Mu2SLAM: Multitask, Multilingual Speech and Language Models

Summary

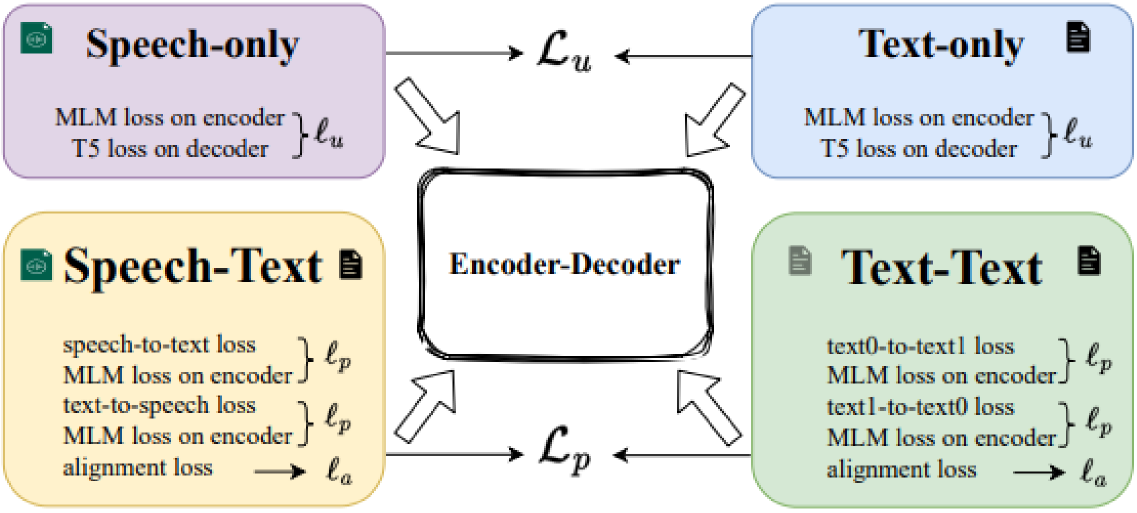
Mu2SLAM is a powerful multilingual sequence-to-sequence model designed to handle a wide range of speech and text tasks across more than 100 languages. It is pre-trained jointly on unlabelled speech, unlabelled text, and supervised data covering Automatic Speech Recognition (ASR), Automatic Speech Translation (AST), and Machine Translation (MT).

By unifying the training of speech and text through a shared encoder-decoder architecture, Mu2SLAM applies a masked language modeling (MLM) objective on the encoder and a T5-style denoising loss on the decoder. It introduces minimal modality-specific components—only a CNN block for processing speech—while aligning speech and text representations through supervised losses.

Mu2SLAM sets a new state-of-the-art on the CoVoST AST benchmark (+1.9 BLEU on xx→en, +1.1 BLEU on en→xx), and achieves ASR performance competitive with larger or more specialized models, despite using a lighter Transformer decoder. It also significantly outperforms previous speech-text models like mSLAM on multilingual NLP tasks (e.g., +6% on XNLI), narrowing the performance gap with strong text-only models like mT5.

The model also proposes innovations like **gradual fine-tuning** and **noisy fine-tuning** to bridge the gap between pre-training and downstream tasks. Overall, Mu2SLAM demonstrates a strong step toward building a single unified model for understanding and generating both speech and text across many languages and tasks.

Architecture



Strengths of Paper

Background

Recent NLP models are moving toward unified architectures for both understanding and generation across languages and modalities, using encoder-only (e.g., BERT, wav2vec 2.0), decoder-only (e.g., GPT, AudioLM), and encoder-decoder (e.g., T5, SpeechT5) designs. However, current speech-text models are limited—they treat text as auxiliary input, aren't evaluated on text-only benchmarks, lack multilingual joint modeling, underuse multi-task learning, and rely on modality-specific networks and loss functions, reducing generalization.

Mu2SLAM

Mu2SLAM, a multitask, multilingual, encoder-decoder model that is pre-trained jointly on a diverse set of tasks: unlabelled speech, unlabelled text, labelled speech-text (ASR and AST), and labelled text-text (machine translation). To maintain consistency and scalability in pre-training, Mu2SLAM uses a unified loss function composed of three components: a masked language modelling (MLM) loss on the encoder (like BERT), a T5-style generation loss on the decoder (reconstructing the target sequence), and an alignment loss applied only to labelled data for better cross-modal mapping. The model architecture minimizes modality-specific layers, with the only dedicated speech component being a conventional convolutional block used to extract speech representations.

Training Strategies

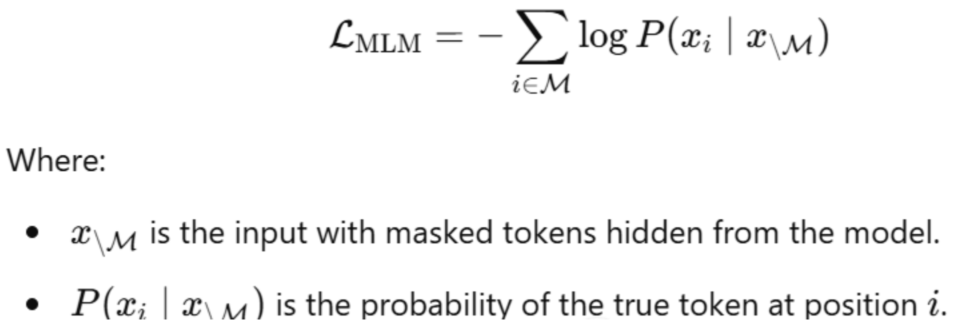
- **Gradual fine-tuning**:
  - Continue training on labeled datasets before task-specific fine-tuning
- **Noisy fine-tuning**:
  - **Perturb decoder inputs** (from Cheng et al., 2019)
  - **Speech input augmentation** (SpecAugment, Park et al., 2019)
  - Improves **robustness and generalization**

Evaluation

- **CoVoST AST Benchmark**:
  - State-of-the-art results:
    - +1.9 BLEU (xx→en)
    - +1.1 BLEU (en→xx)
- **VoxPopuli ASR**:
  - Matches **mSLAM** with RNN-T decoder using a **Transformer decoder**
- **XTREME Benchmark**:
  - +6% over **mSLAM** on XNLI
  - Competitive with **mT5 and TydiQA** (strong unimodal text models)

Loss Function

- **Masked Language Modeling (MLM) Loss**

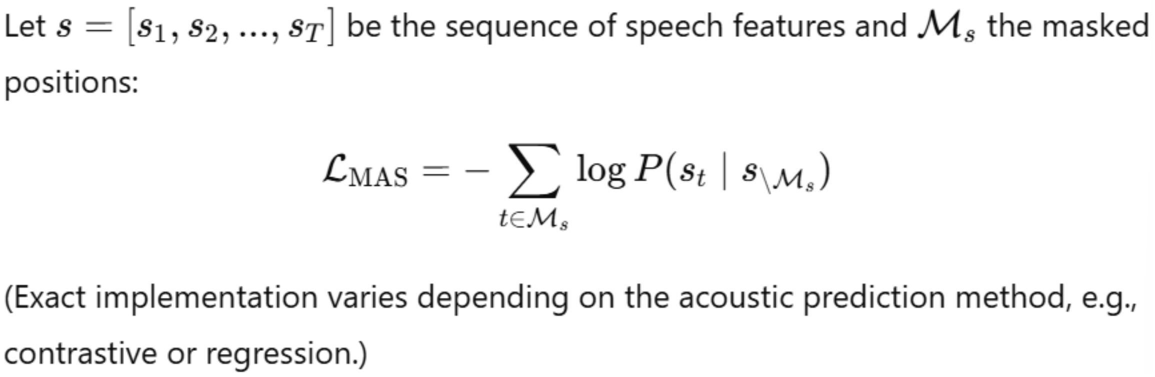


$$\mathcal{L}_{\text{MLM}} = -\sum_{i \in \mathcal{M}} \log P(x_i \mid x_{\setminus \mathcal{M}})$$

Where:

- $x_{\setminus \mathcal{M}}$ is the input with masked tokens hidden from the model.
- $P(x_i \mid x_{\setminus \mathcal{M}})$ is the probability of the true token at position i.

- **Masked Acoustic Modeling Loss (Self-Supervised Speech)**

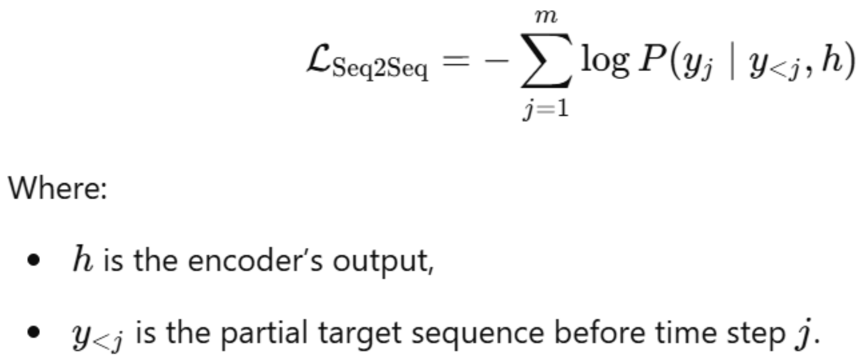


Let $s = [s_1, s_2, \ldots, s_T]$ be the sequence of speech features and $\mathcal{M}_s$ the masked positions:

$$\mathcal{L}_{\text{MAS}} = -\sum_{t \in \mathcal{M}_s} \log P(s_t \mid s_{\setminus \mathcal{M}_s})$$

(Exact implementation varies depending on the acoustic prediction method, e.g., contrastive or regression.)

- **Sequence-to-Sequence (T5-style) Loss**



$$\mathcal{L}_{\text{Seq2Seq}} = -\sum_{j=1}^{m} \log P(y_j \mid y_{<j}, h)$$

Where:

- h is the encoder's output,
- $y_{<j}$ is the partial target sequence before time step j.

- **Alignment Loss (Cross-modal Supervised Learning)**

$$\mathcal{L}_{\text{Align}} = -\log \frac{\exp(\text{sim}(h_s, h_t)/\tau)}{\sum_k \exp(\text{sim}(h_s, h_k)/\tau)}$$

Where:

- h_s: speech encoder output
- h_t: text encoder output
- sim: similarity function (e.g., cosine similarity)
- τ: temperature parameter

(Note: Exact formulation might vary in implementation.)

- **Final Combined Loos**

$$\begin{aligned}
\mathcal{L}_p(\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{s},\mathbf{t})\in D_{st}} \Big\{ & \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{S}^{\mathbf{m}}, \mathbf{t}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \\
& \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{t}^{\mathbf{m}}, \mathbf{z}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_a([\mathbf{S}, \mathbf{t}]^{\mathbf{m}}, [\mathbf{z}, \mathbf{t}]^{\neg\mathbf{m}})] \Big\} \\
& + \mathbb{E}_{(\mathbf{v},\mathbf{t})\in D_{tt}} \Big\{ \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{v}^{\mathbf{m}}, \mathbf{t}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \\
& \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{t}^{\mathbf{m}}, \mathbf{v}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_a([\mathbf{v}, \mathbf{t}]^{\mathbf{m}}, [\mathbf{v}, \mathbf{t}]^{\neg\mathbf{m}})] \Big\}
\end{aligned}$$

Fine Tunning

- **Direct Fine-Tuning**
  This method continues training the pre-trained model on labeled data from a specific downstream task, using a smaller learning rate. It is mainly used for tasks not included during pre-training, such as text classification, allowing the model to adapt without altering its architecture.
- **Gradual Fine-Tuning**
  To reduce mismatch from pre-training with [MASK] tokens, gradual fine-tuning is done in two stages. First, the model is fine-tuned on labeled sequence-to-sequence tasks (like ASR, AST, MT) without masking, using standard sequence loss across multiple languages and tasks—termed multi-task multilingual fine-tuning. Second, it is further fine-tuned on a single target task for specialization.

- **Noisy Fine-Tuning**
  To improve robustness in speech-text tasks, this method applies augmentation to speech inputs (e.g., SpecAugment) and adds noise to decoder inputs by replacing some tokens with synonyms using word embeddings. This helps reduce overfitting and prevents error accumulation during decoding.

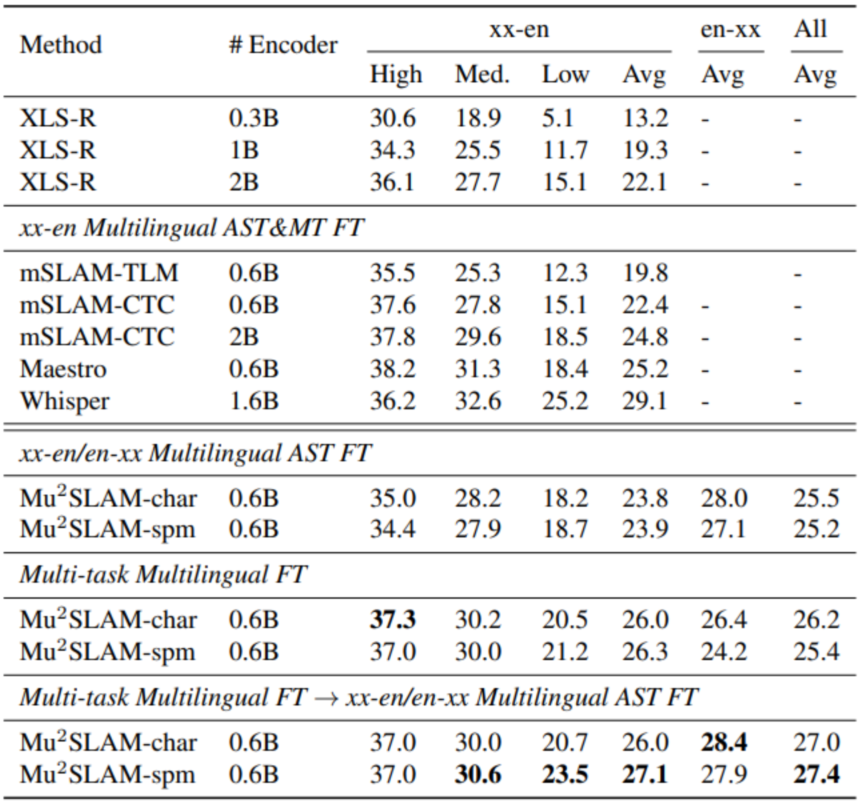
Experiments

- **Speech translation results on the CoVoST 2 dataset**
  The table summarizes BLEU score improvements on the CoVoST 2 dataset when fine-tuning pre-trained models for both English-to-non-English (en-xx) and non-English-to-English (xx-en) tasks. Three strategies are compared:

  Direct Multilingual Fine-tuning: Focusing solely on AST data already outperforms baseline models (like XLS-R and 0.6B mSLAM) by up to 1.5 BLEU points.

  Multi-task Multilingual Fine-tuning: Involving all available language pairs from AST, ASR, and MT can boost xx-en results, though it tends to lower en-xx performance. This suggests a trade-off when the model is adapted for multiple tasks.

  Gradual Fine-tuning: This two-step process (starting with a broader training phase, then further specializing on en-xx or xx-en) achieves the best outcomes, notably setting a new state-of-the-art with a +1.9 BLEU gain over previous work (Maestro).



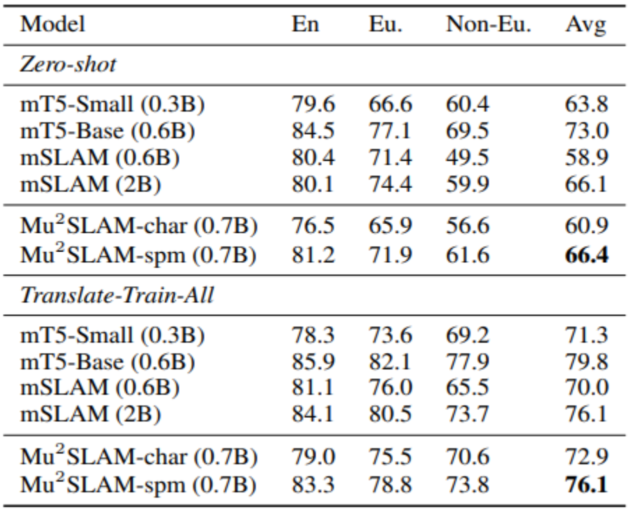
Method	# Encoder	xx-en				en-xx	All
		High	Med.	Low	Avg	Avg	Avg
XLS-R	0.3B	30.6	18.9	5.1	13.2	-	-
XLS-R	1B	34.3	25.5	11.7	19.3	-	-
XLS-R	2B	36.1	27.7	15.1	22.1	-	-
xx-en Multilingual AST&MT FT							
mSLAM-TLM	0.6B	35.5	25.3	12.3	19.8		-
mSLAM-CTC	0.6B	37.6	27.8	15.1	22.4	-	-
mSLAM-CTC	2B	37.8	29.6	18.5	24.8	-	-
Maestro	0.6B	38.2	31.3	18.4	25.2	-	-
Whisper	1.6B	36.2	32.6	25.2	29.1	-	-
xx-en/en-xx Multilingual AST FT							
Mu^{2}SLAM-char	0.6B	35.0	28.2	18.2	23.8	28.0	25.5
Mu^{2}SLAM-spm	0.6B	34.4	27.9	18.7	23.9	27.1	25.2
Multi-task Multilingual FT							
Mu^{2}SLAM-char	0.6B	**37.3**	30.2	20.5	26.0	26.4	26.2
Mu^{2}SLAM-spm	0.6B	37.0	30.0	21.2	26.3	24.2	25.4
Multi-task Multilingual FT → xx-en/en-xx Multilingual AST FT							
Mu^{2}SLAM-char	0.6B	37.0	30.0	20.7	26.0	**28.4**	27.0
Mu^{2}SLAM-spm	0.6B	37.0	**30.6**	**23.5**	**27.1**	27.9	**27.4**

- **Multilingual Speech Recognition**

  ASR-only multilingual fine-tuning: This setup uses only ASR data.

  Multi-task multilingual fine-tuning: This approach jointly fine-tunes on ASR, AST, and other tasks. However, because AST data dominates and disrupts the encoder–decoder alignment, this method yields unreasonable numbers, as ASR tasks are sensitive to heterogeneous data.

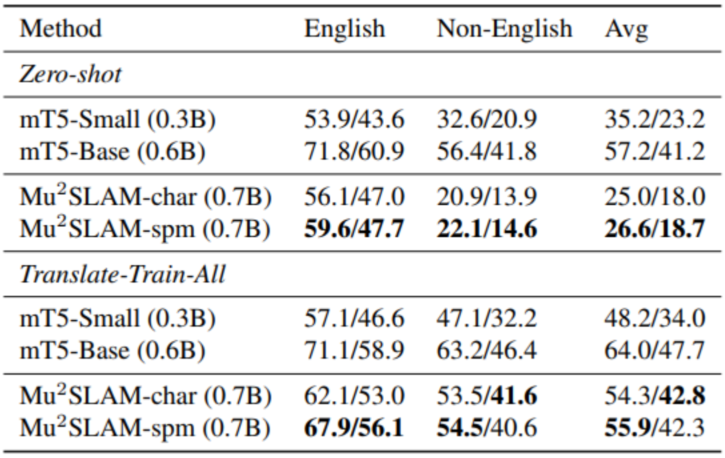
  Gradual fine-tuning: This strategy transitions from multi-task multilingual fine-tuning to ASR-only fine-tuning. Here, Mu2SLAM-spm outperforms Mu2SLAM-char, gaining roughly 0.5 WER improvement. Initially, Mu2SLAM-spm only beat XLS-R, but after multi-task fine-tuning, it reaches performance like mSLAM—though it still lags behind Maestro.



Model	En	Eu.	Non-Eu.	Avg
Zero-shot				
mT5-Small (0.3B)	79.6	66.6	60.4	63.8
mT5-Base (0.6B)	84.5	77.1	69.5	73.0
mSLAM (0.6B)	80.4	71.4	49.5	58.9
mSLAM (2B)	80.1	74.4	59.9	66.1
Mu^{2}SLAM-char (0.7B)	76.5	65.9	56.6	60.9
Mu^{2}SLAM-spm (0.7B)	81.2	71.9	61.6	**66.4**
Translate-Train-All				
mT5-Small (0.3B)	78.3	73.6	69.2	71.3
mT5-Base (0.6B)	85.9	82.1	77.9	79.8
mSLAM (0.6B)	81.1	76.0	65.5	70.0
mSLAM (2B)	84.1	80.5	73.7	76.1
Mu^{2}SLAM-char (0.7B)	79.0	75.5	70.6	72.9
Mu^{2}SLAM-spm (0.7B)	83.3	78.8	73.8	**76.1**

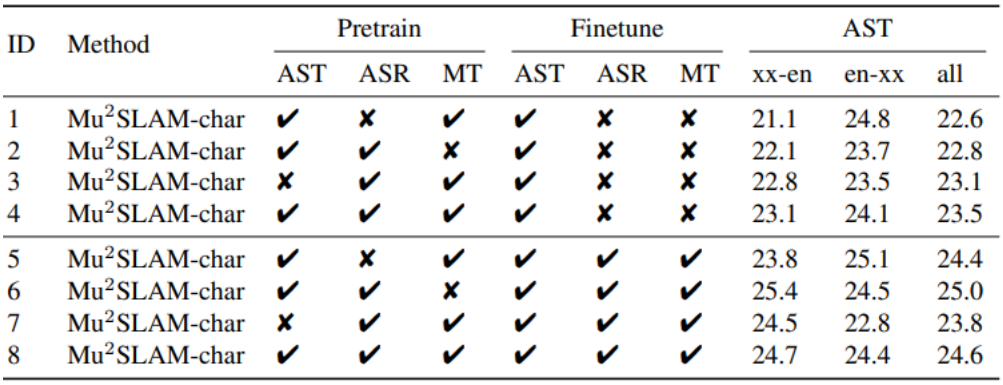
Effect of Paired Data:

- **Experiment Setup:** Different combinations of AST, ASR, and MT data were used during both pre-training and fine-tuning.
- **AST-Only Fine-Tuning :**
    - The best performance was achieved in Row 4 when all available speech–text (AST) data was incorporated during fine-tuning.
    - Notably, removing ASR data led to better performance for en-xx directions.
    - Reasoning: Translation data (AST and MT) help learn a good encoder–decoder alignment, whereas ASR data—with its strong monotonic alignment—can actually hurt alignment, especially when there isn't much non-English translation data for en-xx pairs.
- **Multi-task Multilingual Fine-Tuning:**
    - When using all pre-training data, the model did not perform best.
    - The best AST results came from a setup (Row 6) that omitted MT data during pre-training, likely because text data can crowd the encoder's capacity, though its effect is compensated during fine-tuning.
    - Conclusion: Multitask pre-training is beneficial for general speech–text representations when downstream tasks are not predetermined. However, when evaluating on both speech and text tasks, incorporating all available speech and text data remains advantageous.



Method	English	Non-English	Avg
Zero-shot			
mT5-Small (0.3B)	53.9/43.6	32.6/20.9	35.2/23.2
mT5-Base (0.6B)	71.8/60.9	56.4/41.8	57.2/41.2
Mu^{2}SLAM-char (0.7B)	56.1/47.0	20.9/13.9	25.0/18.0
Mu^{2}SLAM-spm (0.7B)	**59.6/47.7**	**22.1/14.6**	**26.6/18.7**
Translate-Train-All			
mT5-Small (0.3B)	57.1/46.6	47.1/32.2	48.2/34.0
mT5-Base (0.6B)	71.1/58.9	63.2/46.4	64.0/47.7
Mu^{2}SLAM-char (0.7B)	62.1/53.0	53.5/**41.6**	54.3/**42.8**
Mu^{2}SLAM-spm (0.7B)	**67.9/56.1**	**54.5**/40.6	**55.9**/42.3

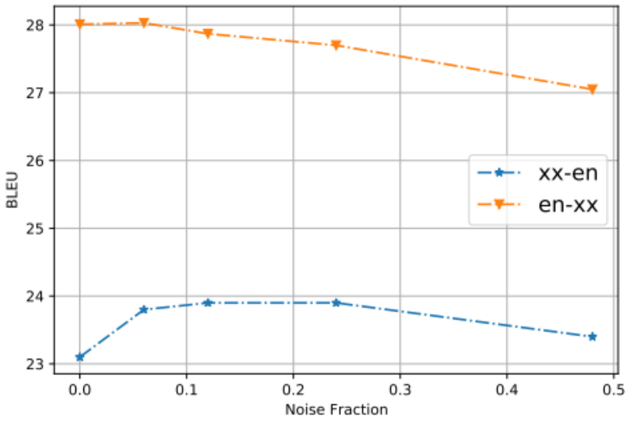
Table 5. TyDiQA-GoldP results (F1/EM) on the test sets.



ID	Method	Pretrain			Finetune			AST		
		AST	ASR	MT	AST	ASR	MT	xx-en	en-xx	all
1	Mu^{2}SLAM-char	✔	✘	✔	✔	✘	✘	21.1	24.8	22.6
2	Mu^{2}SLAM-char	✔	✔	✘	✔	✘	✘	22.1	23.7	22.8
3	Mu^{2}SLAM-char	✘	✔	✔	✔	✘	✘	22.8	23.5	23.1
4	Mu^{2}SLAM-char	✔	✔	✔	✔	✘	✘	23.1	24.1	23.5
5	Mu^{2}SLAM-char	✔	✘	✔	✔	✔	✔	23.8	25.1	24.4
6	Mu^{2}SLAM-char	✔	✔	✘	✔	✔	✔	25.4	24.5	25.0
7	Mu^{2}SLAM-char	✘	✔	✔	✔	✔	✔	24.5	22.8	23.8
8	Mu^{2}SLAM-char	✔	✔	✔	✔	✔	✔	24.7	24.4	24.6

Effect of Noisy Fine-Tuning:

The method involves randomly replacing decoder inputs with synonym tokens during fine-tuning to introduce noise. For xx-en, increasing noise from 0 to 0.06 markedly improves BLEU scores, then the performance plateaus and drops significantly at 0.48. In contrast, for en-xx, a noise ratio of 0.06 yields only modest gains, and further increases severely hurt performance. This suggests that noise favors decoder performance for similar language pairs (xx-en) over diverse ones (en-xx), pointing to word embeddings' limitations in capturing cross-language similarities, as noted by Cheng et al. (2022).



Weakness

It does not evaluate on speech generation tasks despite pre-training on a text-to-speech objective, meaning its capability in generating speech remains unverified.
• The model is confined to only about 100 languages from academic datasets, limiting its scalability compared to systems trained on larger, proprietary datasets.
• It does not incorporate speech-to-speech tasks into its pre-training framework, which could further broaden its capabilities in handling diverse speech modalities.
• Its zero-shot performance on speech translation, recognition, and text generation is limited, indicating challenges in transferring alignment to unseen language pairs or modalities.

Suggestions to Authors (as a Reviewer

- Include evaluations on speech generation and speech-to-speech tasks to validate the model's full multi-modal potential.
- Extend language coverage beyond the 100 academic dataset languages to improve scalability and real-world applicability.
- ddress the weak zero-shot performance by exploring alternate architectures (e.g., decoder-only) or enhanced alignment strategies.
- Integrate speech-to-speech pre-training objectives for more comprehensive multi-modal learning.
- Improve tokenization strategies to better handle non-English and low-resource languages.
- Conduct deeper analysis on why performance deteriorates in certain tasks and directions (e.g., en-xx).
- Strengthen comparisons with stronger text-only baselines to better highlight Mu2SLAM's advantages.

Rating: 7/10
Mu2SLAM presents a strong unified framework with competitive results, but the lack of coverage in key modalities and limited generalization capacity restricts its impact.