Improving End-to-End Speech Processing by Efficient Text Data Utilization with Latent Synthesis

Jianqiao Lu[1][*][†], Wenyong Huang[2][*],
Nianzu Zheng[2], Xingshan Zeng[2], Yu Ting Yeung[2] & Xiao Chen[2]
[1]The University of Hong Kong   [2]Huawei Noah's Ark Lab
jqlu@cs.hku.hk, wenyong.huang@huawei.com

Abstract

Training a high performance end-to-end speech (E2E) processing model requires an enormous amount of labeled speech data, especially in the era of data-centric artificial intelligence. However, labeled speech data are usually scarcer and more expensive for collection, compared to textual data. We propose Latent Synthesis (LaSyn), an efficient textual data utilization framework for E2E speech processing models. We train a latent synthesizer to convert textual data into an intermediate latent representation of a pre-trained speech model. These pseudo acoustic representations of textual data augment acoustic data for model training. We evaluate LaSyn on low-resource automatic speech recognition (ASR) and spoken language understanding (SLU) tasks. For ASR, LaSyn improves an E2E baseline trained on LibriSpeech train-clean-100, with relative word error rate reductions over 22.3% on different test sets. For SLU, LaSyn improves our E2E baseline by absolute 4.1% for intent classification accuracy and 3.8% for slot filling SLU-F1 on SLURP, and absolute 4.49% and 2.25% for exact match (EM) and EM-Tree accuracies on STOP respectively. With fewer parameters, the results of LaSyn are competitive to published state-of-the-art works. The results demonstrate the quality of the augmented training data.

1 Introduction

In the data-centric artificial intelligence era, large quantity and high quality training data are essential for good performance of natural language processing (NLP) models including speech processing models. A conventional speech processing system is usually cascaded with an automatic speech recognition (ASR) module and an NLP module. For example, in spoken language understanding (SLU) which predicts semantic information from speech input, the system first transcribes input speech into

text with ASR, then pipes the text output to the natural language understanding (NLU) model for text analysis. An end-to-end (E2E) speech processing system leverages a single model which takes the input speech and performs spoken language processing tasks simultaneously. E2E models draw increasing attention due to less computational complexity and error propagation mitigation (Shen et al., 2021; Tian and Gorinski, 2020; Sharma et al., 2021; Lugosch et al., 2020; Wang et al., 2020; Chen et al., 2021b). However, a challenge of E2E model training is the collection of enormous annotated spoken data, which are significantly more expensive to collect compared with the text-only counterpart. In contrast, for a cascaded system, the ASR module and NLP module are trained separately with paired speech-transcription data and annotated textual data respectively. Separated types of data are usually more readily available and thus lower data collection costs. As the amount of high quality training data is critical for an E2E model, a strategy to alleviate the inadequate spoken data problem with more abundant textual data.

Two approaches have been proposed for utilizing textual data for E2E speech models in the literature. The first is modality conversion which utilizes a text-to-speech (TTS) system to convert text into speech (Laptev et al., 2020). The disadvantage is the requirement for a high-quality expressive TTS system. Another approach is unified representation learning for matching latent representations of speech and text with alignment losses (Bapna et al., 2021; Chen et al., 2022a). Given the significant difference between speech and text, aligning the hidden latent space of the two modalities is challenging.

We propose Latent Synthesis (LaSyn), a method to utilize text-only data for E2E speech processing models. LaSyn can be seen as an integration of the above two ideas. We train a latent synthesis model which synthesizes textual data into an intermediate

---

latent representation of a pre-trained speech model. Compared to modality conversion, speech latent representation contains fewer details and redundancy than the original speech signal, thus is easier to synthesize. Compared to unified representation learning, instead of aligning two modalities of huge difference, LaSyn learns to map the text into the latent representation of speech directly.

We evaluate LaSyn on low-resource ASR and SLU tasks. Low-resource ASR has gained big progress with the advancement of self-supervised speech pre-training (Baevski et al., 2020; Hsu et al., 2021; Huang et al., 2022). Further performance improvement still relies on external language models (Baevski et al., 2020). We show that LaSyn allows an E2E ASR model to utilize text-only data effectively without external language models, and outperforms ASR models with external language models. We further evaluate LaSyn on two publicly available datasets for SLU tasks, namely SLURP (Bastianelli et al., 2020) and Spoken Task Oriented Semantic Parsing (STOP) (Tomasello et al., 2022). LaSyn achieves comparable performance to the state-of-the-art (SOTA) SLU models but with significantly fewer model parameters. We summarize our contributions as follows:

- We propose LaSyn, an efficient textual data utilization framework for E2E speech processing models. The framework enables cross-modal knowledge transfer from text to E2E speech processing models through latent synthesis.

- We design 2 implementations for latent synthesizer which is the core of LaSyn framework: a fixed-projection latent synthesizer, and a diffusion latent synthesizer which applies recent progress of generative model, diffusion probabilistic model (Ho et al., 2020; Song et al., 2020).

- By improving an E2E ASR model through textual data utilization with LaSyn, we achieve competitive results on a low-resource ASR setup than published supervised ASR models which utilize textual data through an external language model.

- With LaSyn, we demonstrate E2E SLU models can be improved with a diverse set of textual NLP tasks, including NLU, information extraction (IE), named entity recognition (NER), and masked language modeling (MLM). We achieve competitive results to published SOTA works on two publicly available SLU datasets, with significantly fewer model parameters.

This paper is organized as follows. In the next section, we discuss related works of LaSyn. In Section 3, we discuss the model structure and training of LaSyn. We present experimental setup and results in Section 4, and ablation studies on SLU tasks in Section 5. Finally, we conclude our work in Section 6.

2 Related Works

In this section, we discuss the prior works of modality conversion and unified representation learning related to LaSyn.

Modality conversion: Laptev et al. (2020) shows that TTS data augmentation improves ASR performance in a low-resource setting. Sun et al. (2020) further shows that the diversity and quality of the TTS system are important for ASR data augmentation. Chen et al. (2022b) demonstrates similar representations derived from synthesized speech help downstream ASR tasks. Lugosch et al. (2020) confirms the effectiveness of speech synthesis for E2E SLU models, either as a sole source of training data or as a form of data augmentation. Thomas et al. (2021) utilizes artificially synthesized speech to adapt a SLU model based on a recurrent neural network transducer. Huang et al. (2020b) demonstrates the effectiveness of a multi-speaker TTS system under a low-resource SLU setting. Kharitonov et al. (2023) decouples the text-to-semantic and semantic-to-acoustic tasks to realize a multi-speaker text-to-speech system. LaSyn generates pseudo acoustic representations from text without requiring a vocoder for speech waveform generation.

Unified representation learning: Ao et al. (2021) extends the idea of T5 (Raffel et al., 2020) and proposes Speech-T5 with a cross-modal vector quantization in a shared discrete latent space. Kim et al. (2021) learns multi-modal alignment with two cross-modal pre-training tasks of masked language modeling and conditioned language modeling. Qian et al. (2021) unifies a pre-trained ASR encoder for speech and a pre-trained language model encoder for text into a transformer decoder. Sato et al. (2022) introduces an adaptation branch to embed acoustic and linguistic information in the same latent space. Thomas et al. (2022) trains an RNN-T model both on speech and text inputs. Zhang et al. (2022a) introduces two alternative discrete phoneme-unit and hidden-unit tokenizers to
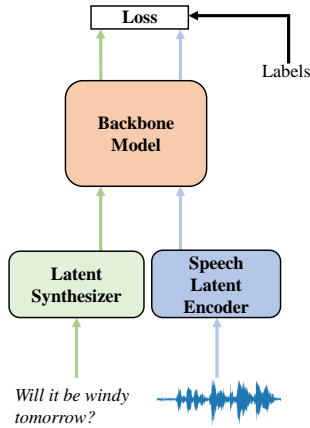
Figure 1: The architecture of LaSyn framework.

Figure 2: Speech Latent Encoder and Guiding Net from a pre-trained ASR model.

bridge speech and text modalities. MAESTRO (Chen et al., 2022a) learns unified representations of text and speech through sequence matching and duration prediction. Chung et al. (2018) attempts to align the individually learned text and speech embedding via adversarial training and a refinement procedure. SpeechUT (Zhang et al., 2022b) leverages hidden units as the bridge between the speech encoder and the text decoder. SpeechGPT(Zhang et al., 2023) applies modality-adaptation pertaining and cross-modal instruction fine-tuning to perceive and generate multi-model content. LaSyn connects text and speech information by mapping text representation directly into the pseudo acoustic latent space of a pre-trained speech model.

3 Method

3.1 Architecture

The LaSyn framework is illustrated in Fig. 1. The framework has 3 components: a speech latent encoder which maps speech data to corresponding speech latent representation, a latent synthesizer that projects text into the speech latent space, and a backbone model which is trained with either speech latent representations or pseudo acoustic latent representations from text.

3.2 Training procedure

3.2.1 Speech Latent Encoder

Speech latent encoder is obtained from a pre-trained speech processing model, which is a supervised ASR model as illustrated in Fig. 2 in this work. The parameters of speech latent encoder are frozen in the latter training stages to fix the speech latent space.
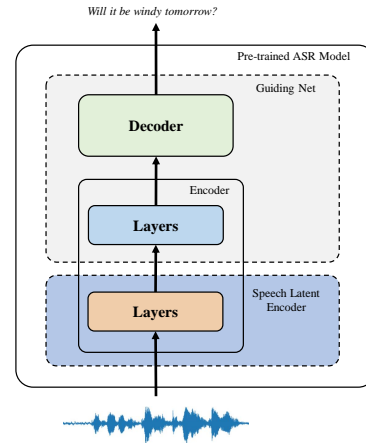
3.2.2 Latent Synthesizer

We then train a latent synthesizer to project textual data into the same speech latent space of the speech latent encoder. Latent synthesizer allows utilizing training samples from textual data, which is the core of the LaSyn framework. We explore two implementations of the latent synthesizer.
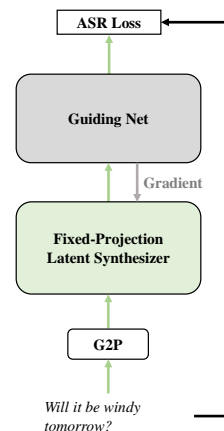

Figure 3: Training process of Fixed-Projection Latent Synthesizer. We freeze parameters of Guiding Net.

Fixed-projection Latent Synthesizer: We train a fixed-projection latent synthesizer with the help of a guiding net. The guiding net is also obtained from the pre-trained ASR model as illustrated in Fig. 2. Note that the guiding net is frozen in this stage. The training procedure is illustrated in Fig. 3. We optimize a fixed-projection latent synthesizer to generate latent representations which are recognizable as input of the guiding net. As the name suggests, the fixed-projection latent synthesizer learns a fixed one-to-one projection between

text data and speech latent representation. The training objective is defined as follows,

$$\underset{\phi}{\arg\min} \, \mathcal{L}_{ASR}\Big(G_\theta\big(P_\phi(\text{G2P}(t))\big), t \Big) \qquad (1)$$

where G_θ and P_ϕ represent the guiding network and the fixed-projection latent synthesizer respectively, ϕ represents the parameters of the latent synthesizer, t is the text input, and G2P is a grapheme-to-phoneme module. $\mathcal{L}_{ASR}$ is the same loss function of the pre-trained ASR model, such as transducer loss (Graves, 2012) or cross-entropy loss for attention-based encoder-decoder (AED) (Vaswani et al., 2017).

Diffusion Latent Synthesizer: We also experiment with diffusion probabilistic models (DPM) (Ho et al., 2020) as the latent synthesizer. DPMs have achieved great success in TTS (Popov et al., 2021; Chen et al., 2021a) and text-conditioned image synthesis (Nichol et al., 2021; Saharia et al., 2022) recently. We use the formulation of DPM proposed in Karras et al. (2022). Diffusion latent synthesizer generates latent representations by sampling an initial latent representation from a noise distribution and iteratively denoising the sample using a denoising model $D(h_{noisy}; \boldsymbol{e}, \sigma)$ where h_{noisy} represents the noisy latent at the current step, $\boldsymbol{e}$ denotes the conditional text. The denoising model is composed of an UNet (Ronneberger et al., 2015) and a text encoder as shown in Fig. 4. To reduce the complexity of the diffusion model, we train an autoencoder to compress the latent representation and use the lower-dimensional latent representation as the target of the diffusion latent synthesizer, similar to Rombach et al. (2022). For succinctness, we do not depict the training of autoencoder in Fig. 4. The training objective is to minimize,

$$\mathbb{E}_{p(h,\boldsymbol{e}),p(\boldsymbol{\epsilon}),p(\sigma)}\left[\lambda(\sigma)\big\|D(h + \sigma\boldsymbol{\epsilon}; \boldsymbol{e}, \sigma) - h\big\|_2^2\right] \quad (2)$$

where h is clean latent representation, $p(h, \boldsymbol{e})$ represents the training data distribution of latent-text pairs. The latent-text pairs are derived from a paired speech-text dataset and a speech latent encoder which converts the speeches into latent representations. $p(\sigma)$ is the distribution of noise levels that defines the corruption schedule (Karras et al., 2022). $p(\boldsymbol{\epsilon}) \in \mathcal{N}(\mathbf{0}, \mathbf{1})$ is the standard normal
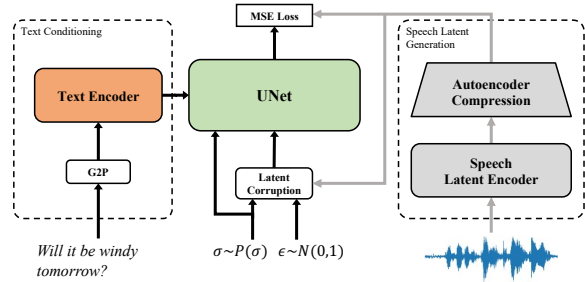

Figure 4: Diffusion Latent Synthesizer training. The gray color indicates that the Speech Latent Encoder and Autoencoder are frozen during training.

distribution, $\lambda(\sigma)$ is the weighting factor of noise levels. We employ classifier-free diffusion guidance (Ho and Salimans, 2022) to control latent quality and text alignment when sampling from the diffusion latent synthesizer.

3.2.3 Backbone Model and Dual-modality Training

After we train the latent synthesizer, we train the backbone model. We freeze the speech latent encoder and the latent synthesizer during backbone model training. We utilize both text and speech data in training. The backbone model takes input latent features from either speech latent encoder or latent synthesizer. We formulate both text-to-text and speech-to-text tasks as a unified sequence-to-sequence problem and refer to as dual-modality training. The training loss is specific to each task, i.e., transducer loss for ASR, and cross-entropy loss for SLU. The amount of textual data is usually significantly larger than speech data. We first train the backbone model with textual data. Then we train the backbone model with both text and speech data.

4 Experiments

4.1 Training Data

4.1.1 ASR

We apply a 100-hour subset (train-clean-100) of LibriSpeech (Panayotov et al., 2015) as low-resource labeled speech data. We use the transcription of the whole 960-hour LibriSpeech training split (LS-960) as text-only data.

4.1.2 SLU

We evaluate LaSyn on two challenging SLU datasets, SLURP (Bastianelli et al., 2020) and

Task	Dataset
NLU	CLINC150 (Larson et al., 2019)
	Redwood (Larson and Leach, 2022)
	GOOGLE-DSTC8 (Rastogi et al., 2020)
	Leyzer (Sowański and Janicki, 2020)
	HINT3 (Arora et al., 2020)
	Chatbot-Corpus (Braun et al., 2017)
	MultiWOZ (Zang et al., 2020)
	BANKING77 (Casanueva et al., 2020)
	FEWSHOTWOZ (Peng et al., 2020)
	ATIS (Tur et al., 2010)
	Schema (Rastogi et al., 2019)
NER	CrossNER (Liu et al., 2020)
	WNUT17 (Derczynski et al., 2017)
	CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003)
	CoNLL-2004 (Carreras and Màrquez, 2004)
IE	OntoNotes (Weischedel et al., 2013)
	SCIERC (Luan et al., 2018)

Table 1: Extra NLP datasets for SLU experiments.

Channel multiplier	$[1, 1, 1, 1]$
Dropout	0.1
Number of channels	256
Number of residual blocks	1
Self attention resolutions	$[4, 2]$

Table 2: Hyper-parameters of UNet model

STOP (Tomasello et al., 2022). SLURP is substantially larger and linguistically more diverse than previous SLU datasets. STOP is a recently released dataset that is the largest and the most complex SLU dataset. We also leverage a diverse set of NLP text datasets from different tasks, including natural language understanding (NLU), named entity recognition (NER), and information extraction (IE). The extra NLP text datasets are listed in Table 1.
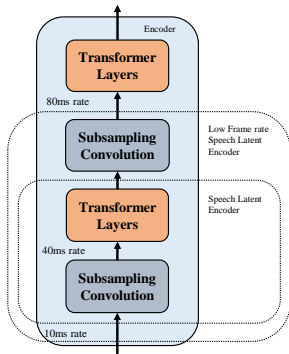

Figure 5: Encoder architecture of the ASR model. The frame rate of input is denoted as '10/40/80 ms'.
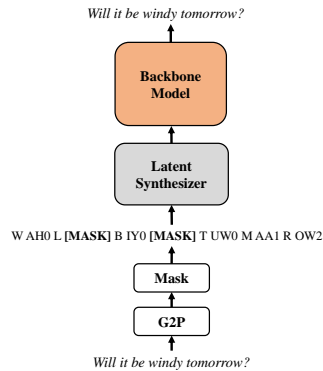

Figure 6: MLM task for utilizing unlabeled text data. [MASK] denotes the masked position.

4.2 Model and Training Setups

4.2.1 ASR

For ASR pre-training, we use a Transformer Transducer model (Tian et al., 2019; Yeh et al., 2019; Zhang et al., 2020). We apply a 128-dimensional log-mel filterbank with 20 ms window length and 10 ms frame rate as input acoustic feature. We interleave strided-convolutions in the encoder to gradually down-sample the input speech as illustrated in Fig. 5, which reduces computation effectively with negligible performance degradation (Peddinti et al., 2018; Han et al., 2020; Huang et al., 2020a). This model is pre-trained with train-clean-100. SpecAugment (Park et al., 2020) is applied to avoid overfitting. This pre-trained model is also our E2E ASR baseline. We obtain a speech latent encoder from this pre-trained model.

For latent synthesizers, we evaluate both fixed-projection latent synthesizer and diffusion latent synthesizer. The fixed-projection latent synthesizer is composed of 4 1-D convolutional layers of 512 filters with a kernel size of 5. We observe that a simple model structure is sufficient. We train the diffusion latent synthesizer with train-clean-100. The text encoder is composed of two convolution layers followed by the two-layer transformer. The number of channels is 256. The UNet model is adapted for 1-D sequence processing. The hyper-parameters of the UNet model are listed in Table 2. We use a small model such that the latent synthesizer generates the pseudo acoustic latent representations on the fly during dual-modality training.

The backbone model is the same as the guiding net in Fig. 2. To utilize textual data in dual-modality training of the backbone model, we design a task similar to masked language modeling

(MLM) ([Devlin et al., 2018](#)) as illustrated in Fig. 6. We randomly mask 30% of input phonemes converted by g2pE[1] according to CMUDict[2], and train the backbone model to predict the corresponding words.

We note that the parameters of the guiding net are frozen in latent synthesizer training. If we do not provide textual data for backbone model training, we just update the E2E baseline with extra epochs with a frozen speech latent encoder.

4.2.2 SLU

We apply an attention-based encoder-decoder model for ASR pre-training. The pre-trained ASR model is trained with LS-960 and SLURP speech data. The structure of the encoder is similar to the one in ASR experiments described in section [4.1.2](#). We apply a 6-layer, 256-dimensional Transformer as the decoder. We evaluate the two implementations of the latent synthesizer similar to ASR experiments. For fixed-projection latent synthesizer, the configuration is the same as ASR experiments. We apply text transcription of LS-960 for training. For diffusion latent synthesizer, we use LS-960 as paired speech-text training data. The backbone model shares the same model structure as the guiding net in Fig. [2](#). We also initialize the parameters from the guiding net. We train the backbone model with multiple tasks, including SLU, NLU, NER, and IE. We convert the annotation of all the datasets to a text-sequence format as illustrated in Fig. [7](#). We formulate all the tasks as a unified sequence-to-sequence problem.

We note that the model structure of the E2E baseline model is the same as the LaSyn model, but the latent synthesizer is disabled. The E2E baseline model does not train with any additional textual data. We fine-tune the E2E baseline model with SLU task after ASR pre-training.

4.3 ASR Results

The experimental results of ASR are shown in Table [3](#). We first compare LaSyn models with our E2E baseline which achieves comparable performance to conformer-based models. The only difference is that the LaSyn models are trained with additional textual data. The LaSyn-Diffusion model, which uses a diffusion latent synthesizer, achieves 40.5% and 22.3% relative WER reductions on test-clean and test-other of Librispeech test sets com-
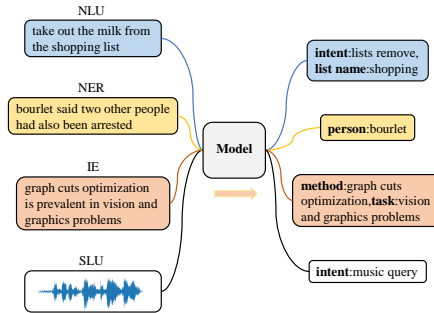
Figure 7: Dual-modality training for SLU with LaSyn. The output labels of different tasks are converted to text sequences as shown in the right blocks. Meta values such as slot type and entry type are in bold.

Model	LM	test	
		clean	other
Hybrid DNN/HMM ([Lüscher et al., 2019](#))	4-gram	5.8	18.6
LAS ([Park et al., 2020](#))	LSTM	5.5	16.9
Conformer-CTC ([Watanabe et al., 2022](#))	-	7.7	20.6
Conformer-CTC/Attention ([Watanabe et al., 2022](#))	-	7.3	19.3
Conformer-Transducer ([Watanabe et al., 2022](#))	-	7.8	19.8
TTS data augm. ([Laptev et al., 2020](#))	-	6.8	19.9
TTS data augm. ([Laptev et al., 2020](#))	LSTM	**4.3**	**13.5**
E2E baseline (ours)	-	7.4	20.1
LaSyn-FixedProj-LFR (ours)	-	4.5	17.1
LaSyn-FixedProj (ours)	-	4.5	16.1
LaSyn-Diffusion (ours)	-	**4.4**	**15.6**

Table 3: Low-resource ASR results trained with train-clean-100 split of LirbiSpeech. We compare LaSyn with published supervised methods. We report WER (%) on dev/test sets.

pared to the E2E baseline. We notice that the improvement on test-clean is more significant than test-other. Both the fixed-projection latent synthesizer and the diffusion latent synthesizer are trained with train-clean-100 which contains only clean speech. We speculate that the limited variety of training data train-clean-100 biases ASR performance toward clean speech.

We also observe that the performance of the model with fixed-projection latent synthesizer (LaSyn-FixedProj) is only slightly worse than LaSyn-Diffusion. The result is surprising, as the fixed-projection latent synthesizer is simpler than the diffusion latent synthesizer. The diffusion latent synthesizer may need further hyper-parameter tuning, or may need more training data for better performance. The LaSyn-FixedProj-LFR model utilizes a low frame rate speech latent encoder as illustrated in Fig. [5](#). The performance is slightly worse than the LaSyn-FixedProj on test-other.

Compared to published supervised ASR mod-

Model	# Params	IC (ACC %)	SF (SLU-F1)
ESPnet-SLU (Arora et al., 2022)	≥ 300 M	86.3	71.9
PF-hbt-base (Wang et al., 2021)	≥ 90 M	87.5	75.3
EF-hbt-large (Wang et al., 2021)	≥ 300 M	**89.4**	78.4
E2E Baseline (ours)	37.8 M	84.4	74.7
LaSyn-Diffusion (ours)	37.8 M	87.4	77.3
LaSyn-FixedProj (ours)	37.8 M	88.5	**78.5**

Table 4: Results on SLURP dataset. We report accuracy (ACC%) for the IC task and SLU-F1 for the SF task.

els that utilize text data through external language models, LaSyn models perform better without an external language model (LM). Compared to the published methods using TTS for data augmentation, the performance of LaSyn models are significantly better without an external LM. Given the existence of real-world scenarios with limited labeled speech data, such as minority languages and specific domains, our proposed method offers a novel approach to developing ASR applications.

4.4 SLU Results

4.4.1 SLURP

The experimental results of SLURP are shown in Table 4. We report accuracy for intent classification (IC), and SLU-F1(Bastianelli et al., 2020) for slot filling (SF).

We first compare LaSyn models with our E2E baseline. Compared to the E2E baseline, LaSyn-FixedProj improves IC accuracy and SF SLU-F1 by absolute 4.1% and 3.8% respectively. The result suggests that knowledge of textual NLP data is effectively transferred to SLU model. LaSyn-Diffusion performs slightly worse than LaSyn-FixedProj. We believe that with further hyperparameter tuning and more training data, the performance of diffusion latent synthesizer should be further improved.

We further compare the LaSyn models with previously published E2E SLU results. The published models are fine-tuned from HuBERT (Hsu et al., 2021) Base (95 M parameters) or Large (300 M parameters). The performance of LaSyn-FixedProj is comparable to ESPnet-SLU (Arora et al., 2022) and PF-hbt-base (Wang et al., 2021). The IC accuracy of LaSyn-FixedProj is slightly worse than EF-hbt-large (Wang et al., 2021), but the number of parameters is 8 times fewer.

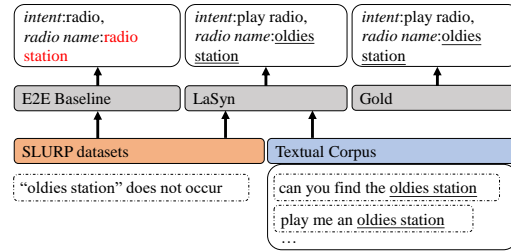To understand how LaSyn improves our baseline E2E SLU model, we further analyze samples from

Figure 8: An example of LaSyn output from SLURP test set. The target "oldies station" does not appear in SLU training data while LaSyn utilizes knowledge from the textual corpus. Meta values such as "intent" and slot type are *italicized*.

the test set that LaSyn performs better than our baseline. An example is shown in Fig. 8. Our E2E baseline model fails for the slot "Oldies Station", as this phrase never occurs in the SLURP training set. In contrast, LaSyn model correctly predicts the slot value. This phrase is included in the textual corpora. The text knowledge is transferred to SLU model with the LaSyn framework. The baseline E2E SLU model does not get the proprietary term 'Oldies Station' while LaSyn predicts this unique vocabulary successfully.

4.4.2 STOP

We present our results of STOP in Table 5. Compared to our E2E baseline, LaSyn-FixedProj improves EM accuracy and EM-Tree accuracy on the test set by absolute 4.49% and 2.25% respectively, again suggesting that there is effective cross-modality text knowledge transfer.

We further compare our results with STOP-E2E and STOP-Cascaded (Tomasello et al., 2022). STOP-E2E is an encoder-decoder based Transformer model fine-tuned from an E2E ASR model. The E2E ASR model is fine-tuned from HuBERT Base (Hsu et al., 2021). STOP-Cascaded is a cascaded system composed of an ASR system fine-tuned from wav2vec2.0 Base (Baevski et al., 2020) and an NLU model fine-tuned from a BART Base model (Lewis et al., 2019). LaSyn-FixedProj performs slightly better than STOP-E2E with 0.25% and 2.63% absolute improvement of EM and EM-Tree accuracies on the test set respectively. However, compared to STOP-Cascaded on the test set, while LaSyn-FixedProj is competitive on EM-Tree accuracy, EM accuracy is slightly inferior. The number of parameters in LaSyn models is much fewer. We expect performance improvement with more model parameters.

Model	# Params	dev EM / EM-Tree	test EM / EM-Tree
STOP-E2E (Tomasello et al., 2022)	≥ 90 M	69.12 / 83.89	69.23 / 82.87
STOP-Cascaded (Tomasello et al., 2022)	≥ 230 M	72.43 / 86.58	**72.36 / 85.77**
E2E Baseline (ours)	37.8 M	64.02 / 82.84	64.99 / 82.25
LaSyn-Diffusion (ours)	37.8 M	67.91 / 85.57	68.33 / 84.92
LaSyn-FixedProj (ours)	37.8 M	69.33 / 86.24	**69.48 / 85.50**

Table 5: Results on STOP dataset. We report the EM and EM-Tree accuracies (%) on dev and test sets.

Model	Text Data	SLURP (IC / SF)	STOP (dev) (EM / EM-Tree)	STOP (test) (EM / EM-Tree)
E2E Baseline	-	84.4 / 74.7	64.02 / 82.84	64.99 / 82.25
LaSyn-FixedProj	labelled	88.5 / 78.5	69.33 / 86.24	69.48 / 85.50
LaSyn-FixedProj	unlabelled	86.1 / 75.4	66.13 / 82.89	66.40 / 82.33

Table 6: Ablation study of unlabeled text data. We report results on SLURP test set, STOP dev and test sets.

5 Ablation Study

5.1 Training with Unlabeled Textual Data

Plain text data without annotation are more abundant than annotated NLP data. We experiment with SLU training with unlabelled textual data. We prepare the unlabelled text data by striping the annotation labels of the NLP datasets and keeping the input text. We apply the MLM task described in section 4.1.2 to utilize the unlabeled textual data. We evaluate LaSyn models with fixed-projection latent synthesizer. The results are listed in Table 6.

The results show that LaSyn still benefits from unlabeled text, compared to our E2E baseline on both SLURP and STOP datasets. With unlabeled text and MLM tasks, LaSyn achieves an absolute improvement of 1.6 % and 0.9 % on IC and SF tasks on SLURP dataset, 2.19 %, and 0.46% on EM and EM-Tree on STOP test set. While the improvement is not as significant as using labeled textual data, data collection is further simplified with unlabelled textual data.

5.2 Training with Diverse NLP Tasks

We do an ablation to observe the effect of training LaSyn with textual data from a diverse set of NLP tasks. The results are shown in Table 7. We observe that including each NLP task brings substantial improvement over the E2E baseline. As the NLU task is the most relevant to SLU, performance improvement is the most significant. When we combine all the NLP tasks, there is marginal further performance improvement.

Model	Text Training data	STOP (dev) (EM / EM-Tree)	STOP (test) (EM / EM-Tree)
E2E baseline	-	64.02 / 82.84	64.99 / 82.25
LaSyn	NLU	68.99 / 86.31	69.40 / 85.45
	NER	68.55 / 85.65	69.24 / 85.05
	IE	68.43 / 85.50	68.88 / 84.99
	NLU + NER + IE	69.33 / 86.24	69.48 / 85.50

Table 7: Results of LaSyn trained with text data of different NLP tasks. We report EM and EM-Tree accuracies (%) on STOP dev and test sets.

Model	SLURP (IC / SF)	STOP (dev) (EM / EM-Tree)	STOP (test) (EM / EM-Tree)
E2E Baseline	84.4 / 74.7	64.02 / 82.84	64.99 / 82.25
LaSyn (Acoustic Aug.)	86.9 / 76.0	67.69 / 85.18	68.25 / 84.50

Table 8: Results of acoustic augmentation with latent synthesizer. We report IC (ACC%) and SF (SLU-F1) for SLURP, EM and EM-Tree accuracies (%) for STOP.

5.3 Latent Synthesizer as Acoustic Augmentation

We experiment with using the fixed-projection latent synthesizer for acoustic augmentation. We extract the transcription and the annotation from the SLU dataset to form an NLU dataset. When training the backbone model, we apply both the SLU and the NLU datasets in dual-modality training. As the NLU dataset is derived from the SLU dataset, the latent synthesizer does not introduce extra textual content. Pseudo speech latent representations from the latent synthesizer are considered as an augmentation of the original speech latent representation.

As shown in Table 8, SLU performance improves significantly over the E2E baseline but does not reach the level of Table 7 which utilizes extra NLP datasets. Further enriching the diversity of pseudo acoustic latent is the potential to improve SLU performance.

6 Conclusion

We present LaSyn, a framework which enables efficient textual data utilization for E2E speech processing. By converting text into pseudo acoustic latent representation with a latent synthesizer, cross-modality knowledge transfer from textual data to E2E speech processing models is achieved. For the low-resource ASR task with Librispeech, LaSyn achieves relative WER reduction from 22.3% to 40.5% on test sets, compared to our E2E baseline with the same model structure. The results are competitive to published works which utilize textual data through external language models. For SLU

tasks, LaSyn improves over our E2E baseline by absolute 4.1% and 3.8% for for IC accuracy and SF SLU-F1 on SLURP, and absolute 4.49% and 2.25% of EM and EM-Tree accuracies on STOP. The results are competitive to published SOTA works with much fewer model parameters. Future improvement of latent synthesizer should further bridge the gap between speech and textual modality, which we leave as next step.

Limitations

The core of our method is the generation of pseudo acoustic representation from text input. We focus on generating consistent latent sequences effectively. We only evaluate two latent synthesis methods, including fixed-projection and diffusion latent synthesizers. There are other probable methods for latent generation, such as generative adversarial networks (GAN) (Goodfellow et al., 2020). Compared with TTS which generates audible speech suitable for human judgment, there is no subjective method to evaluate the quality and intelligibility of generated pseudo acoustic representation from the proposed framework, which is a main limitation. The design of reasonable quality indicators of acoustic representation would be meaningful for future work. Moreover, we have not evaluated the proposed latent synthesis framework on other phonological systems such as tonal languages like Chinese. The effectiveness of the framework on tonal languages is not guaranteed.

Ethics Statement

In this paper, we only use publicly available datasets for experiments. Our experiments do not involve any subjective tests or human data annotations. In the experiments, the latent synthesis framework does not produce any audible speech content. We do not apply any specific speaker information during training and inference.

References

Junyi Ao, Rui Wang, Long Zhou, Shujie Liu, Shuo Ren, Yu Wu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, et al. 2021. SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. arXiv preprint arXiv:2110.07205.

Gaurav Arora, Chirag Jain, Manas Chaturvedi, and Krupal Modi. 2020. HINT3: Raising the bar for intent detection in the wild. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, pages 100–105.

Siddhant Arora, Siddharth Dalmia, Pavel Denisov, Xuankai Chang, Yushi Ueda, Yifan Peng, Yuekai Zhang, Sujay Kumar, Karthik Ganesan, Brian Yan, et al. 2022. ESPnet-SLU: Advancing spoken language understanding through ESPnet. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2022)*, pages 7167–7171.

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

Ankur Bapna, Yu-an Chung, Nan Wu, Anmol Gulati, Ye Jia, Jonathan H Clark, Melvin Johnson, Jason Riesa, Alexis Conneau, and Yu Zhang. 2021. SLAM: A unified encoder for speech and language modeling via speech-text joint pre-training. arXiv preprint arXiv:2110.10329.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. Slurp: A spoken language understanding resource package. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7252–7262.

Daniel Braun, Adrian Hernandez-Mendez, Florian Matthes, and Manfred Langen. 2017. Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 174–185.

Xavier Carreras and Lluís Màrquez. 2004. Introduction to the CoNLL-2004 shared task: Semantic role labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 89–97.

Iñigo Casanueva, Tadas Temcinas, Daniela Gerz, Matthew Henderson, and Ivan Vulic. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on NLP for ConvAI - ACL 2020*.

Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, Najim Dehak, and William Chan. 2021a. WaveGrad 2: Iterative refinement for text-to-speech synthesis. arXiv preprint arXiv:2106.09660.

Yixin Chen, Weiyi Lu, Alejandro Mottini, Li Erran Li, Jasha Droppo, Zheng Du, and Belinda Zeng. 2021b. Top-down attention in end-to-end spoken language understanding. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021)*, pages 6199–6203.

Zhehuai Chen, Yu Zhang, Andrew Rosenberg, Bhuvana Ramabhadran, Pedro Moreno, Ankur Bapna, and Heiga Zen. 2022a. Maestro: Matched speech text representations through modality matching. *arXiv preprint arXiv:2204.03409*.

Zhehuai Chen, Yu Zhang, Andrew Rosenberg, Bhuvana Ramabhadran, Pedro Moreno, and Gary Wang. 2022b. Tts4pretrain 2.0: Advancing the use of text and speech in asr pretraining with consistency and contrastive losses. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7677–7681.

Yu-An Chung, Wei-Hung Weng, Schrasing Tong, and James Glass. 2018. Unsupervised cross-modal alignment of speech and text embedding spaces. *Advances in neural information processing systems*, 31.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. arXiv preprint arXiv:1211.3711.

Wei Han, Zhengdong Zhang, Yu Zhang, Jiahui Yu, Chung-Cheng Chiu, James Qin, Anmol Gulati, Ruoming Pang, and Yonghui Wu. 2020. ContextNet: Improving convolutional neural networks for automatic speech recognition with global context. In *Interspeech 2020*, pages 3610–3614.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851.

Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598.

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.

Wenyong Huang, Wenchao Hu, Yu Ting Yeung, and Xiao Chen. 2020a. Conv-Transformer Transducer: Low latency, low frame rate, streamable end-to-end speech recognition. In *Interspeech 2020*, pages 5001–5005.

Wenyong Huang, Zhenhe Zhang, Yu Ting Yeung, Xin Jiang, and Qun Liu. 2022. SPIRAL: Self-supervised perturbation-invariant representation learning for speech pre-training. In *International Conference on Learning Representations*.

Yinghui Huang, Hong-Kwang Kuo, Samuel Thomas, Zvi Kons, Kartik Audhkhasi, Brian Kingsbury, Ron Hoory, and Michael Picheny. 2020b. Leveraging unpaired text data for training end-to-end speech-to-intent systems. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 7984–7988.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. 2022. Elucidating the design space of diffusion-based generative models. arXiv preprint arXiv:2206.00364.

Eugene Kharitonov, Damien Vincent, Zalán Borsos, Raphaël Marinier, Sertan Girgin, Olivier Pietquin, Matt Sharifi, Marco Tagliasacchi, and Neil Zeghidour. 2023. Speak, read and prompt: High-fidelity text-to-speech with minimal supervision. *arXiv preprint arXiv:2302.03540*.

Minjeong Kim, Gyuwan Kim, Sang-Woo Lee, and Jung-Woo Ha. 2021. ST-BERT: Cross-modal language model pre-training for end-to-end spoken language understanding. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021)*, pages 7478–7482.

Aleksandr Laptev, Roman Korostik, Aleksey Svischev, Andrei Andrusenko, Ivan Medennikov, and Sergey Rybin. 2020. You do not need more data: Improving end-to-end speech recognition by text-to-speech data augmentation. arXiv preprint arXiv:2005.07157.

Stefan Larson and Kevin Leach. 2022. Redwood: Using collision detection to grow a large-scale intent classification dataset. arXiv preprint arXiv:2204.05483.

Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. 2019. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2020. CrossNER: Evaluating cross-domain named entity recognition. ArXiv preprint arXiv:2012.04373.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232.

Loren Lugosch, Brett H Meyer, Derek Nowrouzezahrai, and Mirco Ravanelli. 2020. Using speech synthesis to train end-to-end spoken language understanding models. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 8499–8503.

Christoph Lüscher, Eugen Beck, Kazuki Irie, Markus Kitza, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2019. RWTH asr systems for LibriSpeech: Hybrid vs Attention. In *Interspeech 2019*, pages 231–235.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.

D. S. Park, Y. Zhang, C. Chiu, Y. Chen, B. Li, W. Chan, Q. V. Le, and Y. Wu. 2020. SpecAugment on large scale datasets. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 6879–6883.

Daniel S. Park, Yu Zhang, Ye Jia, Wei Han, Chung-Cheng Chiu, Bo Li, Yonghui Wu, and Quoc V. Le. 2020. Improved noisy student training for automatic speech recognition. In *Interspeech 2020*, pages 2817–2821.

Vijayaditya Peddinti, Yiming Wang, Daniel Povey, and Sanjeev Khudanpur. 2018. Low latency acoustic modeling using temporal convolution and LSTMs. *IEEE Signal Processing Letters*, 25(3):373–377.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182, Online. Association for Computational Linguistics.

Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. 2021. Grad-TTS: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning (ICML)*, pages 8599–8608.

Yao Qian, Ximo Bianv, Yu Shi, Naoyuki Kanda, Leo Shen, Zhen Xiao, and Michael Zeng. 2021. Speech-language pre-training for end-to-end spoken language understanding. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021)*, pages 7458–7462.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. arXiv preprint arXiv:1909.05855.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Schema-guided dialogue state tracking task at dstc8. arXiv preprint arXiv:2002.01359.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. arXiv preprint arXiv:2205.11487.

Hiroaki Sato, Tomoyasu Komori, Takeshi Mishima, Yoshihiko Kawai, Takahiro Mochizuki, Shoei Sato, and Tetsuji Ogawa. 2022. Text-only domain adaptation based on intermediate ctc. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2022, pages 2208–2212.

Bidisha Sharma, Maulik Madhavi, and Haizhou Li. 2021. Leveraging acoustic and linguistic embeddings from pretrained speech and language models for intent classification. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021)*, pages 7498–7502.

Yilin Shen, Yen-Chang Hsu, Avik Ray, and Hongxia Jin. 2021. Enhancing the generalization for intent classification and out-of-domain detection in slu. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2443–2453.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-based generative modeling through

stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Marcin Sowański and Artur Janicki. 2020. Leyzer: A dataset for multilingual virtual assistants. In *International Conference on Text, Speech, and Dialogue*, pages 477–486.

Guangzhi Sun, Yu Zhang, Ron J Weiss, Yuan Cao, Heiga Zen, Andrew Rosenberg, Bhuvana Ramabhadran, and Yonghui Wu. 2020. Generating diverse and natural text-to-speech samples using a quantized fine-grained VAE and autoregressive prosody prior. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 6699–6703.

Samuel Thomas, Brian Kingsbury, George Saon, and Hong-Kwang J Kuo. 2022. Integrating text inputs for training and adapting rnn transducer asr models. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8127–8131. IEEE.

Samuel Thomas, Hong-Kwang J Kuo, George Saon, Zoltán Tüske, Brian Kingsbury, Gakuto Kurata, Zvi Kons, and Ron Hoory. 2021. Rnn transducer models for spoken language understanding. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7493–7497. IEEE.

Yusheng Tian and Philip John Gorinski. 2020. Improving end-to-end speech-to-intent classification with Reptile. In *Interspeech 2020*, pages 891–895.

Zhengkun Tian, Jiangyan Yi, Jianhua Tao, Ye Bai, and Zhengqi Wen. 2019. Self-attention transducers for end-to-end speech recognition. In *Interspeech 2019*, pages 4395–4399.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Paden Tomasello, Akshat Shrivastava, Daniel Lazar, Po-Chun Hsu, Duc Le, Adithya Sagar, Ali Elkahky, Jade Copet, Wei-Ning Hsu, Yossi Adi, Robin Algayres, Tu Ahn Nguyen, Emmanuel Dupoux, Luke Zettlemoyer, and Abdelrahman Mohamed. 2022. STOP: A dataset for spoken task oriented semantic parsing. arXiv preprint arXiv:2207.10643.

Gokhan Tur, Dilek Hakkani-Tür, and Larry Heck. 2010. What is left to be understood in ATIS? In *2010 IEEE Spoken Language Technology Workshop (SLT)*, pages 19–24.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Pengwei Wang, Liangchen Wei, Yong Cao, Jinghui Xie, and Zaiqing Nie. 2020. Large-scale unsupervised pre-training for end-to-end spoken language understanding. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 7999–8003.

Yingzhi Wang, Abdelmoumene Boumadane, and Abdelwahab Heba. 2021. A fine-tuned wav2vec 2.0/hubert benchmark for speech emotion recognition, speaker verification and spoken language understanding. arXiv preprint arXiv:2111.02735.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. Official Results of Conformer-based Models from ESPNET [online]. 2022.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 LDC2013T19. Linguistic Data Consortium.

Ching-Feng Yeh, Jay Mahadeokar, Kaustubh Kalgaonkar, Yongqiang Wang, Duc Le, Mahaveer Jain, Kjell Schubert, Christian Fuegen, and Michael L Seltzer. 2019. Transformer-Transducer: End-to-end speech recognition with self-attention. arXiv preprint arXiv:1910.12977.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. MultiWOZ 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI, ACL 2020*, pages 109–117.

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. *arXiv preprint arXiv:2305.11000*.

Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar. 2020. Transformer Transducer: A streamable speech recognition model with transformer encoders and RNN-T loss. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 7829–7833.

Ziqiang Zhang, Sanyuan Chen, Long Zhou, Yu Wu, Shuo Ren, Shujie Liu, Zhuoyuan Yao, Xun Gong, Lirong Dai, Jinyu Li, et al. 2022a. SpeechLM: Enhanced speech pre-training with unpaired textual data. arXiv preprint arXiv:2209.15329.

Ziqiang Zhang, Long Zhou, Junyi Ao, Shujie Liu, Lirong Dai, Jinyu Li, and Furu Wei. 2022b. SpeechUT: Bridging speech and text with hidden-unit for encoder-decoder based speech-text pre-training. In *Proceedings of the 2022 Conference on*