia, and Furu Wei. 2026b. VibeVoice: Expressive podcast generation with next-token diffusion. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Niketa Penumajji. 2025. Deep learning for speech emotion recognition: A CNN approach utilizing Mel spectrograms. In *Proceedings of the 15th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pages 1–5. IEEE.

Leonardo Pepino, Pablo Riera, and Luciana Ferrer. 2025. EnCodecMAE: Leveraging neural codecs for universal audio representation learning. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, Mikhail Sergeevich Kudinov, and Jiansheng Wei. 2022. Diffusion-based voice conversion with fast maximum likelihood sampling scheme. In *Proceedings of the Tenth International Conference on Learning Representations (ICLR)*.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. 2019. Waveglow: A flow-based generative network for speech synthesis. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3617–3621.

Aurian Quelennec, Pierre Chouteau, Geoffroy Peeters, and Slim Essid. 2025. Masked latent prediction and classification for self-supervised audio representation learning. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Qwen Team. 2025. Qwen3-Omni technical report. *arXiv preprint arXiv:2509.17765*.

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 International Conference on Machine Learning (ICML)*, pages 28492–28518.

Alain Rakotomamonjy and Gilles Gasso. 2015. Histogram of gradients of time–frequency representations for audio scene classification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(1):142–153.

Lukas Rauch, Raphael Schwinger, Moritz Wirth, René Heinrich, Denis Huseljic, Marek Herde, Jonas Lange, Stefan Kahl, Bernhard Sick, Sven Tomforde, and Christoph Scholz. 2025. BirdSet: A large-scale dataset for audio classification in avian bioacoustics. In *Proceedings of the International Conference on Learning Representations (ICLR)*, volume 2025, pages 29482–29520.

Mirco Ravanelli and Yoshua Bengio. 2018. Speaker recognition from raw waveform with SincNet. In *Proceedings of the IEEE Spoken Language Technology Workshop (SLT)*, pages 1021–1028.

Mirco Ravanelli, Jianyuan Zhong, Santiago Pascual, Pawel Swietojanski, João Monteiro, Jan Trmal, and Yoshua Bengio. 2020. Multi-task self-supervised learning for robust speech recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6989–6993.

Pengyu Ren, Wenhao Guan, Kaidi Wang, Peijie Chen, Qingyang Hong, and Lin Li. 2025. ReFlow-VC: Zero-shot voice conversion based on rectified flow and speaker feature optimization. In *Proceedings of the 26th Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2021. FastSpeech 2: Fast and high-quality end-to-end text to speech. In *International Conference on Learning Representations (ICLR)*.

Fabian Retkowski, Maike Züfle, Andreas Sudmann, Dinah Pfau, Shinji Watanabe, Jan Niehues, and Alex Waibel. 2025. Summarizing speech: A comprehensive survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27263–27294.

Douglas A. Reynolds and Richard C. Rose. 1995. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83.

Simon Rouard, Manu Orsini, Axel Roebel, Neil Zeghidour, and Alexandre Défossez. 2026. Continuous audio language models. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Andrew Rouditchenko, Saurabhchand Bhati, Edson Araujo, Samuel Thomas, Hildegard Kuehne, Rogério Feris, and James R. Glass. 2025. Omni-R1: Do you really need audio to fine-tune your audio LLM? In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–7.

Rajarshi Roy, Jonathan Raiman, Sang-Gil Lee, Teodor-Dumitru Ene, Robert Kirby, Sungwon Kim, Jaehyeon Kim, and Bryan Catanzaro. 2026. PersonaPlex: Voice and role control for full duplex conversational speech models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16137–16141.

Paul K. Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, Hannah Muckenhirn, Dirk Padfield, James Qin, Danny Rozenberg, Tara N. Sainath, Johan Schalkwyk, Matthew Sharifi, Michelle Tadmor Ramanovich, Marco Tagliasacchi, and 11 others. 2023. AudioPaLM: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Kohei Saijo, Wangyou Zhang, Samuele Cornell, Robin Scheibler, Chenda Li, Zhaoheng Ni, Anurag Kumar, Marvin Sach, Yihui Fu, Wei Wang, and 1 others. 2025. Interspeech 2025 URGENT speech enhancement challenge. In *Proc. Interspeech 2025*, pages 858–862.

Sakshi Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. 2025. MMAU: A massive multi-task audio understanding and reasoning benchmark. In *Proceedings of the International Conference on Learning Representations (ICLR)*, volume 2025, pages 84929–84964.

Eric D. Scheirer and Malcolm Slaney. 1997. Construction and evaluation of a robust multifeature speech/music discriminator. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1331–1334.

Siavash Shams, Sukru Samet Dindar, Xilin Jiang, and Nima Mesgarani. 2024. SSAMBA: Self-supervised audio representation learning with Mamba state space model. In *Proceedings of the IEEE Spoken Language Technology Workshop (SLT)*, pages 1053–1059.

Maohao Shen, Tejas Jayashankar, Osama Hanna, Naoyuki Kanda, Yancheng Wang, Kateřina Žmolíková, Ruiming Xie, Niko Moritz, Anfeng Xu, Yashesh Gaur, Gregory Wornell, Qing He, and Jilong Wu. 2026. GSRM: Generative speech reward model for speech RLHF. *arXiv preprint arXiv:2602.13891*.

Zhichao Sheng, Shilin Zhou, Chen Gong, and Zhenghua Li. 2026. Think smart, not hard: Difficulty adaptive reasoning for large audio language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 32771–32790.

Yemin Shi, Yu Shu, Siwei Dong, Guangyi Liu, Jaward Sesay, Jingwen Li, and Zhiting Hu. 2025. Voila: Voice-language foundation models for real-time autonomous interaction and voice role-play. *arXiv preprint arXiv:2505.02707*.

Hubert Siuzdak. 2024. Vocos: Closing the gap between time-domain and Fourier-based neural vocoders for high-quality audio synthesis. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Hubert Siuzdak, Florian Grötschla, and Luca A. Lanzendörfer. 2024. SNAC: Multi-scale neural audio codec. In *Proceedings of the NeurIPS Workshop on AI-Driven Speech, Music, and Sound Generation*.

Yuhan Song, Linhao Zhang, Chuhan Wu, Aiwei Liu, Wei Jia, Houfeng Wang, and Zhou Xiao. 2026. Stabletoken: A noise-robust semantic speech tokenizer for resilient speechLLMs. In *The Fourteenth International Conference on Learning Representations*.

Sripathi Sridhar and Mark Cartwright. 2025. Compositional audio representation learning. In *Proceedings of the 2025 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5.

StepFun Audio Team. 2025. Step-Audio 2 technical report. *arXiv preprint arXiv:2507.16632*.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2024. SALMONN: Towards generic hearing abilities for large language models. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*.

Fei Tian, Xiangyu Tony Zhang, Yuxin Zhang, Haoyang Zhang, Yuxin Li, Daijiao Liu, Yayue Deng, Donghang Wu, Jun Chen, Liang Zhao, Chengyuan Yao,

Hexin Liu, Eng Siong Chng, Xuerui Yang, Xiangyu Zhang, Daxin Jiang, and Gang Yu. 2025. Step-Audio-R1 technical report. *arXiv preprint arXiv:2511.15848*.

Jinchuan Tian, Sang gil Lee, Zhifeng Kong, Sreyan Ghosh, Arushi Goel, Chao-Han Huck Yang, Wenliang Dai, Zihan Liu, Hanrong Ye, Shinji Watanabe, Mohammad Shoeybi, Bryan Catanzaro, Rafael Valle, and Wei Ping. 2026a. UALM: Unified audio language model for understanding, generation and reasoning. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Jinchuan Tian, Haoran Wang, Bo-Hao Su, Chien yu Huang, Qingzheng Wang, Jiatong Shi, William Chen, Xun Gong, Siddhant Arora, Chin-Jou Li, Masao Someki, Takashi Maekaku, Keita Goto, Yusuke Shinohara, Jin Sakuma, Chao-Han Huck Yang, and Shinji Watanabe. 2026b. Bagpiper: Solving open-ended audio tasks via rich captions. *arXiv preprint arXiv:2602.05220*.

Zeyue Tian, Zhaoyang Liu, Yizhu Jin, Ruibin Yuan, Liumeng Xue, Xu Tan, Qifeng Chen, Wei Xue, and Yike Guo. 2026c. AudioX: A unified framework for anything-to-audio generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Zeyue Tian, Binxin Yang, Zhaoyang Liu, Jiexuan Zhang, Ruibin Yuan, Hubery Yin, Qifeng Chen, Chen Li, Jing Lv, Wei Xue, and Yike Guo. 2026d. AudioOmni: Extending multi-modal understanding to versatile audio generation and editing. *arXiv preprint arXiv:2604.10708*.

Wei-Cheng Tseng, Xuanru Zhou, Mingyue Huo, Yiwen Shao, Hao Zhang, and Dong Yu. 2026. Revisiting audio-language pretraining for learning general-purpose audio representation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 34244–34263.

Ludovic Tuncay, Etienne Labbe, Emmanouil Benetos, and Thomas Pellegrini. 2025. Audio-JEPA: Joint-embedding predictive architecture for audio representation learning. *Preprint*, arXiv:2507.02915.

Joseph Turian, Björn W. Schuller, Dorien Herremans, Katrin Kirchhoff, L. Paola García-Perera, and Philippe Esling, editors. 2021. *HEAR: Holistic Evaluation of Audio Representations*, volume 166. PMLR.

Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W. Schuller, Christian J. Steinmetz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, and 1 others. 2022. HEAR: Holistic evaluation of audio representations. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 125–145. PMLR.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. 2016. WaveNet: A generative model for raw audio. In *Proceedings of the ISCA Speech Synthesis Workshop (SSW)*, page 125.

Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6306–6315.

Lester Phillip Violeta, Wen-Chin Huang, and Tomoki Toda. 2025. Serenade: A singing style conversion framework based on audio infilling. In *Proceedings of the 33rd European Signal Processing Conference (EUSIPCO)*, pages 411–415. IEEE.

Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, Jeff Wang, Ivan Cruz, Bapi Akula, Akinniyi Akinyemi, Brian Ellis, Rashel Moritz, Yael Yungster, Alice Rakotoarison, Liang Tan, and 5 others. 2023. Audiobox: Unified audio generation with natural language prompts. *arXiv preprint arXiv:2312.15821*.

Zixiang Wan, Guochang Zhang, Yifeng He, and Jianqiang Wei. 2025. SpecTokenizer: A lightweight streaming codec in the compressed spectrum domain. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. 2021. CoVoST 2 and massively multilingual speech translation. In *Proc. Interspeech 2021*, pages 2247–2251.

Chaoren Wang, Heng Lu, Xueyao Zhang, Shujie Liu, Yan Lu, Jinyu Li, and Zhizheng Wu. 2026a. Closing the modality reasoning gap for speech large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18821–18835.

Dingdong Wang, Junan Li, Mingyu Cui, Dongchao Yang, Xueyuan Chen, and Helen Meng. 2025a. Speech discrete tokens or continuous features? a comparative analysis for spoken language understanding in speechllms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24924–24935.

Dingdong Wang, Junan Li, Jincenzi Wu, Dongchao Yang, Xueyuan Chen, Tianhua Zhang, and Helen M. Meng. 2026b. MMSU: A massive multi-task spoken language understanding and reasoning benchmark. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Dingdong Wang, Shujie Liu, Tianhua Zhang, Youjun Chen, Jinyu Li, and Helen M. Meng. 2026c. EmotionThinker: Prosody-aware reinforcement learning for explainable speech emotion reasoning. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

He Wang, Linhan Ma, Dake Guo, Xiong Wang, Lei Xie, Jin Xu, and Junyang Lin. 2025b. ContextASR-Bench: A massive contextual speech recognition benchmark. *Preprint*, arXiv:2507.05727.

Hualei Wang, Yiming Li, Shuo Ma, Hong Liu, and Xiangdong Wang. 2026d. Listening between the frames: Bridging temporal gaps in large audio-language models. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence*, pages 26233–26241. AAAI Press.

Hui Wang, Cheng Liu, Junyang Chen, Haoze Liu, Yuhang Jia, Shiwan Zhao, Jiaming Zhou, Haoqin Sun, Hui Bu, and Yong Qin. 2026e. TTA-Bench: A comprehensive benchmark for evaluating text-to-audio models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33512–33520.

Hui Wang, Jinghua Zhao, Junyang Cheng, Cheng Liu, Yuhang Jia, Haoqin Sun, Jiaming Zhou, and Yong Qin. 2026f. AudioEval: Automatic dual-perspective and multi-dimensional evaluation of text-to-audio-generation. *Preprint*, arXiv:2510.14570.

Jieyi Wang, Yazhe Niu, Dexuan Xu, and Zhongyu Wei. 2026g. Listen, pause, and reason: Toward perception-grounded hybrid reasoning for audio understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 35653–35671.

Mason Long Wang and Cheng-Zhi Anna Huang. 2026. Latent fourier transform. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-weon Jung, Hye-jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi Kinnunen, Nicholas Evans, Kong Aik Lee, Junichi Yamagishi, Myeonghun Jeong, Ge Zhu, Yongyi Zang, You Zhang, Soumi Maiti, Florian Lux, and 10 others. 2026h. ASVspoof 5: Design, collection and validation of resources for spoofing, deepfake, and adversarial attack detection using crowdsourced speech. *Computer Speech & Language*, 95:101825.

Xiong Wang, Yangze Li, Chaoyou Fu, Yike Zhang, Yunhang Shen, Lei Xie, Ke Li, Xing Sun, and Long Ma. 2025c. Freeze-Omni: A smart and low latency speech-to-speech dialogue model with frozen LLM. In *International Conference on Machine Learning (ICML)*, pages 63345–63354. PMLR.

Zhong-Qiu Wang, Samuele Cornell, Shukjae Choi, Younglo Lee, Byeong-Yeol Kim, and Shinji Watanabe. 2023. TF-GridNet: Integrating full- and sub-band modeling for speech separation. *IEEE ACM Transactions on Audio, Speech, and Language Processing*, 31:3221–3236.

Ziqian Wang, Xianjun Xia, Xinfa Zhu, and Lei Xie. 2025d. U-SAM: An audio language model for unified speech, audio, and music understanding. In *Proceedings of the 26th Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA.

Shu wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, and 1 others. 2021. SUPERB: Speech processing universal PERformance benchmark. In *Proc. Interspeech 2021*.

Shu wen Yang, Ming Tu, Andy T. Liu, Xinghua Qu, Hung yi Lee, Lu Lu, Yuxuan Wang, and Yonghui Wu. 2026. ParaS2S: Benchmarking and aligning spoken language models for paralinguistic-aware speech-to-speech interaction. In *International Conference on Learning Representations (ICLR)*.

Gijs Wijngaard, Elia Formisano, Michele Esposito, and Michel Dumontier. 2026. AudSemThinker: Enhancing audio-language models through reasoning over semantics of sound. In *Proceedings of the Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Kevin Wilkinghoff, Sarthak Yadav, and Zheng-Hua Tan. 2026. Temporal pooling strategies for training-free anomalous sound detection with self-supervised audio embeddings. *arXiv preprint arXiv:2603.04605*.

Daiqing Wu, Xuan Zhang, Dongbao Yang, Jiashu Yao, Longfei Chen, Qingsong Liu, Sicheng Zhao, Can Ma, Yangyang Kang, and Yu Zhou. 2026a. Echo: Towards advanced audio comprehension via audio-interleaved reasoning. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Haibin Wu, Ho-Lam Chung, Yi-Cheng Lin, Yuan-Kuei Wu, Xuanjun Chen, Yu-Chi Pai, Hsiu-Hsuan Wang, Kai-Wei Chang, Alex Liu, and Hung-yi Lee. 2024. Codec-SUPERB: An in-depth analysis of sound codec models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10330–10348. Association for Computational Linguistics.

Haibin Wu, Bach Viet Do, Naveen Suda, Julian Chan, Madhavan CR, Gene-Ping Yang, Yi-Chiao Wu, Naoyuki Kanda, Yossef Adi, Xin Lei, and 1 others. 2026b. T-Mimi: A transformer-based mimi decoder for real-time on-phone TTS. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16992–16996.

Shih-Lun Wu, Ge Zhu, Juan-Pablo Caceres, Cheng-Zhi Anna Huang, and Nicholas J. Bryan. 2026c. Stemphonic: All-at-once flexible multi-stem music generation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 15067–15071.

Shilong Wu. 2025. M3SD: Multi-modal, multi-scenario and multi-language speaker diarization dataset. *Preprint*, arXiv:2506.14427.

Kun Xie, Feiyu Shen, Junjie Li, Fenglong Xie, Xu Tang, and Yao Hu. 2025a. FireRedTTS-2: Towards long conversational speech generation for podcast and chatbot. *arXiv preprint arXiv:2509.02020*.

Tianxin Xie, Yan Rong, Pengfei Zhang, Wenwu Wang, and Li Liu. 2025b. Towards controllable speech synthesis in the era of large language models: A systematic survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 764–791.

Yuan Xie, Jiaqi Song, Guang Qiu, Xianliang Wang, Ming Lei, Jie Gao, and Jie Wu. 2026. Rethinking entropy allocation in LLM-based ASR: Understanding the dynamics between speech encoders and LLMs. *arXiv preprint arXiv:2604.08003*.

Zhifei Xie, Mingbao Lin, Zihang Liu, Pengcheng Wu, Shuicheng Yan, and Chunyan Miao. 2025c. Audio-Reasoner: Improving reasoning capability in large audio language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 23829–23851. Association for Computational Linguistics.

Zhifei Xie and Changqiao Wu. 2024. Mini-Omni: Language models can hear, talk while thinking in streaming. *arXiv preprint arXiv:2408.16725*.

Jilan Xu, Carl Thomé, Danijela Horak, Weidi Xie, and Andrew Zisserman. 2026. Scaling audio-text retrieval with multimodal large language models. *arXiv preprint arXiv:2602.18010*.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. 2025. Qwen2.5-Omni technical report. *arXiv preprint arXiv:2503.20215*.

Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald. 2019. CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92). Technical report, University of Edinburgh, The Centre for Speech Technology Research (CSTR).

Kohei Yamamoto and Kosuke Okusa. 2026. AaSP: Aliasing-aware self-supervised pre-training for audio spectrogram transformers. *IEEE Transactions on Audio, Speech and Language Processing*. Early access.

Yongpeng Yan, Yanan Li, Qiyang Xiao, and Yanzhen Ren. 2026. PRoADS: Provably secure and robust audio diffusion steganography with latent optimization and backward euler inversion. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13597–13601.

Dongchao Yang, Haohan Guo, Yuanyuan Wang, Rongjie Huang, Xiang Li, Xu Tan, Xixin Wu, and Helen Meng. 2024. UniAudio 1.5: Large language model-driven audio codec is a few-shot audio task learner. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Dongchao Yang, Songxiang Liu, Rongjie Huang, Jinchuan Tian, Chao Weng, and Yuexian Zou. 2023a. HiFi-Codec: Group-residual vector quantization for high fidelity audio codec. *Preprint*, arXiv:2305.02765.

Dongchao Yang, Yuxin Xie, Yuguo Yin, Zheyu Wang, Xiaoyu Yi, Gongxi Zhu, Xiaolong Weng, Zihan Xiong, Yingzhe Ma, Dading Cong, Jingliang Liu, Zihang Huang, Jinghan Ru, Rongjie Huang, Haoran Wan, Peixu Wang, Kuoxi Yu, Helin Wang, Liming Liang, and 10 others. 2026a. HeartMuLa: A family of open sourced music foundation models. *arXiv preprint arXiv:2601.10547*.

Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. 2023b. Diffsound: Discrete diffusion model for text-to-sound generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1720–1733.

Fei Yang, Xuanfan Ni, Renyi Yang, Jiahui Geng, Qing Li, Chenyang Lyu, Yichao Du, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2026b. LongSpeech: A scalable benchmark for transcription, translation and understanding in long speech. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 15547–15551.

Hao Yang, Minghan Wang, Tongtong Wu, Lizhen Qu, Ehsan Shareghi, and Gholamreza Haffari. 2026c. Resurfacing paralinguistic awareness in large audio language models. *arXiv preprint arXiv:2603.11947*.

Ke Yang, Bolin Chen, Yuejie Li, Yueying Hua, Jianhao Nie, Yueping He, Bowen Li, and Chengjun Mao. 2026d. Attention-guided evidence grounding for spoken question answering. *arXiv preprint arXiv:2603.16292*.

Xiaoyu Yang, Yifan Yang, Zengrui Jin, Ziyun Cui, Wen Wu, Baoxiang Li, Chao Zhang, and Phil Woodland. 2026e. SPEAR: A unified SSL framework for learning speech and audio representations. *Preprint*, arXiv:2510.25955.

Zhengdong Yang, Shuichiro Shimizu, Yahan Yu, and Chenhui Chu. 2025. When large language models meet speech: A survey on integration approaches. *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20298–20315.

Zhen Ye, Wei Xue, Xu Tan, Jie Chen, Qifeng Liu, and Yike Guo. 2023. Comospeech: One-step speech and singing voice synthesis via consistency model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1831–1839.

Hao Yen, Pin-Jui Ku, Ante Jukić, and Sabato Marco Siniscalchi. 2026. MDM-ASR: Bridging accuracy and efficiency in ASR with diffusion-based non-autoregressive decoding. *arXiv preprint arXiv:2602.18952*.

Jun-Wei Yeow, Ee-Leng Tan, Santi Peksi, and Woon-Seng Gan. 2025. Environmental acoustic intelligence

through sound event localization and detection: a review. *npj Acoustics*, 1(1):31.

Xiaofeng Yu, Jiaheng Dong, Jean Honorio, Abhirup Ghosh, Hong Jia, and Ting Dang. 2026. Disentangling reasoning in large audio-language models for ambiguous emotion prediction. *arXiv preprint arXiv:2603.08230*.

Jiajun Yuan, Xiaochen Wang, Yuhang Xiao, Yulin Wu, Chenhao Hu, and Xueyang Lv. 2025. STSR: High-fidelity speech super-resolution via spectral-transient context modeling. *Preprint*, arXiv:2509.03913.

Ruibin Yuan, Yinghao Ma, Yizhi Li, Ge Zhang, Xingran Chen, Hanzhi Yin, Zhuo Le, Yiqi Liu, Jiawen Huang, Zeyue Tian, Binyue Brook Deng, Ningzhi Wang, Chenghua Lin, Emmanouil Benetos, Anton Ragni, Norbert Gyenge, Roger Dannenberg, Wenhu Chen, Gus Xia, and 6 others. 2023. MARBLE: Music audio representation benchmark for universal evaluation. In *Advances in Neural Information Processing Systems*, volume 36, pages 39626–39647. Curran Associates, Inc.

Goksenin Yuksel, Pierre Guetschel, Michael Tangermann, Marcel van Gerven, and Kiki van der Heijden. 2025. WavJEPA: Semantic learning unlocks robust audio foundation models for raw waveforms. *arXiv preprint arXiv:2509.23238*.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2022. SoundStream: An end-to-end neural audio codec. *IEEE ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507.

Neil Zeghidour, Olivier Teboul, Félix de Chaumont Quitry, and Marco Tagliasacchi. 2021. LEAF: A learnable frontend for audio classification. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. 2019. LibriTTS: A corpus derived from LibriSpeech for text-to-speech. In *Proc. Interspeech 2019*.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. 2024. Glm-4-voice: Towards intelligent and human-like end-to-end spoken chatbot. *arXiv preprint arXiv:2412.02612*.

Dan Zhang, Yishu Lei, Jing Hu, Shuwei He, Songhe Deng, Xianlong Luo, Danxiang Zhu, Shikun Feng, Rui Liu, Jingzhou He, Yu Sun, Hua Wu, and Haifeng Wang. 2026a. Eureka-Audio: Triggering audio intelligence in compact language models. *arXiv preprint arXiv:2602.13954*.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15757–15773. Association for Computational Linguistics.

Honglei Zhang, Yuting Chen, Chenpeng Hu, Siyue Zhang, and Yilei Shi. 2026b. ReasonAudio: A benchmark for evaluating reasoning beyond matching in text-audio retrieval. *Preprint*, arXiv:2605.03361.

Junbo Zhang, Heinrich Dinkel, Yadong Niu, Chenyu Liu, Si Cheng, Anbei Zhao, and Jian Luan. 2025. X-ARES: A comprehensive framework for assessing audio encoder performance. In *Proc. Interspeech 2025*, pages 4868–4872.

Qian Zhang, Yuqin Cao, Yixuan Gao, and Xiongkuo Min. 2026c. VidAudio-Bench: Benchmarking V2A and VT2A generation across four audio categories. *Preprint*, arXiv:2604.10542.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. 2024. SpeechTokenizer: Unified speech tokenizer for speech language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yizhou Zhang, Yuan Gao, Wangjin Zhou, Zicheng Yuan, Keisuke Imoto, and Tatsuya Kawahara. 2026d. SONAR: Self-distilled continual pre-training for domain adaptive audio representation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 14752–14756.

Zhisheng Zhang, Xiang Li, Yixuan Zhou, Jing Peng, Shengbo Cai, Guoyang Zeng, and Zhiyong Wu. 2026e. UniSRCodec: Unified and low-bitrate single codebook codec with sub-band reconstruction. *arXiv preprint arXiv:2601.02776*.

Yi Zhao, Wen-Chin Huang, Xiaohai Tian, Junichi Yamagishi, Rohan Kumar Das, Tomi Kinnunen, Zhen-Hua Ling, and Tomoki Toda. 2020. Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion. In *Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*, pages 80–98.

Qixi Zheng, Yuxiang Zhao, Tianrui Wang, Wenxi Chen, Kele Xu, Yikang Li, Qinyuan Chen, Xipeng Qiu, Kai Yu, and Xie Chen. 2026. X-VC: Zero-shot streaming voice conversion in codec space. *arXiv preprint arXiv:2604.12456*.

Jiaming Zhou, Hongjie Chen, Shiwan Zhao, Jian Kang, Jie Li, Enzhi Wang, Yujie Guo, Haoqin Sun, Hui Wang, Aobo Kong, Yong Qin, and Xuelong Li. 2026a. DIFFA: Large language diffusion models can listen and understand. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence*, pages 35112–35120. AAAI Press.

Jiaming Zhou, Xuxin Cheng, Shiwan Zhao, Yuhang Jia, Cao Liu, Ke Zeng, Xunliang Cai, and Yong Qin. 2026b. DIFFA-2: A practical diffusion large language model for general audio understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 4789–4805.

Wei Zhou, Junteng Jia, Leda Sari, Jay Mahadeokar, and Ozlem Kalinli. 2025. Cjst: Ctc compressor based joint speech and text training for decoder-only asr. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Xuanru Zhou, Jiachen Lian, Henry Hong, Xinyi Yang, and Gopala Anumanchipalli. 2026c. Speech world model: Causal state-action planning with explicit reasoning for speech. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Xuanru Zhou, Yiwen Shao, Wei-Cheng Tseng, and Dong Yu. 2026d. Unlocking strong supervision: A data-centric study of general-purpose audio pretraining methods. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24780–24791.

Han Zhu, Lingxuan Ye, Wei Kang, Zengwei Yao, Liyong Guo, Fangjun Kuang, Zhifeng Han, Weiji Zhuang, Long Lin, and Daniel Povey. 2026. OmniVoice: Towards omnilingual zero-shot text-to-speech with diffusion language models. *arXiv preprint arXiv:2604.00688*.

Zhiyuan Zhu, Yu Zhang, Wenxiang Guo, Changhao Pan, and Zhou Zhao. 2025. Asaudio: A survey of advanced spatial audio research. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 417–442.

A Overview of Taxonomy Framework

To provide a structured overview of the current research landscape, we summarize the classification framework proposed in this paper as shown in Figure 3. This systematic categorization organizes existing literature across key design dimensions, including audio representation learning, integration mechanisms, and downstream task modeling paradigms.

B Evaluation Methods for Audio Representation Models

Evaluating audio representations centers on assessing their latent space features. Models' representational and generalization capabilities are typically quantified based on feature independence, transferability, and acoustic fidelity across three mainstream paradigms:

B.1 Frozen Feature Extraction (FFE)

FFE freezes the encoder as a static feature extractor. It encompasses two methods: linear probing (LP), which trains a shallow classifier to test the linear separability of original features (wen Yang et al., 2021; Turian et al., 2022; Wu et al., 2024); and zero-shot (ZS) evaluation, which performs direct inference via cross-modal alignment without downstream training, verifying the representations' intrinsic semantic understanding and generalization (Elizalde et al., 2023; Radford et al., 2023; Dinkel et al., 2026b; Mei et al., 2026).

B.2 Fine-tuning (FT)

FT unfreezes the pre-trained model for joint optimization on specific downstream tasks (Shams et al., 2024; Alex et al., 2024). By utilizing the model as an initialization backbone adapted to target data distributions, this approach explores the transferability limits and performance upper bounds, intuitively reflecting its practical deployment potential in complex tasks.

B.3 Direct Reconstruction (DR)

DR quantifies feature fidelity by restoring original waveforms from compressed representations via a decoder. Targeting reconstruction models like autoencoders (Wang and Huang, 2026; Dinkel et al., 2026a), neural codecs (Jiang et al., 2025; Wan et al., 2025), and vocoders (Lee et al., 2025; Park et al., 2025), it evaluates information compression efficiency and acoustic integrity by measuring the preservation of underlying physical attributes like timbre and phase.

C Summary of Audio Downstream Tasks

Audio downstream tasks and mainstream benchmarks are summarized in Table 2

C.1 Understanding Tasks

Audio understanding maps audio inputs to textual outputs, aligning condensed acoustic representations with textual semantic spaces to reflect human cognition. In multimodal paradigms, these tasks leverage text or video *prompts* as auxiliary instructions to guide specific auditory comprehension. These tasks are typically evaluated using diverse benchmarks, such as LibriSpeech (Panayotov et al., 2015) for automatic speech recognition, MMSU (Wang et al., 2026b) for audio question answering, and Clotho (Drossos et al., 2020) for automated captioning. Classification (Cls) fundamentally encompasses tasks such as acoustic scene classification (ASC), sound event detection (SED),

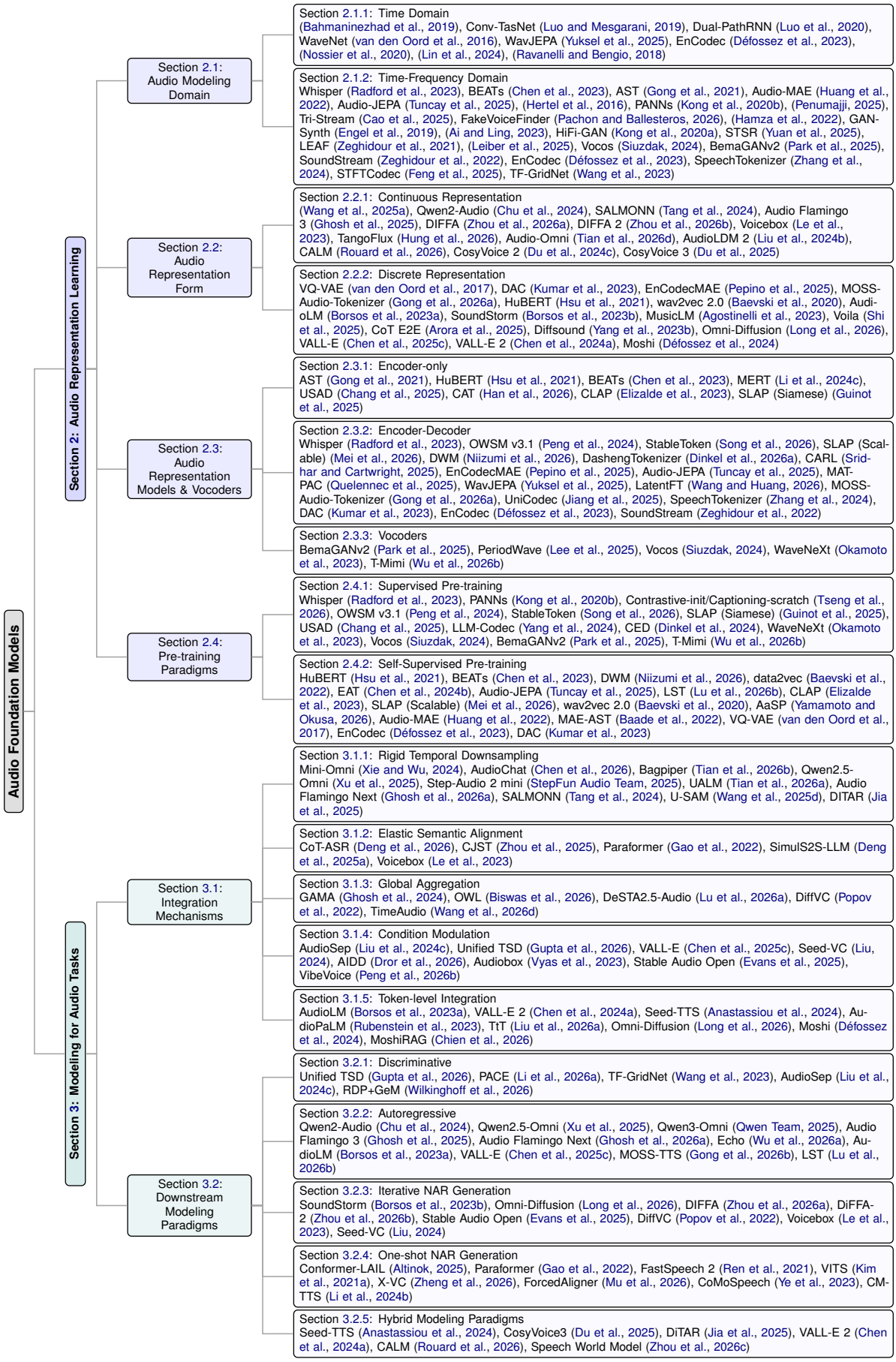


Figure 3: Taxonomy of Audio Representation Learning and Downstream Task Modeling Paradigms.

Task	Benchmarks
Understanding Tasks	
Cls *Classification*	AudioSet(Gemmeke et al., 2017), TUT Urban Acoustic Scenes 2018(Mesaros et al., 2018), X-ARES(Zhang et al., 2025), BirdSet(Rauch et al., 2025), ASVspoof 5(Wang et al., 2026h)
ASR *Automatic Speech Recognition*	Librispeech(Panayotov et al., 2015), GigaSpeech(Chen et al., 2021a), FLEURS(Conneau et al., 2023), ContextASR-Bench(Wang et al., 2025b)
S2TT *Speech-to-Text Translation*	MuST-C(Di Gangi et al., 2019), CoVoST 2(Wang et al., 2021), LongSpeech(Yang et al., 2026b)
AQA *Audio Question Answering*	MMAR(Ma et al., 2026b), MMSU(Wang et al., 2026b), MMAU-Pro(Kumar et al., 2026a), MMAU(Sakshi et al., 2025)
AAC *Automated Audio Captioning*	AudioCaps(Kim et al., 2019), AudioSetCaps(Bai et al., 2025), Clotho(Drossos et al., 2020), BRACE(Guo et al., 2026a)
ATR *Audio-Text Retrieval*	ReasonAudio(Zhang et al., 2026b), MARBLE(Yuan et al., 2023)
SD *Speaker Diarization*	SDBench(Durmus et al., 2025), M3SD(Wu, 2025)
Generation Tasks	
TTS *Text to Speech*	TTSDS2(Minixhofer et al., 2026), EmergentTTS-Eval(Manku et al., 2026), Seed-TTS-Eval(Anastassiou et al., 2024), LibriTTS(Zen et al., 2019), LibriTTS-R(Koizumi et al., 2023)
TTA *Text to Audio*	TTA-Bench(Wang et al., 2026e), AudioEval(Wang et al., 2026f)
VTA *Video to Audio*	Vggsound(Chen et al., 2020), VidAudio-Bench(Zhang et al., 2026c)
TTM *Text to Music*	FakeMusicCaps(Comanducci et al., 2025), MusicCaps(Agostinelli et al., 2023)
ZVC *Zero-shot Voice Cloning*	ClonEval(Christop et al., 2025), VCTK(Yamagishi et al., 2019)
T2ST *Text-to-Speech Translation*	–
Conversion Tasks	
VC *Voice Conversion*	VCC2020(Zhao et al., 2020), SVCC2023(Huang et al., 2023)
SE *Speech Enhancement*	DNS Challenge(Dubey et al., 2024), URGENT SE Challenge(Saijo et al., 2025)
SS *Speech Separation*	WSJ0-mix(Hershey et al., 2016), LibriMix(Cosentino et al., 2020)
AI *Audio Inpainting*	–
S2ST *Speech-to-Speech Translation*	CVSS(Jia et al., 2022), Emphassess(de Seyssel et al., 2024)
Interaction Tasks	
FSD *Full-duplex Spoken Dialogue*	Full-Duplex-Bench(Lin et al., 2025a), Full-Duplex-Bench-v2(Lin et al., 2026b), Full-Duplex-Bench-v3(Lin et al., 2026a), FD-Bench(Peng et al., 2025b)
SDS *Spoken Dialogue System*	ParaS2S(wen Yang et al., 2026), SD-Eval(Ao et al., 2024)

Table 2: Overview of Speech and Audio Tasks and Representative Benchmarks.

speaker identification (SID), speech emotion recognition (SER), and other tasks that output classification labels.

C.2 Generation Tasks

Conditioned on textual, visual, or acoustic signals, audio generation tasks employ deep generative models (*e.g.*, autoregressive or diffusion) to synthesize novel, high-fidelity, diverse, and highly controllable audio content, encompassing human speech, natural sound effects, and music. Key evaluation benchmarks include Seed-TTS-Eval (Anastassiou et al., 2024) for text-to-speech, TTA-Bench (Wang et al., 2026e) for text-to-audio generation, and VidAudio-Bench (Zhang et al., 2026c) for video-to-audio synthesis. Additionally, T2ST and S2TT can share benchmarks.

C.3 Conversion Tasks

Audio conversion tasks leverage source audio as a semantic and temporal backbone. Guided by text, audio conditions, or implicit objectives, they perform feature disentanglement and attribute reconstruction within acoustic latent spaces, yielding signals that preserve source content fidelity while successfully capturing target attributes. Common tasks like voice conversion, speech enhancement, and speech-to-speech translation are evaluated on benchmarks such as SVCC2023 (Huang et al., 2023), DNS Challenge (Dubey et al., 2024), and CVSS (Jia et al., 2022). Particularly, Audio inpainting lacks dedicated benchmarks, relying on synthetic gaps across general datasets.

C.4 Interactive Tasks

Audio interaction tasks facilitate real-time, bidirectional, and multi-turn human-machine communication via continuous audio streams. Driven by natural speech and enriched with paralinguistic and emotional features, they leverage full-duplex mechanisms for seamless listening and speaking, ultimately realizing fluent spoken dialogues. To assess interactive capabilities, researchers employ spoken dialogue system benchmarks like ParaS2S (wen Yang et al., 2026) and full-duplex evaluation suites such as Full-Duplex-Bench-v3 (Lin et al., 2026a).

D Summary of Audio Representation Models

D.1 Encoder-Only

Model	Scale	Audio Modeling Domain	Audio Rep. Form	Pre-training	Architecture	Data			Tasks	Eval
						SP	SO	MU		
PANNs (Kong et al., 2020b)	∼81M	T+TF	Cont.	SL	CNN	✓	✓	✓	CLS	LP, FT
PASE+ (Ravanelli et al., 2020)	–	T	Cont.	SSL (Pred+dist)	CNN+RNN	✓			ASR	LP, FT
wav2vec 2.0 (Baevski et al., 2020)	95M/317M	T	Cont.	SSL (Cont+Pred)	CNN+Transformer	✓			ASR, CLS	FT
AST (Gong et al., 2021)	86M	TF	Cont.	SL	ViT	✓	✓	✓	CLS	FT
HuBERT (Hsu et al., 2021)	95M/317M/964M	T	Cont.	SSL (Pred)	CNN+Transformer	✓			ASR	FT
PaSST (Koutini et al., 2022)	87M	TF	Cont.	SL	ViT	✓	✓	✓	CLS	FT
WavLM (Chen et al., 2022)	94.70M/316.62M	T	Cont.	SSL (Pred)	CNN+Transformer	✓			ASR, SS, CLS, SD	LP, FT
data2vec (Baevski et al., 2022)	∼90M/∼300M	T	Cont.	SSL (dist+Pred)	CNN+Transformer	✓			ASR, CLS	FT
BEST-RQ (Chiu et al., 2022)	0.6B	TF	Cont.	SSL (Pred)	Conformer	✓			ASR	FT
CLAP (Elizalde et al., 2023)	1.9B	TF	Cont.	SSL (Cont)	PANNs+BERT	✓	✓	✓	CLS	ZS, LP, FT
M2D (Niizumi et al., 2023)	86M	TF	Cont.	SSL (Pred+dist)	ViT	✓	✓	✓	CLS	LP+FT
BEATs (Chen et al., 2023)	∼90M	TF	Cont.	SSL (Pred)	ViT	✓	✓	✓	CLS	FT
MERT (Li et al., 2024c)	95M/330M	T	Cont.	SSL (Pred+Reco)	CNN+Transformer			✓	CLS	LP
CED (Dinkel et al., 2024)	5.5M/10M/22M/86M	TF	Cont.	SSL (Reco) + SL (dist)	ViT		✓		CLS	LP
DTF-AT (Alex et al., 2024)	69M	TF	Cont.	SL	ViT	✓	✓	✓	CLS	FT
SSAMBA (Shams et al., 2024)	7M/26M/99M	TF	Cont.	SSL (Cont+Reco)	Bi-Mamba	✓	✓	✓	CLS	FT
SLAP (Siamese) (Guinot et al., 2025)	193M	TF	Cont.	SSL (self-dist)	HTS-AT			✓	ATR, CLS	LP, ZS
USAD (Chang et al., 2025)	24/94/330M	TF	Cont.	SL (dist)	Transformer	✓	✓	✓	ASR, PR, SD, CLS	LP, FT
GLAP (Dinkel et al., 2026b)	∼90M	TF	Cont.	SSL (Cont)	Dasheng + Sonar + MLP	✓	✓	✓	ATR, CLS	ZS
SONAR (Zhang et al., 2026d)	∼90M	TF	Cont.	SSL (Pred+self-dist)	BEATs iter3+	✓	✓	✓	CLS	LP, FT
SPEAR (Yang et al., 2026e)	94M/327M/600M	TF	Cont.	SSL (Pred) + SL (dist)	Zipformer	✓	✓	✓	ASR, PR, CLS, SE, SS, VC	LP, FT
Contrastive-init/Captioning-scratch (Tseng et al., 2026)	∼70M	TF	Cont.	SL	Zipformer-M	✓	✓	✓	CLS, ATR, ACC, AQA	LP
CAT (Han et al., 2026)	–	T	Cont.	SSL (self-dist)	Transformer	✓	✓	✓	CLS, ACC	FT
UTS-Multi-Task (Zhou et al., 2026d)	–	TF	Cont.	SL	Zipformer-M	✓	✓	✓	CLS, ATR, ACC, AQA	LP
HEAR (Kawano and Sasaki, 2026)	15M	TF	Cont.	SSL (Pred)	Transformer		✓		CLS	LP, FT

Audio Rep. Form: **Cont.**: Continuous, **Disc.**: Discrete.
Pre-training: **SSL**: Self-supervised learning, **SL**: Supervised learning, **USL**: Unsupervised learning. **Pred**: Predictive, **Cont**: Contrastive, **Reco**: Reconstructive, **dist**: distillation, **self-dist**: self-distillation.
Data: **SP**: Speech, **SO**: Sound, **MU**: Music.
Tasks: **CLS**: Classification, **ASR**: Automatic Speech Recognition, **SD**: Speaker Diarization, **PR**: Phoneme Recognition, **AQA**: Audio Question Answering, **ATR**: Audio-Text Retrieval, **AAC**: Automated Audio Captioning, **SE**: Speech Enhancement, **SS**: Speech Separation, **VC**: Voice Conversion.
Eval: **LP**: Linear Probing, **ZS**: Zero-shot, **FT**: Fine-tuning.

Table 3: Overview of Encoder-only Audio Representation Learning Models

D.2 Encoder-Decoder

Model	Scale	Audio Modeling Domain	Audio Rep. Form	Pre-training	Architecture			Data			Tasks	Eval
					Encoder	Quantizer	Decoder	SP	SO	MU		
Cross-modal Encoder-Decoder												
Whisper (Radford et al., 2023)	39M/74M/244M/769M/1.55B	TF	Cont	SL	Transformer	-	Transformer	✓			ASR, S2TT	ZS
OWSM v3.1 (Peng et al., 2024)	101M/367M/1.02B	TF	Cont	SL	E-Branchformer	-	Transformer	✓			ASR, S2TT, SLU, CLS	LP
StableToken (Song et al., 2026)	320M	TF	Disc	SL	Transformer	Voting-LFQ	Transformer	✓			ASR, CLS, TTS	FT, DR
SLAP (Scalable) (Mei et al., 2026)	∼2.6b	TF	Cont	SSL (Cont+Pred)	Audio Encoder + Text Encoder	-	Transformer		✓		ATR, CLS, AAC	ZS, LP, FT
Autoencoders												
MAE-AST (Baade et al., 2022)	-	TF	Cont	SSL (Reco)	ViT	-	ViT	✓	✓	✓	CLS	FT
Audio-MAE (Huang et al., 2022)	22M/86M/304M	TF	Cont	SSL (Reco)	ViT	-	Transformer	✓	✓	✓	CLS, REC	FT, DR
EnCodecMAE (Pepino et al., 2025)	43.6M/86.6/261.6M	TF	Cont	SSL (dist+Pred)	Transformer	-	Transformer	✓	✓	✓	CLS, ASR	LP
EAT (Chen et al., 2024b)	88M	TF	Cont	SSL (dist)	ViT + CNN	-	CNN	✓	✓	✓	CLS	LP, FT
CARL (Sridhar and Cartwright, 2025)	-	TF	Cont	SL/SSL (Reco)	AudioMAE + Slot Transformer	-	MLP + CNN + Transformer		✓		CLS	LP
MATPAC (Quelennec et al., 2025)	∼86M	TF	Cont	SSL (Pred+dist)	AST	-	Transformer	✓	✓	✓	CLS	LP
Audio-JEPA (Tuncay et al., 2025)	85.4M	TF	Cont	SSL (dist)	ViT	-	ViT	✓	✓	✓	CLS	LP
WavJEPA (Yuksel et al., 2025)	∼90/300M	T	Cont	SSL (dist)	Wav2Vec 2.0 + ViT	-	ViT	✓	✓	✓	CLS	LP
AaSP (Yamamoto and Okusa, 2026)	87M	TF	Cont	SSL (Pred+Cont)	ViT	-	Cross-attention Blocks	✓	✓	✓	CLS	LP, FT
DashengTokenizer (Dinkel et al., 2026a)	∼800M	TF	Cont	SSL (Reco)	MiDasheng + 2D CNN	-	Vocos	✓	✓	✓	CLS, REC, TTA, SE	LP, DR
DWM (Niizumi et al., 2026)	∼86M	TF	Cont	SSL (Pred)	ViT	-	Transformer	✓	✓	✓	CLS	LP, FT
LatentFT (Wang and Huang, 2026)	-	T/F	Cont	SSL (Reco)	DAC / 1D U-Net / MLP	-	1D U-Net			✓	MG	DR
Neural Audio Codec												
VQ-VAE (van den Oord et al., 2017)	-	T	Disc	SSL (Reco)	CNN	VQ	CNN	✓			REC, CLS	LP, DR
SoundStream (Zeghidour et al., 2022)	8.4M	T	Disc	SL	CNN	RVQ	CNN	✓		✓	REC, SE	DR
EnCodec (Défossez et al., 2023)	-	T	Disc	SSL (Reco)	CNN + LSTM	RVQ	CNN + LSTM	✓	✓	✓	REC	DR
HiFi-Codec (Yang et al., 2023a)	-	T	Disc	SSL (Reco)	CNN + LSTM	Group-RVQ	CNN + LSTM	✓			REC	DR
DAC (Kumar et al., 2023)	76M	T	Disc	SSL (Reco)	CNN	RVQ	CNN	✓	✓	✓	REC	DR
SpeechTokenizer (Zhang et al., 2024)	∼100M	T	Disc	SSL (Reco+dist)	CNN + BiLSTM	RVQ	CNN + BiLSTM	✓			TTS	DR
SemantiCodec (Liu et al., 2024a)	75M	TF	Disc	SSL (Reco)	AudioMAE + BiLSTM	VAE	Diffusion + Vocoder	✓	✓	✓	CLS	LP, DR
Single-Codec (Li et al., 2024a)	-	TF	Disc	SSL (Reco)	Conformer + CNN	VQ	BiLSTM	✓			TTS	DR
Spectral Codecs (Langman et al., 2025)	65M	TF	Disc	SSL (Reco)	CNN	RVQ	HiFi-GAN	✓			TTS	DR
LLM-Codec (Yang et al., 2024)	160M	T	Disc	USL (GAN) + SL (dist)	CNN	RVQ	Transformer	✓	✓		CLS, TTS, SE	ZS, DR
SNAC (Siuzdak et al., 2024)	54.5M/19.8M	T	Disc	SSL (Reco)	CNN	RVQ	CNN	✓	✓	✓	REC	DR
UniCodec (Jiang et al., 2025)	274M	T	Disc	SSL (Pred)	ConvNet + Transformer + MoE	SimVQ	Transformer	✓	✓	✓	REC, CLS	DR
STFTCodec (Feng et al., 2025)	36M∼42M	TF	Disc	SSL (Reco)	ConvNeXt V2 + Self-Attention	RVQ	ConvNeXt V2 + Self-Attention	✓			REC	DR
DualCodec (Li et al., 2025a)	∼628M	T	Disc	SSL (Reco)	CNN + ResNet	RVQ	CNN	✓			TTS	DR
SpecTokenizer (Wan et al., 2025)	450K/70M	TF	Disc	SSL (Reco)	CNN	RVQ	RNN	✓			REC, ASR	DR
UniSRCodec (Zhang et al., 2026e)	-	TF	Disc	SSL (Reco)	Open-MagViT2	SimVQ	Open-MagViT2 + BigVGAN-v2	✓	✓	✓	CLS	LP, DR
MOSS-Audio-Tokenizer (Gong et al., 2026a)	1.6b	T	Disc	SSL (Reco)	Transformer	RVQ	Transformer	✓	✓	✓	TTS, ASR	DR, FT

Form: **Cont**: Continuous, **Disc**: Discrete.
Pre-training: **SSL**: Self-Supervised Learning, **SL**: Supervised Learning, **USL**: Unsupervised Learning, **Pred**: Predictive, **Cont**: Contrastive, **Reco**: Reconstructive, **dist**: distillation.
Data: **SP**: Speech, **SO**: Sound, **MU**: Music.
Tasks: **CLS**: Classification, **ASR**: Automatic Speech Recognition, **TTS**: Text-to-Speech, **TTA**: Text-to-Audio, **ATR**: Audio-Text Retrieval, **AAC**: Automated Audio Captioning, **MG**: Music Generation, **SE**: Speech Enhancement, **REC**: Reconstruction, **S2TT**: Speech-to-Text Translation, **SLU**: Spoken Language Understanding.
Eval: **LP**: Linear Probing, **ZS**: Zero-shot, **FT**: Fine-tuning, **DR**: Direct Reconstruction.

Table 4: Overview of Encoder-Decoder Audio Representation Learning Models

D.3 Vocoder

Model	Scale	Audio Modeling Domain	Input Rep. Form	Pre-training Method	Architecture	Data			Tasks	Eval
						SP	SO	MU		
WaveNet (van den Oord et al., 2016)	-	T → T	Disc	SL	CNN	✓		✓	REC, ASR	DR, FT
WaveRNN (Kalchbrenner et al., 2018)	-	TF → T	Cont	SL	RNN + CNN	✓			REC	DR
WaveGlow (Prenger et al., 2019)	-	TF → T	Cont	SL	CNN	✓			REC	DR
GANSynth (Engel et al., 2019)	-	TF → T	Cont	SL	CNN	✓			REC	DR
WaveGrad (Chen et al., 2021b)	15M/23M	TF → T	Cont	SL	UBlock + DBlock	✓			REC	DR
DiffWave (Kong et al., 2021)	2.64M/6.91M	TF → T	Cont	USL	Diffusion	✓			REC, UG	DR
HiFi-GAN (Kong et al., 2020a)	0.92M/1.46M/13.92M	TF → T	Cont	SL	CNN	✓			REC	DR
Fre-GAN (Kim et al., 2021b)	-	TF → T	Cont	SL	HiFi-GAN	✓			REC	DR
UnivNet (Jang et al., 2021)	4M/14.86M	TF → T	Cont	SL	MelGAN	✓			REC, TTS	DR, FT
iSTFTNet (Kaneko et al., 2022)	13.26M	TF → T	Cont	SL	CNN + DSP	✓			REC, TTS	DR
BigVGAN (Lee et al., 2023)	14M/112M	TF → T	Cont	SL	HiFi-GAN	✓			REC	DR
Vocos (Siuzdak, 2024)	13.5M	Token / TF → T	Disc/Cont	SL	ConvNeXt	✓			REC, TTS, MG	DR, ZS
HiFTNet (Li et al., 2023b)	17.7M	TF → T	Cont	SL	LSTM + CNN	✓			REC	DR
APNet2 (Du et al., 2024a)	-	TF → T	Cont	SL	ConvNeXt V2	✓			REC	DR
WaveNeXt (Okamoto et al., 2023)	-	TF → T	Cont	SL	ConvNeXt + Linear Layer	✓			REC	DR
PeriodWave (Lee et al., 2025)	29.7M	TF → T	Cont	SL	U-Net	✓			REC	DR
BemaGANv2 (Park et al., 2025)	∼14M	TF → T	Cont	SL	CNN	✓			REC	DR
T-Mimi (Wu et al., 2026b)	40.8M	Token → T	Disc	SL	Transformer	✓			REC	DR

Input Rep. Form: **Cont**: Continuous, **Disc**: Discrete. **Pre-training**: **SL**: Supervised Learning, **USL**: Unsupervised Learning.
Data: **SP**: Speech, **SO**: Sound, **MU**: Music.
Tasks: **REC**: Reconstruction, **ASR**: Automatic Speech Recognition, **TTS**: Text-to-Speech, **UG**: Unconditional Generation, **MG**: Music Generation.
Eval: **ZS**: Zero-shot, **FT**: Fine-tuning, **DR**: Direct Reconstruction.

Table 5: Overview of Vocoders

E Training Paradigms for Audio Downstream Task Models

Building a robust downstream audio system requires distinct training paradigms to align acoustic features with semantic goals. Depending on the architecture and data availability, these paradigms range from massive multimodal pre-training to training-free adaptations.

E.1 Pre-training

Pre-training refers to constructing a general-purpose foundation model with strong generalization capability using large-scale universal data. In multimodal audio modeling, it is divided into four primary approaches:

Native Multimodal End-to-End Training Initialized from scratch, this paradigm maps all modalities into a shared semantic space from the outset. Joint modeling under a unified objective enables native multimodal capability, exemplified by GPT-4o (Hurst et al., 2024) and Gemini2.5(Gemini Team, 2025). Despite prohibitive costs and severe convergence challenges, SODA (Manakul et al., 2026) unifies audio and text tokens to pre-train a Transformer backbone entirely from scratch.

Cross-modal Alignment End-to-End Pre-training Relying on pre-trained LLMs and audio extractors, this paradigm uses bridging mechanisms to map audio representations into the dominant text modality's continuous latent or discrete token space. It typically adopts multi-stage training: models like Qwen2.5-Omni (Xu et al., 2025) and Qwen3-Omni (Qwen Team, 2025) initially execute cross-modal alignment before conducting global end-to-end joint optimization. Alternatively, this paradigm can treat one modality as the generative target and the other as a guiding condition, as exemplified by the text-guided speech generation in DiTAR (Jia et al., 2025).

Modular Pre-training This paradigm decomposes complex systems into functionally independent sub-networks trained separately with local objectives, avoiding global gradient sharing. Successive modules are conditioned on preceding outputs to align multimodal capabilities with text. For instance, Seed-TTS (Anastassiou et al., 2024) independently pre-trains a speech tokenizer and vocoder, subsequently concatenating audio and text tokens to pre-train a cascaded Transformer-DiT backbone. Similarly, Moshi (Défossez et al., 2024) separately pre-trains a text backbone and an audio

codec, later leveraging extracted audio tokens to continually train the backbone, thereby unifying the text and audio spaces.

E.2 Post-training

After unsupervised distribution matching, post-training introduces explicit supervision and human preferences to directionally distort the model's generation trajectory for real-world interactions.

Supervised Fine-Tuning (SFT) employs high-quality instruction-output pairs for either full or parameter-efficient updates. It activates pre-trained knowledge, enabling models to strictly follow human instructions and master specific downstream tasks and interaction paradigms. Recent frameworks like MoshiRAG (Chien et al., 2026) leverage SFT to integrate retrieval-augmented generation for dialog systems, while CoT-ASR (Deng et al., 2026) utilizes it to incorporate multi-step reasoning into speech recognition.

Reinforcement Learning (RL) leverages human or AI feedback to construct reward signals via algorithms like RLHF, DPO, or GRPO. By maximizing expected returns, RL compensates for SFT's limitations in optimizing subjective criteria, thereby enhancing generation quality, ensuring human-value alignment, and effectively mitigating hallucinations. To unlock advanced cognitive abilities, systems such as Step-Audio-R1 (Tian et al., 2025) and Audio-DeepThinker (He et al., 2026) employ RL-based reasoning strategies to systematically enhance performance in complex audio question-answering tasks.

E.3 Supervised Training from Scratch

Supervised Training from Scratch optimizes a randomly initialized backbone for a specific downstream task using labeled data. However, its performance ceiling is inherently bounded by the scale of human annotations, resulting in poor transferability. Despite these transferability limits, it remains a robust choice for specialized generation and conversion tasks, as demonstrated in Stable Audio Open (Evans et al., 2025) for text-to-music generation and X-VC (Zheng et al., 2026) for voice conversion.

E.4 Training-free

Training-Free approaches yield decisions or generations without any gradient updates during downstream adaptation. For instance, RDP (Wilkinghoff et al., 2026) detects anomalous sounds by comparing feature distances between test and normal audio, whereas Audio-CoT (Ma et al., 2025) employs inference-time prompting to elicit the model's internalized reasoning capabilities.

F Summary of Audio Downstream Task Models

F.1 Overview of Pre-trained Models

Model	Scale	Audio Representation Model	Base Model	DeTokenizer / Vocoder	Cond. Method	Modeling Paradigm	Pre-training	Post-training	Tasks	Downstream
GSLM (Lakhotia et al., 2021)	-	CPC / wav2vec 2.0 / HuBERT + k-means	Transformer	Tacotron-2 + WaveGlow	TI	AR	Modular	-	Gen	ZVC
VALL-E (Chen et al., 2025c)	∼300M	EnCodec	Decoder-only Transformer	EnCodec Decoder	TI + CM	AR, Iter. NAR	Modular	-	Gen, Conv	TTS, ZVC, SE
AudioPaLM (Rubenstein et al., 2023)	8B	USM-v2 / USM-v1 / w2v-BERT	PaLM/PaLM-2	SoundStorm / SoundStream Decoder	RTD + TI	AR	CMA-E2E	SFT	Und, Gen	ASR, S2TT, S2ST, TTS
AudioLM (Borsos et al., 2023a)	∼0.9B	w2v-BERT + SoundStream	Decoder-only Transformer	SoundStream Decoder	TI + RTD	AR	Modular	-	Gen	Cls
Qwen-Audio (Chu et al., 2023)	∼8B	Whisper-large-v2 Encoder	Qwen-7B	-	RTD	AR	CMA-E2E	SFT	Und	ASR, S2TT, AAC, Cls
SpeechGPT (Zhang et al., 2023)	13B	HuBERT	Transformer	HiFi-GAN	TI	AR	CMA-E2E	SFT	Int, Und, Gen	ASR, SDS, TTS
MusicLM (Agostinelli et al., 2023)	∼1.29B	w2v-BERT + SoundStream + MuLan	Decoder-only Transformer	SoundStream Decoder	TI	AR	Modular	-	Gen	TTM
SoundStorm (Borsos et al., 2023b)	350M	w2v-BERT + SoundStream	Conformer	SoundStream Decoder	TI	Iter. NAR	Modular	-	Gen	TTS, ZVC
Audiobox (Vyas et al., 2023)	330M	EnCodec	Transformer	EnCodec Decoder	CM	Iter. NAR	Modular	SFT	Gen, Conv	TTS, TTA, ZVC, AI
Diffsound (Yang et al., 2023b)	∼160M	VQ-VAE	Transformer	MelGAN	CM	Iter. NAR	Modular	-	Gen	TTA
SeamlessM4T (Communication et al., 2025)	1.2B/2.3B	w2v-BERT 2.0	Seamless M4T-NLLB	HiFi-GAN	RTD + CM	AR	Modular	SFT	Und	ASR, S2TT, T2ST, S2ST
Voicebox (Le et al., 2023)	∼360M	Mel-Spectrogram	Transformer	HiFi-GAN	ESA + CM	Iter. NAR	Modular	-	Gen, Conv	TTS, ZVC, SE
LTU (Gong et al., 2024)	∼7B	AST	LLaMA-7B	-	TI	AR	CMA-E2E	SFT	Und	Cls, AAC, AQA
VALL-E 2 (Chen et al., 2024a)	-	EnCodec	Decoder-only Transformer	Vocos	TI + CM	AR, Iter. NAR	Modular	-	Gen, Conv	TTS, ZVC
Qwen2-Audio (Chu et al., 2024)	8.2B	Whisper-large-v3 Encoder	Qwen-7B	-	RTD	AR	CMA-E2E	SFT, RL	Und	ASR, S2TT, Cls
AudioLDM 2 (Liu et al., 2024b)	712M/346M	AudioMAE + VAE	GPT-2 + Transformer-UNet	HiFi-GAN	CM	AR + Iter. NAR	Modular	-	Gen	TTA, TTM, TTS
SALMONN (Tang et al., 2024)	13B	Whisper-large-v2 Encoder + BEATs	Vicuna-13B	-	RTD	AR	CMA-E2E	SFT	Und	ASR, S2TT, AAC
CosyVoice 2 (Du et al., 2024c)	-	FSQ-SenseVoice	Qwen2.5-0.5B + Transformer-UNet	HiFi-GAN	CM + TI	AR + Iter. NAR	Modular	SFT, RL	Gen	TTS, ZVC
Moshi (Défossez et al., 2024)	∼7.6B	Mimi	RQ-Transformer	Mimi Detokenizer	RTD + TI	AR	Modular	SFT	Int	FSD
Seed-TTS (Anastassiou et al., 2024)	-	Custom Speech Tokenizer	Transformer+DiT	Acoustic Vocoder	TI	AR + Iter. NAR	Modular	SFT, RL	Gen, Conv	TTS, VC
GLM-4-Voice (Zeng et al., 2024)	9B	Whisper-large-v3 Encoder + VQ	GLM-4-9B-Base	HiFi-GAN	TI	AR	Modular	SFT	Int, Gen	AQA, TTS, SDS
GAMA (Ghosh et al., 2024)	7B	AST	LLaMA-2-7B	-	GA	AR	Modular	SFT	Und	Cls, AQA, AAC
GPT-4o (Hurst et al., 2024)	-	-	Transformer	-	TI	AR	NM-E2E	-	Und, Gen, Int, Conv	-
Baichuan-Audio (Li et al., 2025b)	7B	Baichuan-Audio-Tokenizer	Baichuan 7B + Flow-matching Decoder	HiFi-GAN	TI	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	ASR, S2TT, TTS, AQA
Qwen2.5-Omni (Xu et al., 2025)	7B	Qwen2 Audio Encoder	Qwen2.5-7B + Flow-matching Decoder	BigVGAN	TI + RTD	AR + Iter. NAR	CMA-E2E	SFT, RL	Und, Gen, Int	ASR, S2TT, AQA, Cls
Kimi-Audio (KimiTeam et al., 2025)	7B	GLM-4-Voice Tokenizer + Whisper Encoder	Qwen2.5-7B + Flow-matching Decoder	BigVGAN	TI + RTD	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	ASR, AQA, Cls, AAC
Gemini 2.5 (Gemini Team, 2025)	-	-	-	-	TI	AR	NM-E2E	SFT, RL	Und, Gen, Int	ASR, S2TT, TTS, SDS
Step-Audio 2 mini (StepFun Audio Team, 2025)	7B	Qwen2 Audio Encoder	Qwen2.5-7B + Flow Matching Decoder	HiFi-GAN	RTD + TI	AR + Iter. NAR	CMA-E2E	SFT, RL	Und, Gen, Int	ASR, S2TT, AQA, TTS
FireRedTTS-2 (Xie et al., 2025a)	-	Whisper-based Encoder + RVQ	Qwen2.5	Vocos	TI	AR	Modular	SFT	Gen, Conv	ZVC, SDS, TTS
Qwen3-Omni (Qwen Team, 2025)	33B	AuT	Qwen3	Codec2Wav	RTD	AR	CMA-E2E	SFT, RL	Und, Gen, Int	ASR, S2TT, ATR, AQA
LFM2-Audio (Amini et al., 2025)	1.5B	FastConformer	LFM2-1.2B + RQ-Transformer	Vocos	RTD	AR	CMA-E2E	SFT, RL	Und, Gen, Int	ASR, TTS, SDS
Mimo-Audio (LLM-Core Xiaomi, 2025)	7B	Mimo-Audio-Tokenizer	Mimo-7B-Base	Mimo-Audio-Tokenizer / Vocos	RTD + TI	AR	CMA-E2E	SFT	Und, Gen, Conv, Int	ASR, TTS, AQA, VC
Audio Flamingo 3 (Ghosh et al., 2025)	7B	AF-Whisper Encoder	Qwen2.5-7B	Neural Audio Codec	RTD	AR	CMA-E2E	SFT	Und, Gen, Int	ASR, Cls, AAC
GOAT-SLM (Chen et al., 2025b)	7B	Whisper-small Encoder	TeleChat2-7B + Flow-matching Decoder	Vocoder	RTD	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	AQA, SDS, TTS
CoT E2E (Arora et al., 2025)	1.7B	ESPnet-Codec + XEUS	SpeechLM	ESPnet-Codec	TI	AR	CMA-E2E	SFT	Int	SDS
CosyVoice 3 (Du et al., 2025)	0.5B, 1.5B	FSQ-MinMo	Transformer	DiT	TI + CM	AR + Iter. NAR	Modular	SFT, RL	Gen	TTS, ZVC
Voila (Shi et al., 2025)	-	Voila Tokenizer	Voice-language LLM backbone + Audio Transformer	Voila Detokenizer	TI	AR	CMA-E2E	SFT, RL	Und, Gen, Int, Conv	ASR, TTS, SDS, S2TT
DITAR (Jia et al., 2025)	0.1B/0.4B/0.6B/1.0B	VAE	Causal Transformer + LocDiT	BigVGAN	RTD + CM	AR + Iter. NAR	CMA-E2E	SFT	Gen	TTS
HeartMula (Yang et al., 2026a)	3.3B / 7B	HeartCodec Tokenizer	Llama 3.2	HeartCodec Detokenizer	TI	AR	Modular	SFT, RL	Gen	TTM
SODA (Manakul et al., 2026)	135M/600M/1.7B/4B	Mimi Tokenizer	Qwen3-based	Mimi Detokenizer	TI	AR	NM-E2E	SFT	Gen, Conv, Und	TTS, ZVC, ASR
MOSS-TTS (Gong et al., 2026b)	1.7B / 8B	MOSS-Audio-Tokenizer	Decoder-only LLM	MOSS-Audio-Tokenizer	TI	AR	Modular	-	Gen	TTS, ZVC
AudioX (Tian et al., 2026c)	2.4B	AudioLDM 2 Encoder	DiT	-	GA + CM	Iter. NAR	CMA-E2E	-	Gen, Conv	TTA, VTA, TTM
Audio Flamingo Next (Ghosh et al., 2026a)	7B	AF-Whisper Encoder + 2-layer MLP	Qwen2.5-7B	Streaming TTS	RTD	AR	CMA-E2E	SFT, RL	Und, Int	ASR, SD, S2TT, Cls
Eureka-Audio (Zhang et al., 2026a)	1.7B	Whisper-based Encoder	Qwen3-1.7B-base	-	RTD	AR	CMA-E2E	SFT	Und	ASR, AAC, AQA
UALM (Tian et al., 2026a)	1.5B / 7B	Qwen2 Audio Encoder	Qwen2.5-7B	X-codec Decoder + VAE Decoder	RTD	AR	CMA-E2E	SFT, RL	Und, Gen, Int	ASR, TTA, SDS, AQA
OmniVoice (Zhu et al., 2026)	0.8B	Higgs-audio Tokenizer	Qwen3-0.6B	Higgs-audio Detokenizer	TI	Iter. NAR	Modular	-	Gen, Conv	TTS, ZVC, SE
Audio-Omni (Tian et al., 2026d)	7.9B	VAE	Qwen2.5-Omni-3B + DiT	VAE Decoder	CM	AR + Iter. NAR	Modular	-	Gen, Conv	AQA, TTA, TTM, VTA
Omni-Diffusion (Long et al., 2026)	7B	SenseVoiceSmall	Qwen-7B	GLM-4-Voice Decoder	RTD	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	ASR, TTS, AQA, SDS
Bagpiper (Tian et al., 2026b)	8B	Qwen3-Omni-Encoder + X-Codec	Qwen3-8B-Base	X-Codec Decoder	RTD + TI	AR	CMA-E2E	SFT	Und, Gen	ASR, AQA, TTS, TTA
AudioChat (Chen et al., 2026)	3.6B	DAC-VAE Encoder	Gemma 2 2B	DAC-VAE Decoder	RTD	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Conv	ASR, AAC, SD, TTS
Hello-Chat (Hou et al., 2026)	∼8B	MiDashengLM	Qwen2.5-7B-Instruct	CosyVoice 2	RTD	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	ASR, S2TT, Cls
DIFFA-2 (Zhou et al., 2026b)	8.77B	Whisper-large-v3 Encoder	LLaDA-8B-Instruct	-	RTD + GA	Iter. NAR	CMA-E2E	SFT, RL	Und	ASR, AQA
CALM (Rouard et al., 2026)	100M∼3B	VAE Encoder	Transformer / Helium-1	VAE Decoder	RTD +CM	AR + One-shot NAR	CMA-E2E	-	Gen	TTS, TTM
DIFFA (Zhou et al., 2026a)	8B	Whisper-small Encoder	LLaDA-8B-Instruct	-	RTD + GA	Iter. NAR	CMA-E2E	SFT	Und	ASR, AQA
VibeVoice (Peng et al., 2026b)	1.5B, 7B	Acoustic Tokenizer + Semantic Tokenizer	Qwen2.5	Acoustic Detokenizer	RTD + CM	AR + Iter. NAR	CMA-E2E	-	Gen	TTS, ZVC
TiT (Text-to-Talk) (Liu et al., 2026a)	1.5B, 3B	GLM-4-Voice Tokenizer	Qwen2.5-Base	GLM-4-Voice Decoder	TI	AR + Iter. NAR	CMA-E2E	SFT	Und, Gen, Int	AQA, ASR, AAC, TTS
OWL (Biswas et al., 2026)	7B	SAGE	LLaMA-2-7B	-	GA	AR	CMA-E2E	SFT	Und	Cls, AQA
DeSTA2.5-Audio (Lu et al., 2026a)	8.8B	Whisper-large-v3 Encoder	Llama3.1-8B-Instruct	-	GA	AR	Modular	SFT	Und	Cls, AQA, AAC, ASR
TangoFlux (Hung et al., 2026)	515M	Stable Audio Open VAE Encoder	MMDiT + DiT	Stable Audio Open VAE Decoder	CM	Iter. NAR	Modular	SFT, RL	Gen	TTA
SLM-SS (Li et al., 2026b)	-	WavLM-Large + EnCodec	Transformer	EnCodec Decoder	TI + CM	AR + One-shot NAR	Modular	SFT	Conv	SS, SE
UAF (Li et al., 2026c)	3B, 7B, 30B	-	Qwen3-Omni-30B-A3B-Instruct	-	-	AR	CMA-E2E	SFT	Und	Cls, ASR, AQA
LLM-ASR (Xie et al., 2026)	2.3B	Conformer	Qwen3-1.7B	-	RTD	AR	CMA-E2E	SFT	Und	ASR

Cond. Method: **GA**: Global Aggregation, **CM**: Condition Modulation, **TI**: Token-level Integration, **RTD**: Rigid Temporal Downsampling, **ESA**: Elastic Semantic Alignment.
Modeling Paradigm: **AR**: Autoregressive, **Iter. NAR**: Iterative Non-Autoregressive, **One-shot NAR**: One-shot Non-Autoregressive.
Pre-training: **CMA-E2E**: Cross-modal Alignment End-to-End Pre-training, **Modular**: Modular Pre-training, **NM-E2E**: Native Multimodal End-to-End Training.
Tasks: **Gen**: Generation, **Conv**: Conversion, **Und**: Understanding, **Int**: Interaction.
Understanding: **Cls**: Classification, **ASR**: Auto Speech Recognition, **S2TT**: Speech-to-Text Translation, **AQA**: Audio Question Answering, **AAC**: Automated Audio Captioning, **ATR**: Audio Text Retrieval, **SD**: Speaker Diarization.
Generation: **TTS**: Text to Speech, **ZVC**: Zero-shot Voice Cloning, **TTA**: Text to Audio, **VTA**: Video to Audio, **TTM**: Text to Music, **T2ST**: Text-to-Speech Translation.
Conversion: **VC**: Voice Conversion, **SE**: Speech Enhancement, **SS**: Speech Separation, **AI**: Audio Inpainting, **S2ST**: Speech-to-Speech Translation.
Interaction: **FSD**: Full-duplex Spoken Dialogue, **SDS**: Spoken Dialogue System.

Table 6: Overview of Audio Pre-training Models

F.2 Overview of Downstream Fine-tuning Models

Model	Scale	Audio Representation Model	Base Model	DeTokenizer / Vocoder	Cond. Mech.	Modeling Paradigm	Pre-training	Post-training	Tasks	Downstream
FastSpeech 2 (Ren et al., 2021)	~27M	Mel-Spectrogram	Feed-Forward Transformer	Parallel WaveGAN	ESA	One-shot NAR	Sup. Scratch		Gen	TTS
VITS (Kim et al., 2021a)	~12M	Mel-Spectrogram	Transformer	HiFi-GAN	ESA	One-shot NAR	Sup. Scratch		Gen	TTS
DiffVC (Popov et al., 2022)	-	Mel-Spectrogram	Transformer + U-Net	HiFi-GAN	GA	Iter. NAR	Sup. Scratch		Conv	VC
Paraformer (Gao et al., 2022)	41M	Mel-Spectrogram	Paraformer	-	ESA	One-shot NAR	Sup. Scratch		Und	ASR
MUSICGEN (Copet et al., 2023)	300M, 1.5B, 3.3B	EnCodec	Decoder-only Transformer	EnCodec	CM + TI	AR	Sup. Scratch		Gen	TTM
AST-SED (Li et al., 2023a)	-	AST	Bi-GRU	-	RTD	Disc.	-	SFT	Und	Cls
TF-GridNet (Wang et al., 2023)	2.1M - 14.5M	Mel-Spectrogram	TF-GridNet	iSTFT	-	Disc.	Sup. Scratch		Conv	SS, SE, SD
CoMoSpeech (Ye et al., 2023)	-	Mel-Spectrogram	FastSpeech-style FFT Transformer	HiFi-GAN	-	One-shot NAR	Sup. Scratch		Gen	TTS, TTM
Seed-VC (Liu, 2024)	~200M	Whisper-small Encoder / CAM++ / RMVPE	U-DiT	BigVGANv2	CM	Iter. NAR	Sup. Scratch		Conv	VC
AudioSep (Liu et al., 2024c)	238.6M	CLAP / CLIP	ResUNet	-	CM	Disc.	Sup. Scratch		Und	SS
Mini-Omni (Xie and Wu, 2024)	0.5B	Whisper-small Encoder	Qwen2-0.5B	SNAC	RTD	AR	-	SFT	Und, Gen, Int	ASR, TTS, AQA, SDS
Freeze-Omni (Wang et al., 2025c)	7B	CNN+Transformer	Qwen2-7B-Instruct	TiCodec	RTD	AR, One-shot NAR	-	SFT	Int	FSD, SDS, ASR, AQA
Tango 2 (Majumder et al., 2024)	866M	AudioLDM VAE	Tango	HiFi-GAN	CM	Iter. NAR	-	RL	Gen	TTA
CosyVoice (Du et al., 2024b)	~300M	S3 Tokenizer	Transformer + ResNet1D	HiFi-GAN	TI	AR + Iter. NAR	Sup. Scratch		Gen	TTS, ZVC
ARDiT (Liu et al., 2024d)	~130M	Contextual Mel Spectrogram Autoencoder	ARDiT	BigVGAN	RTD	AR + Iter. NAR	Sup. Scratch		Gen, Conv	TTS, ZVC, AI
Wav2Prompt (Deng et al., 2025b)	7B	Conformer	LLaMA-2 / BLOOMZ / Vicuna	-	ESA	AR	-	SFT	Und	ASR, S2TT, AQA
CM-TTS (Li et al., 2024b)	28.6M	Mel-Spectrogram	FastSpeech2-style Transformer	HiFi-GAN	-	One-shot NAR	Sup. Scratch		Gen	TTS
STSR (Yuan et al., 2025)	66.2M	MDCT Coefficient	U-Net	-	RTD	One-shot NAR	Sup. Scratch		Conv	SE
Tri-Stream (Cao et al., 2025)	-	Swin Transformer, HuBERT, openSMILE	GRU	-	GA	Disc.	Sup. Scratch		Und	Cls
Step-Audio-R1 (Tian et al., 2025)	32B	Qwen2 Audio Encoder	Qwen2.5-32B	-	RTD	AR	-	SFT, RL	Und, Int	AQA, SDS
Stable Audio Open (Evans et al., 2025)	1B	VAE	DiT	VAE Decoder	CM	Iter. NAR	Sup. Scratch		Gen	TTA, TTM
Music Flamingo (Ghosh et al., 2026b)	3B / 7B	-	Audio Flamingo 3	-	-	AR	-	SFT, RL	Und	ATR, ASR, AQA
PAC (FireRed-LLM) (Fu et al., 2026)	7B	-	FireRed-LLM-7B	-	-	AR	-	SFT, RL	Und	ASR
YingMusic-SVC (Chen et al., 2025a)	~200M	-	Seed-VC	BigVGAN	-	Iter. NAR	-	SFT, RL	Conv	VC
Omni-R1 (Rouditchenko et al., 2025)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	RL	Und	AQA
DAR-LALM (Sheng et al., 2026)	7B	-	Qwen2.5-Omni-7B / Qwen2-Audio-7B-Instruct	-	-	AR	-	SFT, RL	Und	AQA, Cls
Rebellion (Huang et al., 2025)	7B	-	Qwen2-Audio-7B	-	-	AR	-	SFT	Und	AQA
AsyncVoice Agent (Lin et al., 2025b)	-	-	GPT-4o	Azure TTS	-	-		Tr. Free	Int	AQA, FSD
AudioStory (Guo et al., 2025)	3B	Whisper-large-v3 Encoder	Qwen-2.5-3B-Instruct	TangoFlux	RTD + CM	AR + Iter. NAR	-	SFT	Gen, Und	TTA, AQA, AAC, VTA
AlignFormer (Fan et al., 2025)	4.3B	Conformer	Phi3.1-mini-instruct	HiFi-GAN	ESA	AR	-	SFT	Und	ASR, S2TT, AQA
Audio-Reasoner (Xie et al., 2025c)	8.4B	-	Qwen2-Audio-Instruct	-	-	AR	-	SFT	Und	AQA, Cls, S2TT
SoundMind-RL (Diao et al., 2025)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	SFT, RL	Und, Gen	AQA, TTA
Serenade (Violeta et al., 2025)	-	ContentVec, Prior Encoder, Style Encoder	1D U-Net	HIFIGAN, SIFIGAN	CM	Iter. NAR	Sup. Scratch		Conv	VC
Reflow-VC (Ren et al., 2025)	-	Average Voice Encoder, HuBERT-Soft, Speaker Encoder, VQ-VAE	U-Net	HiFi-GAN	CM	Iter. NAR	Sup. Scratch		Conv	VC
U-SAM (Wang et al., 2025d)	8B	Whisper Encoder, CED, MERT	LLaMA-3.1-8B	-	RTD	AR	-	SFT	Und	ASR, S2TT, AAC, Cls
Audio-CoT (Ma et al., 2025)	7B	-	Qwen2-Audio-7B-Instruct	-	-	AR		Tr. Free	Und	AQA, ASR, AAC
SimulS2S-LLM (Deng et al., 2025a)	7B/8B	Streaming Transformer	BLOOMZ-7B / Llama3-8B	HiFi-GAN	ESA	AR	-	SFT	Und, Conv	S2ST, S2TT
CJST (Zhou et al., 2025)	~110M	Conformer+VGG	Decoder-only Transformer	-	RTD + ESA	AR	-	SFT	Und	ASR
Conformer-LAIL (Altinok, 2025)	8.3B	Wav2Vec 2.0	Wav2Vec 2.0 Conformer Large	-	RTD	One-shot NAR	-	SFT	Und	ASR
HyPeR (Wang et al., 2026g)	-	Whisper Encoder	Qwen2-Audio-7B-Instruct	-	-	AR	-	SFT, RL	Und	AQA, S2TT, Cls, SD
Nemotron-0.6B INT4 k-quant (Banfic et al., 2026)	-	-	Nemotron-0.6B	-	-	AR	-	-	Und	ASR
Music Flamingo (Ghosh et al., 2026c)	7B	-	Audio Flamingo 3	-	-	AR	-	SFT, RL	Und	AQA, AAC, ATR
MDM-ASR (Yen et al., 2026)	180M / 1B	FastConformer	Non-causal Transformer Decoder	-	RTD + CM	Iter. NAR	-	-	Und	ASR
Audio-DeepThinker (He et al., 2026)	30B	-	Qwen3-Omni-30B-A3B-Instruct	-	-	AR	-	RL	Und	AQA
X-VC (Zheng et al., 2026)	539M	SAC Encoder, ERes2Net	Transformer	SAC Decoder	RTD + CM	One-shot NAR	Sup. Scratch		Conv	VC
MoshiRAG (Chien et al., 2026)	7B	-	Moshi	-	TI	AR	-	SFT	Und, Int	AQA, SDS, ATR
Speech World Model (SWM) (Zhou et al., 2026c)	~8B	WavLM	Qwen2-Audio, LLaMA3.1-8B	-	TI	Disc. + AR	-	SFT	Und	Cls
Granite-speech (Aronowitz et al., 2026)	8B	Conformer	Granite-3.3-8b-instruct	-	GA	AR	-	SFT	Und	ASR, SD, Cls
Projector-based LLM-ASR (Li et al., 2026d)	1.7B / 8B	Whistle, WavLM	Qwen3-Base	-	RTD + CM	AR	-	SFT	Und	ASR
Phoneme-based LLM-ASR (Li et al., 2026d)	1.7B / 8B	Whistle, WavLM	Qwen3-Base	-	ESA	AR	-	SFT	Und	ASR
STEMPHONIC (Wu et al., 2026c)	~1B	VAE	DiT	VAE	CM	Iter. NAR	-	SFT	Gen	TTM
TASU2 (Peng et al., 2026a)	~1.8B	SenseVoice-Small	Qwen2.5-1.5B	-	ESA	AR	-	SFT	Und	ASR, AAC
AEG (Yang et al., 2026d)	7B	-	Qwen2-Audio-7B	-	-	AR	-	SFT	Und	AQA
PE-FT LALMs (Yang et al., 2026c)	-	-	Qwen2.5-Omni / Kimi-Audio	-	-	AR	-	SFT	Und, Int	Cls, SDS
PROADS (Yan et al., 2026)	-	EzAudio Codec Encoder	EzAudio	EzAudio Codec Decoder	-	Iter. NAR	-	-	Gen	AI
ALARM-E (Grinberg and Shahmohammadi, 2026)	4B	Whisper Encoder, W2V-BERT-2.0, MuQ, SSLAM	Qwen3-4B-Thinking-2507	-	RTD	AR	-	SFT	Und	AQA
Ambiguity-Aware LALM (Yu et al., 2026)	7B	-	Qwen2-Audio-7B-Instruct	-	-	AR	-	SFT, RL	Und	Cls
RDP / RDP+GeM (Wilkinghoff et al., 2026)	-	OpenL3, BEATs, EAT, Dasheng	-	-	-	Disc.		Tr. Free	Und	Cls
StyleStream (Liu et al., 2026b)	-	HuBERT, WavLM	DiT	Vocos (Causal)	CM	Iter. NAR	Sup. Scratch		Conv	VC
ReHear (Liu et al., 2026c)	3B	-	Whisper-large-v3	-	-	AR	-	SFT	Und	ASR
AuroLA (Xu et al., 2026)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	SFT	Und	ATR
TAC (Kumar et al., 2026b)	-	-	Qwen2-Audio	-	-	AR	-	SFT	Und	AAC, Cls, AQA
AIDD (Dror et al., 2026)	90M	WavTokenizer	DiT	WavTokenizer	CM	Iter. NAR	Sup. Scratch		Conv	AI
GSRM (Shen et al., 2026)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	SFT	Und	AQA
Echo (Wu et al., 2026a)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	SFT, RL	Und	AQA
LLM-ForcedAligner (Mu et al., 2026)	0.9B	AuT	Qwen3-0.6B	-	RTD	One-shot NAR	-	SFT	Und	ASR
EmotionThinker (Wang et al., 2026c)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	SFT, RL	Und	Cls
TARS (Wang et al., 2026a)	7B	-	Qwen2.5-Omni-7B / Phi-4-MM-7B	-	-	AR	-	SFT, RL	Und	AQA, ASR
TimeAudio (Wang et al., 2026d)	7B	BEATs, Whisper Encoder	SALMONN-7B	-	GA	AR	-	SFT	Und	AAC, AQA
CESAR (Fan et al., 2026)	7B	-	Qwen2.5-Omni-7B	-	-	AR	-	RL	Und	AQA
USR 2.0 (Haliassos et al., 2026)	86M/171M/503M/953M	ResNet-18	USR	-	ESA	Disc. + AR	-	SFT	Und	ASR
AudSemThinker (Wijngaard et al., 2026)	7B	-	Qwen2.5-Omni	-	-	AR	-	SFT, RL	Und	AQA, AAC
STITCH (Chiang et al., 2026)	9B	-	GLM-4-Voice-9B	-	-	AR	-	SFT	Int	AQA, SDS
Mellow (Deshmukh et al., 2026)	167M	HTSAT	SmolLM2	-	RTD	AR	-	SFT	Und	AQA, AAC
Unified TSD (Gupta et al., 2026)	-	ConvNeXT	BiGRU	-	CM	Disc.	-	SFT	Und	Cls
SLM-SS (Li et al., 2026b)	~600M	Encodec, WavLM-large	Transformer	Encodec Decoder	CM + TI	AR + Iter. NAR	Sup. Scratch		Conv	SS
CoT-ASR (Deng et al., 2026)	3.8B	Conformer	Phi-4-mini-instruct	-	ESA	AR	-	SFT	Und	ASR
MeanVC (Ma et al., 2026a)	14M	ECAPA-TDNN, Timbre Encoder, Fast-U2++	DiT	Vocos	CM + GA	AR + One-shot NAR	Sup. Scratch		Conv	VC
PersonaPlex (Roy et al., 2026)	7B	-	Moshi	-	-	AR	-	SFT	Int	FSD, SDS, S2ST, ZVC

Cond. Method: **CM**: Condition Modulation, **TI**: Token-level Integration, **RTD**: Rigid Temporal Downsampling, **GA**: Global Aggregation, **ESA**: Elastic Semantic Alignment.
Modeling Paradigm: **AR**: Autoregressive, **Iter. NAR**: Iterative Non-Autoregressive, **One-shot NAR**: One-shot Non-Autoregressive, **Disc.**: Discriminative.
Pre-training: **Sup. Scratch**: Supervised training from scratch, **Tr. Free**: Training-free.
Tasks: **Und**: Understanding, **Gen**: Generation, **Conv**: Conversion, **Int**: Interaction.
Understanding: **Cls**: Classification, **ASR**: Auto Speech Recognition, **S2TT**: Speech-to-Text Translation, **AQA**: Audio Question Answering, **AAC**: Automated Audio Captioning, **ATR**: Audio Text Retrieval, **SD**: Speaker Diarization.
Generation: **TTS**: Text to Speech, **ZVC**: Zero-shot Voice Cloning, **TTA**: Text to Audio, **VTA**: Video to Audio, **TTM**: Text to Music, **T2ST**: Text-to-Speech Translation.
Conversion: **VC**: Voice Conversion, **SE**: Speech Enhancement, **SS**: Speech Separation, **AI**: Audio Inpainting, **S2ST**: Speech-to-Speech Translation.
Interaction: **FSD**: Full-duplex Spoken Dialogue, **SDS**: Spoken Dialogue System.

Table 7: Overview of Downstream Task Fine-tuning Models

Author Index

Ahmad, Raman, 84
Arigala, Adarsh, 47

Baumann, Timo, 84
Bhargava, Neha, 19

Chen, Meiyi, 84

Dogan, Mahîr, 84

Gangwar, Arjun, 47
Gayed, John Maurice, 1
Genabith, Josef Van, 84
Geoffre, Thierry, 65

He, Liang, 100
Huang, Keran, 33
Huang, Yixuan, 1, 65

Labib, Adnan, 1, 65
Lam, Wai, 94
Lu, Hongyuan, 94

Miao, Shujie, 94
Mă, Quánzī, 84

Nagashima, Tomohiro, 84

Punglia, Shaanil, 19

Rathore, Vaibhav, 19

Schuler, Christian, 84
Sureka, Siddharth, 19

Telwadkar, Dadhichi, 19

Umesh, Srinivasan, 47

Wang, Kaihong, 100
Wang, Qiao, 1, 65
Wu, Jiahui, 1

Xie, Xiaochuan, 33

Yuan, Zheng, 1, 65

Zaghouani, Wajdi, 56
Zhang, Lexuan, 77
Zhang, Xueqing, 84