Mu^{2}SLAM: Multitask, Multilingual Speech and Language Models

Yong Cheng [1] **Yu Zhang** [1] **Melvin Johnson** [1] **Wolfgang Macherey** [1] **Ankur Bapna** [1]

Abstract

We present Mu^{2}SLAM, a multilingual sequence-to-sequence model pre-trained jointly on unlabeled speech, unlabeled text and supervised data spanning Automatic Speech Recognition (ASR), Automatic Speech Translation (AST) and Machine Translation (MT), in over 100 languages. By leveraging a quantized representation of speech as a target, Mu^{2}SLAM trains the speech-text models with a sequence-to-sequence masked denoising objective similar to T5 on the decoder and a masked language modeling objective (MLM) on the encoder, for both unlabeled speech and text, while utilizing the supervised tasks to improve cross-lingual and cross-modal representation alignment within the model. On CoVoST AST, Mu^{2}SLAM establishes a new state-of-the-art for models trained on public datasets, improving on xx-en translation over the previous best by 1.9 BLEU points and on en-xx translation by 1.1 BLEU points. On Voxpopuli ASR, our model matches the performance of an mSLAM model fine-tuned with an RNN-T decoder, despite using a relatively weaker Transformer decoder. On text understanding tasks, our model improves by more than 6% over mSLAM on XNLI, getting closer to the performance of mT5 models of comparable capacity on XNLI and TydiQA, paving the way towards a single model for all speech and text understanding tasks.

1. Introduction

The recent rapid developments in NLP have witnessed the tremendous success of moving towards unified text models for both understanding and generation tasks across hundreds of languages, evolving into numerous pre-trained models from encoder-only models focusing on text understanding (Devlin et al., 2019; Devlin, 2018), to decoder-only models (Radford et al., 2018; Chowdhery et al., 2022) and encoder-decoder models (Song et al., 2019; Lewis et al., 2019; Raffel et al., 2020; Xue et al., 2020) for both understanding and generation. The speech pre-training methods have shown a similar trend towards unified models from the dominant encoder-only models (Baevski et al., 2020; Hsu et al., 2021; Babu et al., 2021; Bapna et al., 2021; 2022) to generative models on cross-modal speech and text data, exemplified by a couple of recent trails such as decoder-only models (Borsos et al., 2022) and encoder-decoder models (Ao et al., 2021; Chen et al., 2022; Sainath et al., 2022; Zhou et al., 2022; Zhang et al., 2022b; Tang et al., 2022).

Although these works have achieved impressive performance, they only consider partial aspects of the unified models in speech and text. First, except for SLAM and mSLAM (Bapna et al., 2021; 2022), most of them merely focus on speech-related tasks by taking text data as auxiliary inputs while ignoring evaluations on text-related benchmarks, which leaves us unknown to gauge the effect of interference and capacity dilution. Second, there are few studies investigating multilingual modeling with both speech and text (Bapna et al., 2022; Chen et al., 2022), which limits them in leveraging cross-lingual transfer to enrich the speech and text joint representations. Third, multi-task learning has demonstrated the effectiveness of inductive transfer to improve model generalization, yet it is understudied in speech-text pre-training (Tang et al., 2022; Zhang et al., 2022b; Chen et al., 2022) where they explicitly differentiate the utilization of labeled data in pre-training by introducing customized networks and losses. Fourth, it is essential for prior speech-text models to design modality-specific blocks and losses to yield high performance (Bapna et al., 2022; Chen et al., 2022; Tang et al., 2022; Zhang et al., 2022b;a) which somewhat violates the principle of the unified models by using one model for all tasks, thus undermining the language and modality transfer to learn general speech-text shared representations.

In this work, we propose a multi-task multilingual pre-training method based on an encoder-decoder model, called Mu^{2}SLAM. The speech-text model is jointly pre-trained on a set of different tasks involving unlabeled speech, unlabeled text, labeled speech-text (ASR&AST), and labeled text-text (MT). We scale up the language type in both

[1]Google Research, Google LLC, USA. Correspondence to: Yong Cheng <chengyong@google.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

speech and text to more than 100, covering the majority of mainstream spoken languages. For the simplicity of extending our current pre-training to more data, we unify the pre-training losses for unlabeled and labeled data by defining a masked language modeling (MLM) loss on the encoder (Devlin et al., 2019), a similar T5 loss on decoder (Song et al., 2019; Raffel et al., 2020) and an alignment loss only for the labeled data. To enforce the sharing and take full advantage of modality capacity for speech and text, we minimize the number of modality-specific layers in our model design with only a conventional CNN block used to extract speech representations, which pushes forward speech-text models towards the unified models. As our pre-training method inherits the idea of BERT (Devlin et al., 2019) to reconstruct the masked tokens according to the contextual unmasked tokens, the artificial token [MASK] used in pre-training is absent from labeled data in fine-tuning (Yang et al., 2019). The discrepancy between pre-training and fine-tuning hinders the model from being adequately optimized on the downstream applications. To alleviate this issue, we propose a gradual fine-tuning by continuing training the models on the set of labeled sets then turning to a specific task. To further boost the model performances on speech-text tasks during fine-tuning, we propose a noisy fine-tuning by perturbing the decoder inputs (Cheng et al., 2019) in addition to the speech augmentation in the encoder (Park et al., 2019).

Extensive experimental results on the multilingual CoVoST AST (Wang et al., 2021b), Voxpopuli ASR (Wang et al., 2021a) and XTREME benchmarks show that our joint speech-text pre-trained models can achieve competitive results on both speech and text tasks. More specifically, Mu^{2}SLAM establishes a new SOTA for models trained on public datasets on CoVoST, with up to 1.9 BLEU points on xx-en and 1.1 BLEU points on en-xx against the previous best results. On Voxpopuli ASR, our model based on a Transformer decoder matches the performance of an mSLAM model fine-tuned with an RNN-T decoder although the RNN-T decoder is more favorable to ASR tasks. On the multilingual text XTREME, Mu^{2}SLAM outperforms mSLAM with 6% on XNLI, getting closer to the performance of mT5 models of comparable capacity on XNLI and TydiQA. In analyses, we conduct ablation studies to gain further insight into which combination set of supervised datasets in our approach matters the most during the pre-training and fine-tuning. We also vary the noise ratio to investigate the effect of noisy fine-tuning for different speech translation directions.

These results demonstrate that Mu^{2}SLAM is the first truly multi-modal speech and text model which is capable of performing a wide variety of understanding and generation tasks for speech and text, attaining competitive results with uni-modal text models and vastly improving over speech-only models.

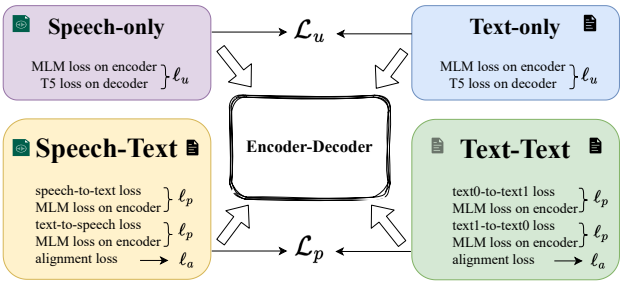

Figure 1. An overview of Mu^{2}SLAM. A ℓ_u loss is used to train speech-only and text-only data by computing masked language modeling (MLM) loss on the encoder and a similar T5 loss on the decoder. The supervised speech-text and text-text data also share the pre-training loss including forward and backward ℓ_p and an alignment loss ℓ_a between different languages or modalities. ℓ_p consists of a translation loss from input to target and a MLM loss on the encoder. Our speech-text models are pre-trained with $\mathcal{L}_u$ on unlabeled data and $\mathcal{L}_p$ on labeled data. In practice, we incorporate an additional CTC loss for ASR.

2. Approach

We propose a multi-task multilingual pre-training method, Mu^{2}SLAM, for speech and text, aiming to pre-train speech-text models on arbitrary tasks related to speech and/or text. The speech and text data can be cast into two types of data, unlabeled data without supervised labels and labeled data usually accompanied with human-annotated labels. As Figure 1 shows, we consider four types of data, *i.e.*, speech-only, text-only, speech-text and text-text. The main idea is to unify these training examples into the sequence-to-sequence format and apply similar optimization objectives on the encoder and decoder. The losses on unlabeled data ($\mathcal{L}_u$) and labeled data ($\mathcal{L}_p$) are combined to pre-train the speech-text models.

2.1. Model Architecture

Mu^{2}SLAM is based on an encoder-decoder backbone model. For speech inputs, we follow mSLAM (Bapna et al., 2022) to convert an acoustic feature sequence of 80-dimensional log Mel spectrograms into a sequence of latent speech representations via a CNN block. The CNN block consisting of two 2D-convolutional layers with strides (2, 2) also acts as a sub-sampling mechanism with a 4x reduction in the sequence length dimension. A subsequent linear projection layer is used to map the dimension of the latent speech representations to that of the encoder stack, we denote the speech representations as $\mathbf{S}$. The text input $\mathbf{t}$ simply goes through a token embedding layer to be transformed as a sequence of embeddings. To specify the language and modality, we add language and modality embeddings to word embeddings or speech representations $\mathbf{S}$ in addition to the conventional positional embeddings. The speech and text representations are

then fed into a shared multi-modal encoder-decoder model. We prefer a deep encoder with 24 Conformer layers (Gulati et al., 2020) (a similar encoder as mSLAM) and a shallow decoder with 6 Transformer layers (Vaswani et al., 2017), which favors faster inference while maintaining competitive quality (Kasai et al., 2020).

2.2. Speech Tokenization

The basis of the proposed speech-text pre-training approach is to treat the speech inputs as an additional language, which requires a speech tokenizer to quantize the continuous speech representations $\mathbf{S} = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$ into discrete ids $\mathbf{z} = (z_1, z_2, ..., z_N)$. To this end, each speech representation vector $\mathbf{s}$ is independently projected into a discrete id z by finding its nearest neighbour in the speech codebook $\mathcal{G}$.

$$z = \underset{i}{\operatorname{argmin}} \|\mathcal{G}_i - \mathbf{s}\|. \tag{1}$$

In mSLAM, the parameters of the speech tokenizer are learned from scratch by a contrastive loss (Baevski et al., 2020) over a speech-only encoder. For simplicity, we directly utilize the pretrained speech tokenizer in mSLAM and keep it constant during our model training.

2.3. Pre-training Objectives

In this paper, we have four different training sets related to speech and/or text: a speech-only set D_s, a text-only set D_t, a speech-text set D_{st} and a text-text set D_{tt}. We want to unify the pre-training losses for unlabeled data and labeled data, which make our pre-training methods easily extensible to more datasets.

Losses on unlabeled data Given an unlabeled training example $\mathbf{x} = (x_1, x_2, ..., x_N)$, we first use it as a source-target pair $(\mathbf{x}, \mathbf{x})$ for the sequence-to-sequence model training. Then we randomly construct a $0/1$ masking vector $\mathbf{m}$ sampled from a prior distribution. We apply the masking vector $\mathbf{m}$ to the source $\mathbf{x}$ by replacing the token x_i with a [MASK] token if $m_i = 1$. The corrupted source $\mathbf{x}$ is denoted as $\mathbf{x}^{\mathbf{m}}$. For the target $\mathbf{x}$, we employ the complementary masking operation $\neg\mathbf{m}$ by setting x_i to the [MASK] token if $m_i = 0$ and denote it as $\mathbf{x}^{\neg\mathbf{m}}$. Finally, to enable the model to predict the masked source tokens on both the encoder and the decoder, the loss $\ell_u(\mathbf{x}^{\mathbf{m}}, \mathbf{x}^{\neg\mathbf{m}}; \boldsymbol{\theta})$ on the pseudo pair data $(\mathbf{x}^{\mathbf{m}}, \mathbf{x}^{\neg\mathbf{m}})$ is computed as:

$$\sum_{m_i=1} \log P(x_i|\mathbf{x}^{\mathbf{m}}; \boldsymbol{\theta_e}) + \sum_{m_i=1} \log P(x_i|\mathbf{x}^{\neg\mathbf{m}}_{<i}, \mathbf{x}^{\mathbf{m}}; \boldsymbol{\theta}), \tag{2}$$

where the parameter set $\boldsymbol{\theta} = \{\boldsymbol{\theta_e}, \boldsymbol{\theta_d}\}$ is split into two parts, *i.e.*, $\boldsymbol{\theta_e}$ for the encoder and $\boldsymbol{\theta_d}$ for the decoder.

It is natural to use the above loss to pre-train the model on the text-only data set D_t. For speech, we take the representations $\mathbf{S}$ extracted by the CNN block as inputs and their corresponding discrete ids $\mathbf{z}$ quantized by the speech tokenizer as targets. Likewise, we can also mask out the speech representations by substituting the embedding of the [MASK] token for $\mathbf{S}_i$ if $m_i = 1$ and applying $\neg\mathbf{m}$ to $\mathbf{z}$, because they have identical lengths. The loss on the unlabeled speech-only (D_s) and text-only (D_t) sets is:

$$\begin{aligned}\mathcal{L}_u(\boldsymbol{\theta}) =& \mathbb{E}_{\mathbf{s}\in D_s}\mathbb{E}_{\mathbf{m}}[\ell_u(\mathbf{S}^{\mathbf{m}}, \mathbf{z}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \\ & \mathbb{E}_{\mathbf{t}\in D_t}\mathbb{E}_{\mathbf{m}}[\ell_u(\mathbf{t}^{\mathbf{m}}, \mathbf{t}^{\neg\mathbf{m}}; \boldsymbol{\theta})],\end{aligned} \tag{3}$$

where we allow the sequence of speech representations $\mathbf{S}$ to be passed into ℓ_u for convenience.

Losses on labeled data For a labeled example $(\mathbf{x}, \mathbf{y})$ where $\mathbf{x} = (x_1, x_2, ..., x_N)$ and $\mathbf{y} = (y_1, y_2, ..., y_M)$, we employ its forward and backward sequence-to-sequence loss, *i.e.*, $P(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta})$ and $P(\mathbf{x}|\mathbf{y}; \boldsymbol{\theta})$. Since the labeled training data is usually not abundant compared to the unlabeled data, we introduce a similar mask operation $\mathbf{m}$ for the source part of the labeled data to avoid overfitting. Meanwhile, the reconstruction loss in the encoder is also applied to enhance the representation learning for the deeper encoder. Thus the forward sequence-to-sequence loss $\ell_p(\mathbf{x}^{\mathbf{m}}, \mathbf{y}; \boldsymbol{\theta})$ on the paired data $(\mathbf{x}^{\mathbf{m}}, \mathbf{y})$ is calculated as:

$$\sum_{m_i=1} \log P(x_i|\mathbf{x}^{\mathbf{m}}; \boldsymbol{\theta_e}) + \sum_i \log P(y_i|y_{<i}, \mathbf{x}^{\mathbf{m}}; \boldsymbol{\theta}). \tag{4}$$

To better align learned representations between different languages and modalities, except for the fully shared encoder-decoder model to implicitly encourage the alignment, we also introduce an explicit alignment loss on the encoder and decoder. Given the paired data $(\mathbf{x}, \mathbf{y})$, they are concatenated into a new sentence $[\mathbf{x}, \mathbf{y}]$ where $[\cdot, \cdot]$ stands for the concatenation along the sequence dimension. Similar to computing the loss on unlabeled data, a randomly sampled mask vector $\mathbf{m}$ and its complementary mask $\neg\mathbf{m}$ are manipulated over $[\mathbf{x}, \mathbf{y}]$, which results in a masked pair $([\mathbf{x}, \mathbf{y}]^{\mathbf{m}}, [\mathbf{x}, \mathbf{y}]^{\neg\mathbf{m}})$. We compute the pre-training loss over this masked pair through Eq. (2). In practice, we observe that the individual predictions for masked tokens of either $\mathbf{x}^{\neg\mathbf{m}}$ or $\mathbf{y}^{\neg\mathbf{m}}$ on the decoder performs better, particularly, it exhibits stronger stability during pre-training. Therefore, we compute the alignment loss over $\ell_a([\mathbf{x}, \mathbf{y}]^{\mathbf{m}}, [\mathbf{x}, \mathbf{y}]^{\neg\mathbf{m}})$ as:

$$\begin{aligned}&\sum_{m_i=1} \log P([\mathbf{x}, \mathbf{y}]|[\mathbf{x}, \mathbf{y}]^{\mathbf{m}}; \boldsymbol{\theta_e}) \\ &+ \sum_{m_i=1} \log P(x_i|\mathbf{x}^{\neg\mathbf{m}}_{<i}, [\mathbf{x}, \mathbf{y}]^{\mathbf{m}}; \boldsymbol{\theta}) \\ &+ \sum_{m_i=1} \log P(y_i|\mathbf{y}^{\neg\mathbf{m}}_{<i}, [\mathbf{x}, \mathbf{y}]^{\mathbf{m}}; \boldsymbol{\theta}).\end{aligned} \tag{5}$$

To sum up, the losses on the speech-text (D_{st}) and text-text (D_{tt}) sets involve the forward and backward sequence-to-sequence losses and an alignment loss for each example:

$$\mathcal{L}_p(\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{s},\mathbf{t})\in D_{st}}\Big\{\mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{S}^{\mathbf{m}}, \mathbf{t}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{t}^{\mathbf{m}}, \mathbf{z}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_a([\mathbf{S}, \mathbf{t}]^{\mathbf{m}}, [\mathbf{z}, \mathbf{t}]^{\neg\mathbf{m}})]\Big\} + \mathbb{E}_{(\mathbf{v},\mathbf{t})\in D_{tt}}\Big\{\mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{v}^{\mathbf{m}}, \mathbf{t}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_p(\mathbf{t}^{\mathbf{m}}, \mathbf{v}^{\neg\mathbf{m}}; \boldsymbol{\theta})] + \mathbb{E}_{\mathbf{m}}[\ell_a([\mathbf{v}, \mathbf{t}]^{\mathbf{m}}, [\mathbf{v}, \mathbf{t}]^{\neg\mathbf{m}})]\Big\} \tag{6}$$

where the sequence of speech representations $\mathbf{S}$ and the sequence of text ids $\mathbf{t}$ are allowed to be concatenated in the formula just for convenience. In addition to the above conventional loss across all the paired training sets, we follow mSLAM (Bapna et al., 2022) to leverage the CTC loss (Graves et al., 2006; Graves & Jaitly, 2014) to enforce the alignment of the encoder representations between speech and text, which is only activated on ASR data.

2.4. Fine-tuning

The fine-tuning method is crucial to unlock the capability of pre-trained models. In this section, We explain the process of fine-tuning our multilingual speech-text models for downstream tasks related to both speech and text.

Direct fine-tuning A common way of adapting a pre-trained model to a specific downstream task is to continue training the model exclusively on labeled data from that task, usually in combination with a relatively smaller learning rate. We use this direct fine-tuning for those tasks that are not included in our labeled data sets, *e.g.*, text classification.

Gradual fine-tuning To mitigate the discrepancy between pre-training and fine-tuning due to the artificial [MASK] tokens in pre-training (Yang et al., 2019), we propose a two-stage gradual fine-tuning method. At the first stage, as we incorporate labeled datasets from the downstream tasks of our interest during the pre-training phase, we keep training the models by using the sequence-to-sequence loss (Eq. (4)) on the paired data but disabling [MASK] tokens. Only $P(\mathbf{y}|\mathbf{x};\boldsymbol{\theta})$ for a pair $(\mathbf{x}, \mathbf{y})$ is used after the mask operation is eliminated. Since we fine-tune the model over the combination set of multiple tasks with numerous languages, we call this fine-tuning method at this stage as *multi-task multilingual fine-tuning*. At the second stage, we further continue fine-tuning the model on one of tasks from the first-stage combination set.

Noisy fine-tuning When fine-tuning speech-text tasks, an augmentation method directly acting on speech spectrogram is exploited to prevent the overfit on the limited supervised set (Park et al., 2019). However, the perturbations in the source introduced by the augmentation method tend to affect the decoder predictions in which errors may be accumulated and amplified at the later steps. To defend these errors from the decoder, we follow (Cheng et al., 2019) to add some noise to decoder inputs. More specifically, we randomly replace some tokens in the decoder inputs with their synonym tokens measured by the word embeddings.

3. Experiments

3.1. Setup[1]

Data Following mSLAM (Bapna et al., 2022), we use the same unlabeled speech data of approximately 429k hours in 51 languages. The mC4 dataset spanning 101 languages is used as unlabeled text data. ASR data come from VoxPopuli, MLS, Babel, CoVoST and FLEURS. We only have two sources for AST data, CoVoST and FLEURS. We collect MT data from WMT and TED.

Model We use an identical Conformer layer from SLAM (Bapna et al., 2021) and mSLAM (Bapna et al., 2022). The Transformer layers in the decoder share a similar setting as Conformer layer. The Adam optimizer (Kingma & Ba, 2014) is applied to pre-traing while AdamW (Loshchilov & Hutter, 2017) is used for fine-tuning.

Pre-training The batch sizes per TPU for speech-only, text-only, AST, ASR and MT data are 4, 8, 1, 1 and 1. We mask approximately 50% of the speech frames with spans of length up to 10 (Chung et al., 2021). However, for text inputs, we mask a continuous span of around 50% of words except for MT tasks where the mask ratio is 25%. The loss coefficients related to speech-only and text-only data are set to 1. The loss coefficients for the text to speech and alignment tasks are 0.1 while speech to text tasks need a slightly higher loss coefficient 0.3 for the decoder loss. We pre-train two sets of speech-text models in which two different text vocabularies are used, *i.e.*, a character-level model (Mu^{2}SLAM-char) of 4096 chars and a spm-level model (Mu^{2}SLAM-spm) of $64k$ word pieces. These two models run on 256 TPUv4 chips for $1.5M$ steps.

Fine-tuning We fine-tune our pre-trained models on CoVoST-2 multilingual speech translation (Wang et al., 2021b), VoxPopuli multilingual speech recognition (Wang et al., 2021a), and XTREME multilingual text understanding (Hu et al., 2020) benchmarks. We report the detokenized BLEU scores calculated by the SacreBLEU script (Post, 2018). For each fine-tuning tasks, we use grid search to tune the hyperparameters including batch sizes per TPU over $\{2, 4, 8\}$, learning rates over $\{0.5, 1, 2, 3, 5\}$, dropout ratios for encoder inputs and Transformer decoder over $\{0.1, 0.3\}$, warm-up steps over $\{4k, 8k, 16k\}$. Generally, we observe that speech-related and text-related tasks are not very sen-

[1]More details for the setup can be found in the appendix.

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**

Table 1. Speech translation results on the CoVoST 2 dataset.

sitive to the batch size so we use 8. Speech-related tasks prefer a larger learning rate of 5 while text-related tasks needs a smaller one of 1 or 0.5. The warm-up steps are universally set to $16k$. The pre-trained spm-level model is in favor of a larger dropout of 0.3. In our multi-task multilingual fine-tuning experiments, the training examples of AST, ASR and MT for a batch is set to 4, 2 and 2. For AST, ASR and MT, we randomly incorporate synonym noises into decoder inputs, the noise ratio is set to 0.06. All of fine-tuning experiments are conducted on 64 TPUv4 chips. Except for the multi-task multilingual fine-tuning experiments in which we select a maximum fine-tuning step of $300k$ and report results from the last checkpoint, we pick the best model based on validation sets.

3.2. Multilingual Speech Translation

Table 1 shows BLEU scores on the CoVoST 2 dataset by fine-tuning the pre-trained models on English to non-English (en-xx) and non-English to English (xx-en) language pairs. We try three fine-tuning setups: (1) direct multilingual fine-tuning with xx-en or en-xx language pairs; (2) multi-task multilingual fine-tuning with all of available language pairs from AST, ASR and MT; (3) gradual fine-tuning by further training the model on xx-en or en-xx language pairs. We observe that direct fine-tuning with only AST data can already obtain better performance against XLS-R and 0.6B mSLAM models (up to $+1.5$ BLEU points). When multi-task multilingual fine-tuning is applied, the model can achieve better results on xx-en but lower scores on en-xx. As this model can be used for multiple tasks, not just limited to AST, it is reasonable that the good performance on en-xx can not be kept. We believe that a model with larger capacity is able to improve the results on en-xx. However, in terms of average scores on all of language pairs in AST, the model still makes some improvements (up to $+0.7$ BLEU points). The best models are delivered by using gradual fine-tuning on Mu^{2}SLAM-char and Mu^{2}SLAM-spm. We establish new SOTA results on xx-en with $+1.9$ BLEU gains compared to Maestro (Chen et al., 2022). Meanwhile, we also notice that Mu^{2}SLAM-spm reaps more benefits from multi-task multilingual fine-tuning, particularly on en-xx. We speculate that smaller granularity of character-level vocabulary is conductible to language transfer because of more characters shared across different languages and the domination of English tokens during the SPM vocabulary creation. Thus Mu^{2}SLAM-char already gets good results without very advanced fine-tuning techniques. The comparison between direct fine-tuning and gradual fine-tuning clearly show that the multi-stage fine-tuning is indispensable for our models to get better results.

Model	# Encoder	En-De	En-Ca	En-Ar	En-Tr	Avg
wav2vec-2.0	0.3B	23.8	32.4	17.4	15.4	22.3
wav2vec-2.0 + LM	0.3B	24.9	34.0	18.0	16.7	23.4
SLAM-TLM	0.6B	27.5	33.4	18.9	16.6	24.1
SLMA-TLM-STM→w2v-bert	0.6B	27.1	34.2	21.2	17.5	25.0
SpeechLM-P	0.3B	27.6	35.9	21.7	19.5	26.2
Multi-task Multilingual FT						
Mu^{2}SLAM-char	0.6B	26.9	34.2	20.2	18.4	24.9
Mu^{2}SLAM-spm.	0.6B	25.1	32.4	17.5	16.1	22.8
Multi-task Multilingual FT → Per-language FT						
Mu^{2}SLAM-char	0.6B	**29.4**	**36.1**	**23.3**	**20.4**	**27.3**
Mu^{2}SLAM-spm.	0.6B	29.1	**36.1**	22.8	20.3	27.1

Table 2. Speech translation results on four major English to non-English tasks from the CoVoST 2 dataset.

Method	WER
Transducer as Decoder	
XLS-R (0.3B)	12.8
XLS-R (1B)	10.6
mSLAM-TLM (0.6B)	9.4
mSLAM-CTC (0.6B)	9.2
mSLAM-CTC (2B)	9.1
Maestro (0.6B)	8.1
Whisper (1.6B)	13.6
Transformer as Decoder, ASR Multilingual FT	
Mu^{2}SLAM-char (0.7B)	9.8
Mu^{2}SLAM-spm (0.7B)	9.5
Transformer as Decoder, Multi-task Multilingual FT	
Mu^{2}SLAM-char (0.7B)	31.5
Mu^{2}SLAM-spm (0.7B)	32.5
Transformer as Decoder, Multi-task Multilingual FT → ASR multilingual FT	
Mu^{2}SLAM-char (0.7B)	9.7
Mu^{2}SLAM-spm (0.7B)	9.2

Table 3. Speech recognition results on the VoxPopuli dataset.

Additionally, we compare the results between Whisper (Radford et al., 2022), which evaluates the benchmark in a zero-shot manner, and our proposed method Mu^{2}SLAM. The superior performance of Whisper on xx-en suggests that the straightforward scaling of weakly supervised pre-training holds great potential. However, we think the comparison isn't exactly fair since Whisper (Radford et al., 2022) is using an order of magnitude larger proprietary dataset and also evaluating out of domain. The Whisper (Radford et al., 2022) paper highlights the effectiveness of weakly supervised pre-training for building a general-purpose ASR + AST system, whereas our method focuses more on leveraging unlabeled speech and text pre-training data, and multilingual multitask supervised data towards learning a single model for speech and text understanding. Morever, the Whisper model is much larger than our Mu^{2}SLAM model (1.6B vs. 0.6B).

Table 2 shows AST results on four English to Non-English (en-xx) directions from CoVoST 2 by following the identical setting in baseline methods. To make a fair comparison, at the second stage in the gradual fine-tuning, we fine-tune the model only with a single language pair. Our models outperform the previous best SpeechLM (Zhang et al., 2022a) with up to $+1.1$ BLEU points and Mu^{2}SLAM-char still performs slightly better than Mu^{2}SLAM-spm on en-xx.

To sum up, we have the following findings from these two tables: (1) gradual fine-tuning is tremendously helpful to improve model performance; (2) Mu^{2}SLAM-spam model gains much more from gradual fine-tuning; (3) Mu^{2}SLAM-spam is in favor of xx-en translation directions while Mu^{2}SLAM-char performs much better on en-xx.

3.3. Multilingual Speech Recognition

We present ASR results on the multilingual VoxPopuli dataset in Table 3. There are three different fine-tuning setups, multilingual fine-tuning only with ASR, multi-task multilingual fine-tuning and gradual fine-tuning from multi-task multilingual fine-tuning to ASR-only multilingual fine-tuning. If we directly evaluate the models fine-tuned with multi-task multilingual fine-tuning, we can find both of them can not achieve reasonable numbers in this benchmark. It might be because AST data dominates the multi-task multilingual fine-tuning and deteriorates the monotonic alignments between encoder and decoder but ASR tasks are very sensitive to heterogeneous data.

In the other two setups, Mu^{2}SLAM-spm performs better than Mu^{2}SLAM-char, particularly, Mu^{2}SLAM-spm bene-

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**

Table 4. Text classification results on the validation sets in XNLI.

fits more from gradual fine-tuning with around +0.5 WER gains. It indicates that speech-text model based on the SPM vocabulary has more potential of attaining better results if being elaborately fine-tuned with observing more paired data. Before multi-task multilingual fine-tuning, our best model (Mu^{2}SLAM-spm) can only beat XLS-R. However, the exploitation of multi-task multilingual fine-tuning makes our best model achieve similar performance against mSLAM although it still lags behind Maestro. That is because Transformer (Vaswani et al., 2017) rather than RNN Transducer (Graves, 2012) is applied as a decoder in our speech-text ASR model, but it has demonstrated the effectiveness of jointly pre-training encoder and decoder for learning better speech-text representations to even dispense with Transducer decoder. The Whisper model also adopts the Transformer model with a larger model capacity and benefits from a larger amount of pre-training data. However, it is worth mentioning that the Whisper model is not particularly proficient in ASR tasks and, in fact, it demonstrates significantly poorer performance compared to our Mu^{2}SLAM approach.

3.4. Multilingual Text Understanding

We also investigate the capability of our speech-text models with respect to text understanding on two representative evaluation tasks from XTREME (Hu et al., 2020), from which we pick up two representative evaluation tasks, XNLI classification and TyDiQA question answering tasks.

XNLI classification task As shown in Table 4, in the zero-shot setting, our model underperforms the mono-modal multilingual model of the similar size, mT5-Base (0.6B). We think that capacity dilution results in the degeneration of our speech-text models against text-only modeling. The best speech-text joint model Mu^{2}SLAM-spm can achieve better results than mT5-Small (0.3B) which is roughly half the model size of Mu^{2}SLAM-spm. We believe the increased model capacity can compensate for the decease of Mu^{2}SLAM against mT5. When compared with mSLAM models, Mu^{2}SLAM consistently performs significantly better than the mSLAM(0.6B) model (66.4 vs. 58.9) and comparable with the mSLAM model (2B). More specifically, Mu^{2}SLAM obtains notable improvements on non-European languages. The comparison between spm-level and character-level models indicates that spm-level tokens are better to capture the text meaning. Similar findings can be observed when moving to the *Translate-Train-All* setting.

TyDiQA task Table 5 shows F1/EM results on a multilingual question answering task, TyDiQA. In the zero-shot setting, similar to classification results in Table 4, Mu^{2}SLAM-spm maintains better performance than Mu^{2}SLAM-char. However, our Mu^{2}SLAM models can not even surpass mT5-base although we achieve significantly better results than mT5-base on XNLI. We analyze the breakdown results on each language pair attached in the appendix. We find our models are able to deliver good results on English but relatively worse results on non-English languages. More specifically, our models nearly approach 0 on Bengali language. After looking into the model outputs, we find the model can not properly output the correct tokens in non-English languages. We ascribe this issue to the language embeddings which do not specialize in generation outside the fine-tuning languages. Likewise, we evaluate our models on the *Translate-Train-All* setting. As expected, our models are better than mT5-Small but worse than mT5-Base.

3.5. Analysis

Effect of paired data To study the effect of speech-text and text-text labeled data, we conduct extensive experiments by using different combination sets of AST, ASR and MT data during pre-training and fine-tuning. In Row 1-4, only AST data is enabled in fine-tuning. The best model comes from Row 4 which digests all available speech-text data. It is interesting that removing ASR data improves the model performance on en-xx directions. We speculate that translation data (AST and MT) is important to learn the alignment between encoder and decoder but ASR with very strong monotonic alignment hurts the alignment, particularly for en-xx in which a specific non-English translation data is not as abundant as English. When applying multi-task multilingual fine-tuning, we find the model with all data in pre-training does not perform the best. Row 6 without MT involved in pre-training takes the lead in the AST results. It is probably because text occupies model capacity in encoder pre-training but the effect of MT data is made up during the fine-tuning stage. These comparisons suggest that multi-task pre-training is beneficial to learning general speech-text representations if we do not have any assumption on down-

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

Table 6. Effect of paired data when being incorporated into pretraining or finetuning stages.

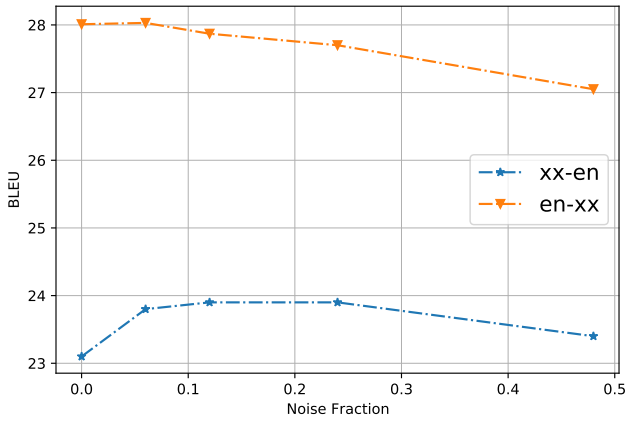

Figure 2. Effect of noisy fine-tuning when changing the noise ratio.

stream tasks. As we want to evaluate our models on both speech and text tasks, we incorporate all of available labeled data related to speech and text in our model pre-training.

Effect of noisy fine-tuning We randomly replace decoder inputs with their synonym tokens in noisy fine-tuning. Figure 2 shows the sensitive study results on the noise ratio. When using the Mu^{2}SLAM-char model on xx-en, the BLEU score has a drastic change when the noise ratio increases from 0 to 0.06, then reaches the plateau, finally drops a lot as it increases to 0.48. However, for en-xx, the noisy fine-tuning has subtle improvements when using a non-zero small noise ratio (0.06). If increasing it further, it hurts the model performance severely. The improvement differences between xx-en and en-xx imply that the noise on the decoder is more favourable to the same or similar languages (xx-en) rather than a set of diverse languages (en-xx) (Cheng et al., 2022). In addition, the word embeddings can not accurately measure the similarities between different languages.

4. Related Work

Speech-text Pre-training Pre-training methods have dominated research and industry fields due to their superior capabilities of exploiting unlabeled data. Particularly, in NLP and speech. A lot of representatives, such as BERT (Devlin et al., 2019), XLNET (Yang et al., 2019), T5 (Raffel et al., 2020), MASS (Song et al., 2019), wav2vec (Baevski

et al., 2020), Hubert (Hsu et al., 2021) and so on, come out to improve mono-modal model performance. In recent days, the research community has started to move towards speech-text joint training, aiming to learn the shared representation space between speech and text, which can be roughly categorized into, encoder-only pre-training (Bapna et al., 2021; 2022; Zhang et al., 2022a), encoder-decoder pre-training (Ao et al., 2021; Lakhotia et al., 2021; Chen et al., 2022; Sainath et al., 2022; Zhou et al., 2022; Zhang et al., 2022b; Tang et al., 2022; Popuri et al., 2022; Radford et al., 2022). Mu^{2}SLAM adopts an encoder-decoder backbone model by minimizing the utilization of modality-specific blocks only with a CNN used to extract speech representations, which dramatically simplifies the cross-modal model architecture and also enforces the representation sharing between different languages and modalities. Among them, the most related Maestro (Chen et al., 2022) also incorporates ASR, AST and MT data into their method, however, their model training has to rely on a pre-trained mSLAM as initialization and applies a duration model to over-sample the text which can not be activated during fine-tuning. In contrast, Mu^{2}SLAM pre-trains the model from scratch which can be applied in the downstream tasks without wasting any parameter. We also verify our models on text-related benchmarks while they just focus on speech tasks. In addition, a text-to-speech loss is also introduced in pre-training which endows our model with the ability of speech generation. We leave it as the future exploration.

Multilingual Pre-training The great success of multilingual text pre-training like mBERT (Devlin, 2018), XLM-R (Conneau & Lample, 2019) and mT5 (Xue et al., 2020) incentivizes the speech research to move toward multilingual modeling and pre-training (Conneau et al., 2020; Babu et al., 2021; Bapna et al., 2022; Chen et al., 2022; Radford et al., 2022), which benefits from the cross-lingual transfer for learning joint representations across massive amounts of data across multiple languages (Conneau et al., 2019; Hu et al., 2020). Our approach is inspired by this research direction by involving multilingual speech and text spanning over 100 languages.

Multi-task Learning Multi-task learning is an effective approach that utilizes the training signals of related tasks to enhance the generalization performance of a model (Caruana, 1997). This technique has been successfully applied to improve various speech-related tasks, as evidenced by previous studies (Weiss et al., 2017; Tang et al., 2021; 2022; Chen et al., 2022; Bapna et al., 2022). In our paper, we adopt a unified approach by combining speech and text-related tasks into a single sequence-to-sequence model during the pre-training stage. Our aim is to leverage the training signals from diverse speech and/or text-related tasks, encompassing speech-only, text-only, ASR, AST, MT, and potentially TTS, in order to maximize the benefits of multi-task learning.

5. Conclusion

We have presented Mu^{2}SLAM for speech and text joint models based on a fully encoder-decoder model. Our pre-training models span more than 100 languages in speech and text and involve unlabeled data and labeled data from speech/text-only data, ASR, AST to MT. We introduce two training objectives to unify the unlabeled and labeled data in pre-training, and gradual fine-tuning and noisy fine-tuning to improve the model performance on downstream tasks. Extensive experiments on multilingual benchmarks show that our pre-training models can achieve very strong results with new SOTA on CoVoST and comparable performance against mSLAM on VoxPopuli, and narrow the gap between speech-text models and text-only models on text tasks.

6. Limitations and Future Work

Based on our extensive experiments, we have identified several limitations in this paper, which we believe open up potential avenues for future research and development.

1. While we pre-train our model on the text-to-speech task, we have not evaluated our approach on speech generation benchmarks. It would be beneficial to include them in our evaluation process to provide a more comprehensive assessment of the model's performance across different modalities.
2. Our model is limited to only 100 languages available in academic datasets unlike models trained on proprietary datasets, *e.g.*USM (Zhang et al., 2023). We plan to scale up our model beyond 100 language.
3. We have not integrated speech to speech tasks into our pre-training framework to further expand the capabilities of the model and explore the potential benefits of jointly training on these tasks (Popuri et al., 2022).
4. Our pre-trained model is limited in zero-shot speech translation and recognition, as well as zero-shot text generation task. It is worthwhile improving model's alignment transfer between unseen language pairs and modalities. One of possible directions is to switch to decoder-only models (Anil et al., 2023).
5. Our proposed approach still relies on speech representations as inputs in encoder , rather than solely relying on token-to-token transformations.

Acknowledgements

The authors would like to thank anonymous reviewers for insightful comments, which greatly contributed to the improvement of this paper. Special thanks are given to Yuan Cao and Zhehuai Chen for their invaluable discussions and contributions during the model training.

References

Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Ao, J., Wang, R., Zhou, L., Liu, S., Ren, S., Wu, Y., Ko, T., Li, Q., Zhang, Y., Wei, Z., et al. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. *arXiv preprint arXiv:2110.07205*, 2021.

Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. M., and Weber, G. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*, 2019.

Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., et al. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*, 2021.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. 2020.

Bapna, A., Chung, Y.-a., Wu, N., Gulati, A., Jia, Y., Clark, J. H., Johnson, M., Riesa, J., Conneau, A., and Zhang, Y. Slam: A unified encoder for speech and language modeling via speech-text joint pre-training. *arXiv preprint arXiv:2110.10329*, 2021.

Bapna, A., Cherry, C., Zhang, Y., Jia, Y., Johnson, M., Cheng, Y., Khanuja, S., Riesa, J., and Conneau, A. mslam: Massively multilingual joint pre-training for speech and text. *arXiv preprint arXiv:2202.01374*, 2022.

Barrault, L., Bojar, O., Costa-jussà, M. R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., Malmasi, S., Monz, C., Müller, M., Pal, S., Post, M., and Zampieri, M. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 1–61, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5301. URL https://aclanthology.org/W19-5301.

Barrault, L., Biesialska, M., Bojar, O., Costa-jussà, M. R., Federmann, C., Graham, Y., Grundkiewicz, R., Haddow, B., Huck, M., Joanis, E., Kocmi, T., Koehn, P., Lo, C.-k., Ljubešić, N., Monz, C., Morishita, M., Nagata, M., Nakazawa, T., Pal, S., Post, M., and Zampieri, M. Findings of the 2020 conference on machine translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pp. 1–55, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.wmt-1.1.

Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pp. 1–44, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL https://aclanthology.org/W13-2201.

Bojar, O., Chatterjee, R., Federmann, C., Haddow, B., Huck, M., Hokamp, C., Koehn, P., Logacheva, V., Monz, C., Negri, M., Post, M., Scarton, C., Specia, L., and Turchi, M. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 1–46, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-3001. URL https://aclanthology.org/W15-3001.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi, M. Findings of the 2017 conference on machine translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pp. 169–214, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4717. URL https://aclanthology.org/W17-4717.

Bojar, O., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Koehn, P., and Monz, C. Findings of the 2018 conference on machine translation (WMT18). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pp. 272–303, Belgium, Brussels, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6401. URL https://aclanthology.org/W18-6401.

Borsos, Z., Marinier, R., Vincent, D., Kharitonov, E., Pietquin, O., Sharifi, M., Teboul, O., Grangier, D., Tagliasacchi, M., and Zeghidour, N. Audiolm: a language modeling approach to audio generation. *arXiv preprint arXiv:2209.03143*, 2022.

Caruana, R. c. *Machine Learning*, 28(1):41–75, Jul 1997. ISSN 1573-0565. doi: 10.1023/A:1007379606734. URL https://doi.org/10.1023/A:1007379606734.

Chen, Z., Bapna, A., Rosenberg, A., Zhang, Y., Ramabhadran, B., Moreno, P., and Chen, N. Maestro-u: Leveraging joint speech-text representation learning for zero supervised speech asr. *arXiv preprint arXiv:2210.10027*, 2022.

Cheng, Y., Jiang, L., and Macherey, W. Robust neural machine translation with doubly adversarial inputs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4324–4333, 2019.

Cheng, Y., Bapna, A., Firat, O., Cao, Y., Wang, P., and Macherey, W. Multilingual mix: Example interpolation improves multilingual neural machine translation. In *ACL*, 2022.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Chung, Y.-A., Zhang, Y., Han, W., Chiu, C.-C., Qin, J., Pang, R., and Wu, Y. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021.

Conneau, A. and Lample, G. Cross-lingual language model pretraining. In *NeurIPS*, 2019.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. Unsupervised cross-lingual representation learning at scale. In *ACL*, 2019.

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., and Auli, M. Unsupervised cross-lingual representation learning for speech recognition. *arXiv preprint arXiv:2006.13979*, 2020.

Conneau, A., Ma, M., Khanuja, S., Zhang, Y., Axelrod, V., Dalmia, S., Riesa, J., Rivera, C., and Bapna, A. Fleurs: Few-shot learning evaluation of universal representations of speech. *arXiv preprint arXiv:2205.12446*, 2022.

Devlin, J. Multilingual bert. *arXiv preprint arXiv:2010.11934*, 2018.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

Gales, M. J. F., Knill, K., Ragni, A., and Rath, S. P. Speech recognition and keyword spotting for low-resource languages: Babel project research at cued. In *SLTU*, 2014.

Graves, A. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*, 2012.

Graves, A. and Jaitly, N. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pp. 1764–1772. PMLR, 2014.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., and Mohamed, A. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

Hu, J., Ruder, S., Siddhant, A., Neubig, G., Firat, O., and Johnson, M. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *ICML*, 2020.

Kasai, J., Pappas, N., Peng, H., Cross, J., and Smith, N. A. Deep encoder, shallow decoder: Reevaluating non-autoregressive machine translation. *arXiv preprint arXiv:2006.10369*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lakhotia, K., Kharitonov, E., Hsu, W.-N., Adi, Y., Polyak, A., Bolte, B., Nguyen, T.-A., Copet, J., Baevski, A., Mohamed, A., et al. On generative spoken language modeling from raw audio. *TACL*, 2021.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., and Le, Q. V. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

Popuri, S., Chen, P.-J., Wang, C., Pino, J., Adi, Y., Gu, J., Hsu, W.-N., and Lee, A. Enhanced direct speech-to-speech translation using self-supervised pre-training and data augmentation. *arXiv preprint arXiv:2204.02967*, 2022.

Post, M. A call for clarity in reporting bleu scores. *WMT*, 2018.

Pratap, V., Xu, Q., Sriram, A., Synnaeve, G., and Collobert, R. Mls: A large-scale multilingual dataset for speech research. *arXiv preprint arXiv:2012.03411*, 2020.

Qi, Y., Sachan, D., Felix, M., Padmanabhan, S., and Neubig, G. When and why are pre-trained word embeddings useful for neural machine translation? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 529–535, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2084. URL https://aclanthology.org/N18-2084.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding by generative pre-training, 2018.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*, 2022.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P. J., et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 2020.

Sainath, T. N., Prabhavalkar, R., Bapna, A., Zhang, Y., Huo, Z., Chen, Z., Li, B., Wang, W., and Strohman, T. Joist: A joint speech and text streaming model for asr. *arXiv preprint arXiv:2210.07353*, 2022.

Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. Mass: Masked sequence to sequence pre-training for language generation. In *ICML*, 2019.

Tang, Y., Pino, J., Wang, C., Ma, X., and Genzel, D. A general multi-task learning framework to leverage text data for speech to text tasks. In *ICASSP*, 2021.

Tang, Y., Gong, H., Dong, N., Wang, C., Hsu, W.-N., Gu, J., Baevski, A., Li, X., Mohamed, A., Auli, M., et al. Unified speech-text pre-training for speech translation and recognition. *arXiv preprint arXiv:2204.05409*, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Wang, C., Riviere, M., Lee, A., Wu, A., Talnikar, C., Haziza, D., Williamson, M., Pino, J., and Dupoux, E. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. *arXiv preprint arXiv:2101.00390*, 2021a.

Wang, C., Wu, A., and Pino, J. Covost 2 and massively multilingual speech-to-text translation. In *Interspeech*, 2021b.

Weiss, R. J., Chorowski, J., Jaitly, N., Wu, Y., and Chen, Z. Sequence-to-sequence models can directly translate foreign speech. *arXiv preprint arXiv:1703.08581*, 2017.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*, 2020.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, 2019.

Zhang, Y., Han, W., Qin, J., Wang, Y., Bapna, A., Chen, Z., Chen, N., Li, B., Axelrod, V., Wang, G., et al. Google usm: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*, 2023.

Zhang, Z., Chen, S., Zhou, L., Wu, Y., Ren, S., Liu, S., Yao, Z., Gong, X., Dai, L., Li, J., et al. Speechlm: Enhanced speech pre-training with unpaired textual data. *arXiv preprint arXiv:2209.15329*, 2022a.

Zhang, Z., Zhou, L., Ao, J., Liu, S., Dai, L., Li, J., and Wei, F. Speechut: Bridging speech and text with hidden-unit for encoder-decoder based speech-text pre-training. *arXiv preprint arXiv:2210.03730*, 2022b.

Zhou, X., Wang, J., Cui, Z., Zhang, S., Yan, Z., Zhou, J., and Zhou, C. Mmspeech: Multi-modal multi-task encoder-decoder pre-training for speech recognition. *ArXiv*, abs/2212.00500, 2022.

A. Setup

A.1. Data

Speech-only Data Following mSLAM (Bapna et al., 2022), we use the same unlabeled speech data of approximately 429k hours in 51 languages from VoxPopuli (Wang et al., 2021a), Common Voice (Ardila et al., 2019), MLS (Pratap et al., 2020) and Babel (Gales et al., 2014).

Text-only Data The mC4 dataset (Xue et al., 2020) spanning 101 languages is used as unlabeled text data by adopting a temperature-based sampling to over-sample the low-resource languages where temperature is 3.

Speech-Text Data We use ASR data from VoxPopuli of approximately 1.3k hours with 14 languages, MLS of 80 hours with 8 languages, Babel of 1k hours with 17 languages, CoVoST of 2.9k hours with 22 languages (Wang et al., 2021b), FLEURS of 1.4k (Conneau et al., 2022) hours with 101 languages. We only have two sources for AST data, CoVoST of 9.5k hours spanning 22 languages and FLEURS of 1.4k hours spanning 101 languages.

Text-Text Data The paired text-text data comes from WMT and TED translation tasks, which are identical as the MT sets in mSLAM(Bapna et al., 2022). More specifically, we collect MT data from WMT and TED which has the similar language coverage as CoVoST. We pair WMT20 (Barrault et al., 2020) for ja, ta, WMT19 (Barrault et al., 2019) for de, ru, zh, WMT18 (Bojar et al., 2018) for et, tr, WMT17 (Bojar et al., 2017) for lv, WMT15 (Bojar et al., 2015) for fr, WMT13 (Bojar et al., 2013) for es, and TED59 (Qi et al., 2018) for ar, fa, id, it, mn, nl, pt, sl, sv, leaving ca and cy unpaired. Because the language distribution in this combination set is highly skewed, we also apply the similar temperature-based data sampling with temperature as 2.

A.2. Model and Hyperparameters

Model setup We use an identical Conformer layer from SLAM (Bapna et al., 2021) and mSLAM (Bapna et al., 2022), in which the model dimension is 1024, feedforward hidden dimension is 4096, convolution kernel size is 5 and the number of attention heads is 8. The Transformer layers in the decoder share the same setting as Conformer layer in terms of model dimension, hidden dimension and attention heads but we set dropout to 0.1 for the Transformer layers rather than the default 0 in the Conformer layers. We use the same learning schedule as Transformer during pre-training and fine-tuning but warmup steps are set to $40k$ and $16k$ respectively. The Adam optimizer (Kingma & Ba, 2014) is applied to pre-traing with learning rate as 3 while AdamW (Loshchilov & Hutter, 2017) is used as fine-tuning optimizer with weight decay rate as 0.01.

Pre-training setup For text masking, we follow (Devlin et al., 2019) by replacing the masked tokens with (1) the [MASK] token 80% of the time (2) a random token 10% of the time (3) the unchanged i-th token 10% of the time. For speech masking, if a token needs to be masked, we just simply replace it with the [MASK] token.

Table 7. BLEU results on xx-en CoVoST.

	High-resource				Mid-resource					Low-resource		
xx-en	fr	de	es	ca	fa	it	ru	pt	zh	tr	ar	et
Train Hours	264h	184h	113h	136h	49h	44h	18h	10h	10h	4h	2h	3h
XLS-R (0.3B)	32.9	26.7	34.1	28.7	5.9	29.0	26.4	28.3	4.9	4.6	3.0	3.5
XLS-R (1B)	36.2	31.2	37.9	31.9	9.6	33.1	37.0	39.3	8.7	12.8	12.2	8.3
XLS-R (2B)	37.6	33.6	39.2	33.8	12.9	34.9	39.5	41.8	9.4	16.7	17.1	11.1
xx-en Multilingual AST&MT FT												
mSLAM-TLM (0.6B)	36.8	32.8	38.8	33.6	9.7	34.6	41.2	32.1	8.8	12.2	12.6	16.6
mSLAM-CTC (0.6B)	38.6	36.1	40.6	35.2	7.2	37.0	47.5	36.4	10.8	15.6	14.2	20.3
mSLAM-CTC (2B)	39.0	35.9	41.0	35.4	9.7	37.3	48.4	42.8	10.0	24.2	19.3	22.6
xx-en/en-xx Multilingual AST FT												
Mu^{2}SLAM-char(0.6B)	36.5	31.3	38.1	34.0	12.2	33.8	40.0	40.3	14.5	21.2	25.6	14.0
Mu^{2}SLAM-spm (0.6B)	35.6	30.8	37.9	33.1	11.1	33.3	40.0	40.4	14.8	20.7	27.3	12.8
Multi-task Multilingual FT												
Mu^{2}SLAM-char(0.6B)	39.1	34.9	40.6	34.6	14.7	34.7	42.7	43.7	15.3	23.5	26.3	14.8
Mu^{2}SLAM-spm (0.6B)	38.4	34.7	40.5	34.2	14.9	35.0	41.2	43.2	15.8	23.3	29.5	14.9
Multi-task Multilingual FT → xx-en/en-xx Multilingual AST FT												
Mu^{2}SLAM-char(0.6B)	38.7	35.0	40.2	34.2	14.5	34.2	43.2	43.3	15.2	23.6	27.2	14.6
Mu^{2}SLAM-spm (0.6B)	38.5	34.9	40.6	34.4	15.0	34.9	43.4	44.0	16.3	23.9	31.4	15.4

	Low-resource									Average			
xx-en	mn	nl	sv	lv	sl	ta	ja	id	cy	high	mid	low	all
Train Hours	3h	7h	2h	2h	2h	2h	2h	2h	2h				
XLS-R (0.3B)	0.4	22.0	10.3	6.0	6.6	0.2	0.6	1.4	2.5	30.6	18.9	5.1	13.2
XLS-R (1B)	0.8	28.2	24.7	16.0	16.7	0.3	1.9	10.3	8.6	34.3	25.5	11.7	19.3
XLS-R (2B)	1.6	31.7	29.6	19.5	19.6	0.5	3.5	16.5	14.0	36.1	27.7	15.1	22.1
xx-en Multilingual AST&MT FT													
mSLAM-TLM (0.6B)	0.3	33.2	26.3	15.2	19.8	0.5	1.3	3.7	5.6	35.5	25.3	12.3	19.8
mSLAM-CTC (0.6B)	0.9	36.3	31.7	19.8	25.6	0.5	2.4	6.1	7.7	37.6	27.8	15.1	22.4
mSLAM-CTC (2B)	0.8	37.6	38.5	26.8	32.3	0.6	3.3	8.8	6.7	37.8	29.6	18.5	24.8
xx-en/en-xx Multilingual AST FT													
Mu^{2}SLAM-char(0.6B)	2.8	27.0	33.0	20.1	21.7	1.4	7.3	27.1	17.3	35.0	28.2	18.2	23.8
Mu^{2}SLAM-spm(0.6B)	2.7	26.8	30.6	19.7	24.3	2.4	9.1	29.4	18.9	34.4	27.9	18.7	23.9
Multi-task Multilingual FT													
Mu^{2}SLAM-char(0.6B)	4.0	31.6	34.6	20.7	26.0	1.9	9.0	29.2	24.2	37.3	30.2	20.5	26.0
Mu^{2}SLAM-spm (0.6B)	4.3	31.6	31.5	20.5	25.7	2.3	11.0	33.8	26.2	37.0	30.0	21.2	26.3
Multi-task Multilingual FT → xx-en/en-xx Multilingual AST FT													
Mu^{2}SLAM-char(0.6B)	4.1	31.6	34.2	21.1	25.6	2.2	9.1	30.8	24.0	37.0	30.0	20.7	26.0
Mu^{2}SLAM-spm (0.6B)	4.5	32.1	34.8	20.4	27.5	2.5	11.8	36.1	27.3	37.0	30.6	23.5	27.1

Table 8. BLEU results on en-xx CoVoST.

en-xx Train Hours	ar 430h	ca 430h	cy 430h	de 430h	et 430h	fa 430h	id 430h	ja 430h	lv 430h	mn 430h	sl 430h	sv 430h	ta 430h	tr 430h	zh 430h	all 430h
xx-en/en-xx Multilingual AST FT																
Mu^{2}SLAM-char(0.6B)	22.5	35.7	37.0	28.6	24.6	20.1	33.1	32.2	24.9	17.6	30.1	35.6	21.2	19.9	37.4	28.0
Mu^{2}SLAM-spm (0.6B)	21.2	35.0	35.3	27.7	23.5	20.2	33.0	32.0	23.4	16.8	28.0	34.4	20.5	18.7	36.8	27.1
Multi-task Multilingual FT																
Mu^{2}SLAM-char(0.6B)	20.2	34.2	35.9	26.9	23.1	19.2	31.8	30.9	23.0	16.3	27.8	34.5	19.4	18.4	34.9	26.4
Mu^{2}SLAM-spm (0.6B)	17.5	32.4	32.8	25.1	20.8	18.9	31.0	28.9	19.6	14.6	23.4	32.0	17.7	16.1	32.9	24.2
Multi-task Multilingual FT → xx-en/en-xx Multilingual AST FT																
Mu^{2}SLAM-char(0.6B)	22.9	36.3	37.4	29.0	25.1	20.2	33.5	32.5	25.5	17.9	30.7	36.2	20.9	20.1	38.1	28.4
Mu^{2}SLAM-spm (0.6B)	22.2	35.6	36.6	28.7	24.2	20.6	33.8	32.1	24.4	17.2	29.4	35.5	20.9	19.4	37.4	27.9

Table 9. VoxPopuli ASR results in terms of WER.

	en	de	it	fr	es	pl	ro	hu
Train Hours	543h	282h	91h	211h	166h	111h	89h	63h
XLS-R (0.3B)	10.2	13.0	19.2	12.6	9.8	9.6	7.9	11.6
XLS-R (1B)	8.8	11.5	15.1	10.8	8.2	7.7	7.3	9.6
mSLAM-TLM (0.6B)	7.3	8.9	15.6	9.3	8.7	6.5	8.5	8.4
mSLAM-CTC (0.6B)	7.1	8.9	15.6	9.3	8.6	6.5	8.5	8.1
mSLAM-CTC (2B)	7.0	8.7	15.4	9.4	8.4	6.4	7.8	8.4
Transformer as Decoder, ASR Multilingual FT								
Mu^{2}SLAM-char (0.7B)	8.0	10.2	16.4	9.7	9.1	7.0	8.0	9.0
Mu^{2}SLAM-spm (0.7B)	7.5	8.9	14.4	9.1	7.9	7.1	7.5	8.9
Transformer as Decoder, Multi-task Multilingual FT								
Mu^{2}SLAM-char (0.7B)	28.1	29.4	48.5	32.2	36.0	29.2	32.5	31.5
Mu^{2}SLAM-spm (0.7B)	28.2	29.7	49.5	32.4	36.3	29.7	32.9	32.5
Transformer as Decoder, Multi-task Multilingual FT $\rightarrow$ ASR multilingual FT								
Mu^{2}SLAM-char (0.7B)	7.8	9.5	16.4	9.4	9.2	7.0	8.1	8.8
Mu^{2}SLAM-spm (0.7B)	7.2	8.5	13.9	8.6	7.5	6.8	7.1	8.7

	nl	cs	sl	fi	hr	sk	Avg
Labeled data	53h	62h	10h	27h	43h	35h	
Prior work (Babu et al., 2021)							
XLS-R (0.3B)	14.8	10.5	24.5	14.2	12.3	8.9	12.8
XLS-R (1B)	12.5	8.7	19.5	11.3	10.0	7.1	10.6
mSLAM-TLM (0.6B)	10.5	7.1	15.8	9.0	10.0	6.2	9.4
mSLAM-CTC (0.6B)	10.3	7.0	14.2	9.2	9.1	5.9	9.2
mSLAM-CTC (2B)	10.5	6.8	15.1	8.7	9.1	6.0	**9.1**
Transformer as Decoder, ASR Multilingual FT							
Mu^{2}SLAM-char (0.7B)	11.3	7.7	15.0	10.1	8.9	6.4	9.8
Mu^{2}SLAM-spm (0.7B)	11.4	7.6	17.1	10.5	8.8	6.4	9.5
Transformer as Decoder, Multi-task Multilingual FT							
Mu^{2}SLAM-char (0.7B)	28.1	20.0	34.6	31.9	39.2	29.3	32.8
Mu^{2}SLAM-spm (0.7B)	28.4	29.5	35.8	32.8	39.8	29.9	33.4
Transformer as Decoder, Multi-task Multilingual FT $\rightarrow$ ASR multilingual FT							
Mu^{2}SLAM-char (0.7B)	11.6	7.1	16.3	10.3	8.5	6.1	9.7
Mu^{2}SLAM-spm (0.7B)	10.9	7.4	16.3	10.7	8.6	6.5	9.2

Table 10. XNLI dev accuracy for all 15 languages.

Model	en	ar	bg	de	el	es	fr	hi	ru	sw	th	tr	ur	vi	zh	Avg
mT5-Small (0.3B)	79.6	62.2	67.8	64.8	65.8	68.4	66.2	59.0	65.3	55.4	63.2	58.9	54.5	61.8	63.4	63.8
mT5-Base (0.6B)	84.5	71.2	76.9	75.6	76.3	79.0	77.7	66.9	74.9	63.6	70.0	69.2	64.8	72.0	72.5	73.0
Zero-shot																
mSLAM-TLM (0.6B)	75.7	47.3	56.7	55.1	52.2	60.9	62.8	48.6	58.5	46.0	46.9	51.3	47.2	50.7	41.0	53.4
mSLAM-CTC (0.6B)	80.4	46.5	69.8	72.1	67.5	74.7	72.9	42.0	68.7	45.5	42.9	48.7	44.2	63.3	43.3	58.9
mSLAM-CTC (2B)	80.1	61.1	73.3	74.7	72.7	76.0	75.3	59.4	70.9	52.2	56.8	63.9	59.0	65.9	50.1	66.1
Mu^{2}SLAM-char (0.7B)	76.5	60.6	62.1	64.1	62.6	68.0	66.4	58.3	61.7	44.4	55.8	58.4	55.4	60.0	59.6	60.9
Mu^{2}SLAM-spm (0.7B)	81.2	65.7	67.4	71.4	65.7	74.1	74.2	57.2	69.2	51.1	63.0	63.9	55.1	66.0	70.8	66.4
Translate-Train-All																
mT5-Small (0.3B)	78.3	68.8	73.5	73.2	73.4	74.4	73.5	67.4	71.1	67.2	71.1	69.9	63.6	70.5	72.9	71.3
mT5-Base (0.6B)	85.9	78.8	82.2	81.6	81.4	83.0	82.1	77.0	81.1	74.8	78.6	78.4	73.3	78.9	80.2	79.8
mSLAM-TLM (0.6B)	74.3	64.2	68.7	69.5	69.2	70.2	71.4	64.5	65.4	63.4	65.6	65.9	62.4	67.3	64.4	67.1
mSLAM-CTC (0.6B)	81.1	63.5	76.7	76.0	73.1	77.8	76.4	63.6	73.1	64.1	64.9	66.8	60.5	68.4	64.5	70.0
mSLAM-CTC (2B)	84.1	80.2	80.1	78.7	82.9	80.5	74.4	72.1	76.8	71.7	73.8	76.2	69.8	75.9	72.8	76.1
Mu^{2}SLAM-char (0.7B)	79.0	72.0	75.0	74.0	75.2	77.7	75.4	70.1	72.4	69.7	70.9	72.1	65.7	73.2	71.2	72.9
Mu^{2}SLAM-spm (0.7B)	83.3	74.6	77.0	77.9	77.6	80.6	79.2	71.9	76.2	71.3	73.1	76.7	67.8	76.5	78.5	76.1

Table 11. TyDiQA GoldP test results (F1/EM) for all 9 languages.

Model	ar	bn	en	fi	id	ko	ru	sw	te	avg
Zero-shot										
mT5-Small (0.3B)	41.1/26.0	18.9/13.3	53.9/43.6	39.2/22.6	44.4/31.7	24.9/16.3	40.5/24.3	34.8/21.2	16.9/11.5	34.9/23.4
mT5-Base (0.6B)	67.1/50.4	40.7/22.1	71.8/60.9	67.0/52.2	71.3/54.5	49.5/37.7	54.9/32.6	60.4/43.9	40.6/31.1	58.1/42.8
Mu^{2}SLAM-char (0.7B)	10.8/3.4	1.8/1.8	58.1/50.2	31.6/21.9	37.7/24.8	12.3/9.8	23.1/12.1	37.0/27.1	13.4/10.8	25.1/18.0
Mu^{2}SLAM-spm (0.7B)	17.9/9.3	1.3/0.9	62.3/51.4	33.5/22.0	39.3/25.3	7.2/6.2	26.1/15.8	36.5/25.6	15.3/11.5	26.6/18.7
Translate-Train-All										
mT5-Small (0.3B)	56.8/39.7	37.2/21/2	57.1/46.6	50.9/37.2	60.1/45.1	40.4/29.3	50.7/33.6	51.5/35.3	29.3/18.1	48.2/34.0
mT5-Base (0.6B)	68.0/50.2	57.4/35.4	71.1/58.9	68.8/55.2	73.5/57.2	56.5/43.8	64.0/45.8	65.8/48.3	51.2/34.1	64.0/47.7
Mu^{2}SLAM-char (0.7B)	60.1/41.8	46.6/35.4	62.1/53.0	58.5/47.3	55.4/40.5	51.5/45.3	52.4/36.1	67.0/57.1	36.2/29.0	54.4/42.8
Mu^{2}SLAM-spm (0.7B)	61.8/42.8	46.0/32.7	67.3/55.9	61.8/47.8	66.3/48.0	48.0/39.5	53.9/33.9	69.8/61.1	28.1/18.8	55.9/42.3