INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Integrating Pretrained ASR and LM to Perform Sequence Generation for Spoken Language Understanding

Siddhant Arora[1], *Hayato Futami*[2], *Yosuke Kashiwagi*[2], *Emiru Tsunoo*[2], *Brian Yan*[1], *Shinji Watanabe*[1]

[1]Carnegie Mellon University, U.S.A.
[2]Sony Group Corporation, Japan

siddhana@andrew.cmu.edu

Abstract

There has been an increased interest in the integration of pretrained speech recognition (ASR) and language models (LM) into the SLU framework. However, prior methods often struggle with a vocabulary mismatch between pretrained models, and LM cannot be directly utilized as they diverge from its NLU formulation. In this study, we propose a three-pass end-to-end (E2E) SLU system that effectively integrates ASR and LM subnetworks into the SLU formulation for sequence generation tasks. In the first pass, our architecture predicts ASR transcripts using the ASR subnetwork. This is followed by the LM subnetwork, which makes an initial SLU prediction. Finally, in the third pass, the deliberation subnetwork conditions on representations from the ASR and LM subnetworks to make the final prediction. Our proposed three-pass SLU system shows improved performance over cascaded and E2E SLU models on two benchmark SLU datasets, SLURP and SLUE, especially on acoustically challenging utterances.

Index Terms: spoken language understanding, pretrained language models, semi-supervised learning

1. Introduction

Spoken Language Understanding (SLU) involves extracting semantic meaning or linguistic structure from spoken utterances. It has a wide range of applications in commercial devices like chatbots and voice assistants [1, 2]. To advance research in this field, new tasks and benchmarks [3, 4] have been proposed. Recently, there has been interest in performing sequence labeling (SL) tasks, such as entity recognition [5, 6] and semantic parsing [3], directly from spoken utterances. SL systems aim to tag each token in an utterance to provide insights into its linguistic structure and semantic meaning [7]. When performing SL directly on spoken utterances, there is an additional complexity of recognizing the mention of the entity tag [8, 9].

Conventional SLU systems [10–12] employ a cascaded approach for sequence labeling, where an automatic speech recognition (ASR) system first recognizes the spoken words from the input audio and a natural language understanding (NLU) system then tags the words in the predicted text. These cascaded approaches can effectively utilize pretrained ASR and NLU systems. However, they suffer from error propagation [13] as errors in the ASR transcripts can adversely affect downstream SLU performance. As a result, these systems particularly struggle in noisy or adverse scenarios where we do not have a good ASR system. These scenarios are particularly common for spontaneous speech [6, 14], for instance, in spoken conversation scenarios. Consequently, in this work, we focus on end-to-end (E2E) SLU systems. E2E SLU systems [15, 16] aim to predict entity tags and mentions directly from speech. These E2E SLU systems can avoid the cascading of errors but cannot directly utilize strong acoustic and semantic representations from pretrained ASR systems and language models (LMs) due to a vocabulary mismatch between pretrained ASR and LMs [6].

Most SLU datasets have little labeled data since they are expensive and time-consuming to collect. To address this challenge, there has been a significant amount of interest in using pretrained ASR and LMs in E2E SLU architecture. Prior works [6, 15, 17–22] have explored either pretraining SLU models on larger ASR corpora or using the encoder of a pretrained ASR model as a frontend. Recently, compositional E2E SLU [8] models have been proposed which show better adaptability with pretrained ASR systems, but they are still ineffective at utilizing pretrained NLU models. Similar efforts [23–28] have been made to incorporate LMs within the SLU framework. One approach [25, 29] aligns the acoustic embedding of the SLU model with text embeddings produced by the LM encoder. Another approach [30] fuses the ASR and LM encoders using a linear layer called an interface. Other works [20, 31, 32] concatenate the acoustic embeddings and semantic embeddings generated from a pretrained LM to make the final prediction using a deliberation network [33, 34].

In this study, we draw inspiration from previous work on 2-pass SLU [31] models and introduce a formulation that integrates ASR and LM subnetwork inside the SLU framework using a probabilistic approach. In the first pass, the ASR subnetwork predicts the ASR transcript from the input speech, followed by the LM subnetwork, which inputs the ASR transcript to predict the SLU label sequence. In the third pass, our formulation incorporates the ASR encoder embedding, ASR decoder embedding, and pretrained LM decoder embedding using an encoder-decoder architecture. Thus, we achieve modular-based end-to-end SLU system by maintaining the ability of each pretrained subnetwork. This not only helps both the subnetworks to be trained using their own vocabulary but also leads to better generation ability of LM as it's closer to it's NLU formulation. Further, by conditioning on ASR and LM representation during deliberation, it can recover from errors in ASR transcript, helping to outperform existing cascaded systems.

The key contributions of our work are:

- Integrating ASR and LM networks inside SLU formulation using probabilistic approach.
- Proposing a novel 3-pass SLU model that can incorporate pretrained ASR and LM for the sequence generation task.
- Demonstrating that our proposed model can improve the SLU performance for named entity recognition on two benchmark datasets, namely SLURP [5] and SLUE-VoxPopuli [6].
- Conducting an analysis that reveals that our improvements are most pronounced on acoustically challenging utterances.

10.21437/Interspeech.2023-1962

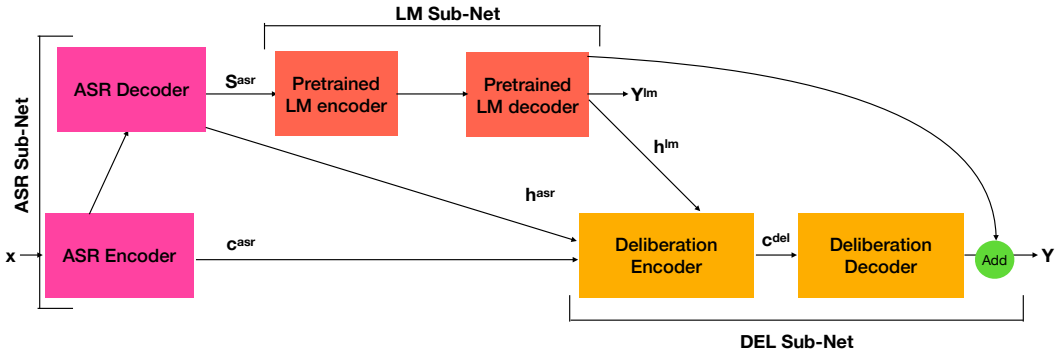

Figure 1: *Schematics of our three-pass SLU system that incorporates pretrained ASR and LM.*

2. Problem Formulation

The task of sequence labeling has been extensively studied in NLU systems, which take a text sequence S of length N and vocabulary $\mathcal{V}$ as input, represented as $S = \{s_n \in \mathcal{V} | n = 1, \ldots, N\}$. The model generates an output sequence $Y = \{y_o \in \mathcal{L} | o = 1, \ldots, O\}$, where $\mathcal{L}$ is the label set and O is the length of the output sequence. NLU systems, usually based on pretrained LMs, use the maximum a posteriori (MAP) theory to maximize the posterior distribution $P(Y|S)$.

In contrast, spoken language understanding (SLU) systems aim to generate the output sequence directly from the speech feature sequence $X = \{\mathbf{x}_t | t = 1, \ldots, T\}$ where T is the length of this sequence. SLU models use the MAP theory to output $\hat{Y}$ such that $\hat{Y} = \text{argmax}\ P(Y|X)$. We describe how we model this posterior distribution in Sec. 3 below.

3. Our three-pass SLU model

Our method aims to incorporate both pretrained ASR and LMs inside the SLU framework. We assume a pretrained ASR model that computes ASR transcript S^{asr} from spoken sequence X by maximizing $P(S|X)$ and pretrained LM that seeks to estimate label sequence Y^{lm} from text sequence S by maximizing $P(Y|S)$. By regarding the S^{asr} and Y^{lm} as a probabilistic variable, we can theoretically incorporate the ASR and LM modules into our formulation of the posterior distribution as shown:

$$P(Y|X) = \sum_{S^{\text{asr}}} \sum_{Y^{\text{lm}}} P(Y|X, S^{\text{asr}}, Y^{\text{lm}}) P(Y^{\text{lm}}|X, S^{\text{asr}}) P(S^{\text{asr}}|X) \tag{1}$$

We use Viterbi approximation to make the above computation tractable. We further assume the conditional independence of $Y^{\text{lm}}|S^{\text{asr}}$ from X to simplify Eq. 1:

$$P(Y|X) \approx \max_{S^{\text{asr}}} \max_{Y^{\text{lm}}} \underbrace{P(Y|X, S^{\text{asr}}, Y^{\text{lm}})}_{\text{DEL Sub-Net}} \underbrace{P(Y^{\text{lm}}|S^{\text{asr}})}_{\text{LM Sub-Net}} \underbrace{P(S^{\text{asr}}|X)}_{\text{ASR Sub-Net}} \tag{2}$$

To achieve the formulation described in Eq. 2, this work proposes a three-pass SLU architecture, which is illustrated in Figure 1. The architecture comprises three subnetworks: ASR, LM, and deliberation. In the first pass, the ASR subnetwork models $P(S^{\text{asr}}|X)$, as detailed in Sec. 3.1. The output of this pass, S^{asr}, serves as the input to the LM subnetwork in the second pass, which models $P(Y^{\text{lm}}|S^{\text{asr}})$ and generates Y^{lm}, as discussed in Sec. 3.2. Finally, in the third pass, the deliberation subnetwork uses the outputs of the first and second passes to model $P(Y|X, S^{\text{asr}}, Y^{\text{lm}})$ and make the final predictions, as explained in Sec. 3.3.

3.1. ASR subnetwork $P(S^{\text{asr}}|X)$

The input speech X is first passed through ASR encoder ($\text{Encoder}^{\text{asr}}$) to produce acoustic embeddings ($\mathbf{c}^{\text{asr}}$) as shown:

$$\mathbf{c}^{\text{asr}} = \text{Encoder}^{\text{asr}}(X) \tag{3}$$

This acoustic embedding is a sequence of latent representations $\mathbf{c}^{\text{asr}} = (\mathbf{c}_1^{\text{asr}}, \mathbf{c}_t^{\text{asr}}, .., \mathbf{c}_T^{\text{asr}})$. The ASR decoder ($\text{Decoder}^{\text{asr}}$) then maps the acoustic embedding $\mathbf{c}^{\text{asr}}$ and preceding ASR tokens generated by the ASR decoder $s_{1:n-1}^{\text{asr}}$ to ASR decoder hidden representations $\mathbf{h}_n^{\text{asr}}$.

$$\mathbf{h}_n^{\text{asr}} = \text{Decoder}^{\text{asr}}(\mathbf{c}_{1:T}^{\text{asr}}, s_{1:n-1}^{\text{asr}}) \tag{4}$$

The likelihood of each token in the ASR transcript is given by Out^{asr} which denotes a linear layer that maps decoder output $\mathbf{h}_n^{\text{asr}}$ to vocabulary $\mathcal{V}$ followed by softmax function.

$$P(s_n^{\text{asr}}|X, s_{1:n-1}^{\text{asr}}) = \text{Softmax}(\text{Out}^{\text{asr}}(\mathbf{h}_n^{\text{asr}})) \tag{5}$$

The likelihood of the entire ASR transcript is computed by composing the conditional probabilities at each step for N tokens.

$$P(S^{asr}|X) = \prod_{n=1}^{N} P(s_n^{\text{asr}}|X, s_{1:n-1}^{\text{asr}}) \tag{6}$$

3.2. LM subnetwork $P(Y^{\text{lm}}|S^{\text{asr}})$

The ASR transcript is then provided as input to the encoder of pretrained LM ($\text{Encoder}^{\text{lm}}$) as shown below.

$$\mathbf{c}^{\text{lm}} = \text{Encoder}^{\text{lm}}(S^{\text{asr}}) \tag{7}$$

Similar to the ASR decoder, the LM decoder ($\text{Decoder}^{\text{lm}}$) then maps the semantic embedding $\mathbf{c}^{\text{lm}}$ to hidden LM decoder representations $\mathbf{h}^{\text{lm}}$ as shown below:

$$\mathbf{h}_o^{\text{lm}} = \text{Decoder}^{\text{lm}}(\mathbf{c}_{1:N}^{\text{lm}}, y_{1:o-1}^{\text{lm}}) \tag{8}$$

The likelihood of the entire SLU label sequence is computed similarly to Eq. 6 as shown below

$$P(y_o^{\text{lm}}|S^{\text{asr}}, y_{1:o-1}^{\text{lm}}) = \text{Softmax}(\text{Out}^{\text{lm}}(\mathbf{h}_o^{\text{lm}})) \tag{9}$$

$$P(Y^{\text{lm}}|S^{\text{asr}}) = \prod_{o=1}^{O} P(y_o^{\text{lm}}|S^{\text{asr}}, y_{1:o-1}^{\text{lm}}) \tag{10}$$

3.3. Deliberation subnetwork $P(Y|X, S^{\text{asr}}, Y^{\text{lm}})$

Finally, we formulate $P(Y|X, S^{\text{asr}}, Y^{\text{lm}})$ by conditioning on X using ASR encoder representation $\mathbf{c}^{\text{asr}}$ (Eq. 3), S^{asr} using ASR decoder representations $\mathbf{h}^{\text{asr}}$ (Eq. 4) and Y^{lm} using LM decoder representations $\mathbf{h}^{\text{lm}}$ (Eq. 8) in the following two ways.

3.3.1. Concatenation Integration

We ensure that $\mathbf{h}^{\text{lm}}$ has the same embedding dimension as acoustic embedding $\mathbf{c}^{\text{asr}}$ by passing it through a linear layer. The ASR encoder, ASR decoder, and LM decoder representations are concatenatenated to produce a sequence of length K where $K = T+N+O$. This sequence is fed as input to a deliberation encoder $\text{Encoder}^{\text{del}}$ to produce joint embedding ($\mathbf{c}^{\text{del}}$):

$$\mathbf{c}^{\text{del}} = \text{Encoder}^{\text{del}}(\mathbf{c}^{\text{asr}}||\mathbf{h}^{\text{asr}}||\mathbf{h}^{\text{lm}}) \quad (11)$$

Similar to Eq. 9, the deliberation decoder (Decoder) maps this joint embedding $\mathbf{c}^{\text{del}}$ and previous tokens output by the decoder $y_{1:o-1}$ to generate SLU output sequence Y.

$$\mathbf{h}_o = \text{Decoder}(\mathbf{c}^{\text{del}}_{1:K}, y_{1:o-1}) \quad (12)$$

$$P(y_o|X, S^{\text{asr}}, Y^{\text{lm}}, y_{1:o-1}) = \text{Softmax}(\text{Out}(\mathbf{h}_o)) \quad (13)$$

Our architecture further adds a residual connection to posteriors obtained from pretrained LM (Eq. 9) and performs a weighted combination of the 2 posteriors to make the final prediction.

$$\hat{h}_o = (\alpha * \text{Out}(\mathbf{h}_o) + (1-\alpha) * \text{Out}^{\text{lm}}(\mathbf{h}^{lm}_o)) \quad (14)$$

$$P(y_o|X, S^{\text{asr}}, Y^{\text{lm}}, Y_{1:o-1}) = \text{Softmax}(\hat{h}_o) \quad (15)$$

where α is a learnable parameter. Note that 3-pass SLU model is trained using LM vocabulary to facilitate this combination.

3.3.2. Cross Attention Integration

Inspired by prior work on compositional speech processing models [8, 35], we also experiment with incorporating acoustic embedding $\mathbf{c}^{\text{asr}}$ using the multi-sequence cross attention in our deliberation decoder. In the formulation, the ASR decoder and LM representations are concatenated together and then a joint embedding is produced by the deliberation encoder:

$$\mathbf{c}^{\text{del}} = \text{Encoder}^{\text{del}}(\mathbf{h}^{\text{asr}}||\mathbf{h}^{\text{lm}}) \quad (16)$$

The deliberation decoder then conditions on acoustic embedding $\mathbf{c}^{\text{asr}}$ and joint embedding $\mathbf{c}^{\text{del}}$ to make the final prediction:

$$\mathbf{h}_o = \text{Decoder}(\mathbf{c}^{\text{del}}, \mathbf{c}^{\text{asr}}, y^{1:o-1}) \quad (17)$$

The final prediction is similarly made by using a weighted combination with LM posteriors as shown in Eq. 14.

3.4. Training and Inference

Similar to prior work on 2-pass SLU systems [31], we use a three step training process for our SLU system. In the first step, only the ASR subnetwork is trained using Eq. 6. In the second step, we train both the LM encoder and decoder with ground truth ASR transcript using Eq. 10. In the third step, the deliberation encoder ($\text{Encoder}^{\text{del}}$) and decoder (Decoder) are trained using Eq. 15 while using the pretrained ASR and LM subnetwork from step 1 and step 2. We also tried fine-tuning the entire network in step 3 but our initial experiments did not show any significant performance gains. We use teacher forcing on LM and deliberation decoder and also experiment with and without teacher forcing of ASR transcripts during step 3 in Sec. 4.4.1.

Our decoding process based on Eq. 2 also consists of three steps. In the first step, the ASR subnetwork generates the ASR transcript by maximizing $P(S^{\text{asr}}|X)$ (Eq. 6). In the second step, the LM subnetwork obtains the initial SLU predictions Y^{lm} by maximizing $P(Y^{\text{lm}}|S^{\text{asr}})$ (Eq. 10). The deliberation decoder then generates the final SLU predictions, Y, by maximizing $P(Y|X, S^{\text{asr}}, Y^{\text{lm}})$ (Eq. 15). Beam search is used at all 3 steps of inference which also helps improve ASR ($\mathbf{h}^{\text{asr}}$) and LM ($\mathbf{h}^{\text{lm}}$) decoder representations.

4. Experiments

4.1. Datasets

To demonstrate the effectiveness of our 3-pass SLU model, we conducted experiments on two publicly available spoken named entity recognition datasets, namely SLURP [5] and SLUE-VoxPopuli [6]. The SLURP corpus consists of single-turn user conversations with a home assistant, making it more relevant to commercial SLU applications. To ensure consistency with prior work [5, 15], we augment our training set with synthetic data. The corpus comprises 40.2 hours of audio, with 11,514 utterances in train set, 6.9 hours with 2,033 utterances in development set, and 10.3 hours with 2,974 utterances in test set.

The recently released SLUE benchmark is a publicly available, naturally produced speech dataset that contrasts with the read speech data used in most SLU benchmarks. We are specifically interested in the SLUE-VoxPopuli dataset [6], which contains NER annotations and consists of European Parliament recordings. The corpus comprises 5,000 utterances, with 14.5 hours of audio in the train set and 1,753 utterances, containing nearly 5 hours of audio in the development set. The test sets released for SLUE are blind, without ground truth labels; therefore, we compare different methods using the development set.

We report performance on SLURP using the SLU F1 [5] which weights the match of an entity label with the character and word error rate of the entity mention. SLUE is evaluated using F1 which exactly matches both the entity label and entity mention. We report micro-averaged F1 for all our experiments.

4.2. Baseline

We compare our proposed 3-pass SLU model with cascaded and other E2E SLU systems. We also train Cascaded SLU model with BART-large as NLU which is essentially the same as the first 2 passes of our 3 pass SLU model. We also report performance of a compositional E2E SLU model that predicts the SLU label sequence using a decoder (referred to as "Compositional E2E SLU with Direct E2E formulation" in [8][1]). We also compare our approach with the 2-pass SLU [31] model, where the second pass model can utilize semantic representations from a pretrained LM, along with acoustic representations.

4.3. Experimental Setups

Our models are implemented in pytorch [36] and the experiments are conducted through the ESPNet-SLU [15] toolkit. Following the approach in [8], we use a 12-layer conformer block for the encoder and a 6-layer transformer block for the decoder in our ASR model for both datasets. Each attention block consists of 8 attention heads, a dropout rate of 0.1, an output dimension of 512, and a feedforward dimension of 2048 for the SLURP dataset. For the SLUE dataset, each attention block has 4 attention heads, a dropout rate of 0.1, an output dimension of 256 for the encoder and 2048 for the decoder, and a feedforward dimension of 1024 for the encoder and 2048 for the decoder. Our ASR models achieved a word error rate (WER) of 30.4 for the SLUE dataset and 16.3 for the SLURP dataset. We train our ASR models to generate BPE tokens with a size of 500 for the SLURP dataset and 1000 for the SLUE dataset.

We use BART large [37] as our pretrained LM and train it with the HuggingFace Seq2Seq Trainer [38]. We train the models with a learning rate of 1e-5 and 5k warmup steps. Our 3-pass

[1]Note that we did not compare with token classification models, as our focus was on sequence generation for SLU tasks.

Model	SLURP SLU F1 ↑	SLUE F1 ↑
Casacaded SLU [8]	73.3	48.6
Casacaded SLU with BART (Ours)	77.9	55.8
Direct E2E SLU [8]	77.1	54.7
Compositional E2E SLU [8]	77.2	50.0
2-pass E2E SLU [31]	77.4	52.2
3-pass SLU		
w/ Concatenation §3.3.1	78.4	56.7
w/ Cross attention §3.3.2	**78.5**	**57.2**

Table 1: *Results presenting performance of 3-pass SLU system.*

Model	SLURP SLU F1 ↑	SLUE F1 ↑
3-pass SLU w/ Cross attention	78.5	57.2
w/o teacher forcing	78.4	54.1
w/o h^{asr} in Eq. 16	78.4	56.7
w/o c^{asr} in in Eq. 17	78.4	56.6
w/o h^{lm} in Eq. 16	74.5	56.3
w/o Residual connection in Eq. 15	77.8	50.9

Table 2: *Results presenting ablation of our 3-pass SLU system.*

SLU architecture uses the above-described ASR and LM as the ASR and LM subnetworks, respectively. For our 3-pass SLU architecture with concatenation integration, our deliberation encoder consists of a 4-layer conformer block with a feedforward dimension of 2048 for both datasets. For our 3-pass SLU architecture with cross attention integration, our deliberation encoder consists of a 6-layer transformer block with a feedforward dimension of 2048 for both datasets. The attention heads, output dimension, and other parameters of the deliberation encoder are the same as the ASR encoder of the respective datasets. The third-pass decoder for both datasets has the same architecture as the ASR decoder giving a total parameter size of 618 M for SLURP and 482 M for SLUE dataset. We train our 3-pass SLU models to generate BART large vocabulary tokens to facilitate combination with posteriors from the LM subnetwork (Eq. 14).

We also train 2-pass SLU model [31] using the same encoder and decoder as our ASR model and the same deliberation encoder as our 3-pass SLU model w/ concatenation integration. We perform SpecAugment [39] for data augmentation and apply dropout [40] and label smoothing [41]. Models were trained using 4 NVIDIA A40 GPUs. All model, training and inference parameters were selected based on validation performance.

4.4. Results and Discussion

Table 1 shows that our proposed 3-pass SLU model outperforms both cascaded and E2E SLU models on both datasets. It also performs significantly better than compositional E2E SLU models. Our 3-pass SLU models are a more effective way of utilizing pretrained LMs than the previously proposed 2-pass SLU model. Our experiments show that cross attention integration (Eq. 17) is a better way of conditioning on ASR and LM output than concatenation integration (Eq. 11). Our performance gains over cascaded systems are more significant for the SLUE dataset, which consists of natural conversation speech instead of read speech and is therefore more acoustically challenging. Hence our approach of E2E integration of ASR and LM subnetworks helps avoid cascading errors from the ASR transcript.

4.4.1. Ablation study

We conduct ablation experiments on our 3-pass SLU model to gain a better understanding of the relative importance of different formulation decisions. Specifically, we first perform an ablation study in which we used ASR hypotheses instead of ground truth transcripts during the third step of training (see Sec. 3.4). Table 2 demonstrates that teacher forcing enhances the SLU performance on both datasets. The impact of teacher forcing is more substantial on the SLUE dataset since it has a higher WER, and errors in ASR transcripts can pose a challenge to the model's training. We also experiment with masking out representations of the ASR encoder c^{asr} and ASR decoder h^{asr} from Eq.17 and Eq.16, respectively, and observe a slight drop in performance. Masking out the LM decoder representation h^{lm} from Eq.16 has a more significant negative impact, highlighting its importance. The exclusion of a residual connection (Eq. 15) to posteriors from a pretrained LM also results in significant performance degradation. We conclude that our formulation decisions as described in Sec. 3 achieve best performance.

WER	Cascaded SLU SLU F1 ↑	Cascaded SLU Label F1 ↑	3 pass SLU SLU F1 ↑	3 pass SLU Label F1 ↑	# Utterances
0.0	95.8	96.6	95.7	96.5	7164
(0.0,0.25]	75.2	85.7	76.3	86.3	2847
(0.25,1.0]	51.1	62.1	51.9	62.7	3067

Table 3: *Results comparing performance of Cascaded model using BART large & 3-pass model with cross attention integration across different ASR difficulties on SLURP dataset.*

4.4.2. Performance based on ASR WER

To assess the efficacy of our proposed 3-pass SLU model, we compare its performance with that of a cascaded model across utterances with varying levels of acoustic complexity on SLURP dataset. As in prior work [17], we also divide our test set into groups based on acoustic complexity, as quantified by the WER of ASR transcripts, with each group containing a similar number of utterances. Our findings in Table 3 indicate that both models perform similar when the WER is perfect, and that most of the performance gains are observed for utterances with errors in the ASR transcript. Further the maximum gain in performance is seen for utterances with WER between 0 and 0.25.

We also evaluate using Label-F1 metric, which only considers the matches of entity labels. The results show that improvement is smaller for Label-F1, suggesting that a part of the improvement is due to correctly identifying entity mentions, even when there are errors in the ASR transcript.

5. Conclusion

We present a novel 3-pass SLU model that integrates ASR and LM sub-networks into the E2E SLU formulation. Our study demonstrates that our model outperforms both cascaded and other E2E SLU models on sequence generation tasks, such as named entity recognition. We conduct an ablation study to evaluate the relative contributions of our design choices. Additionally, we observe that the majority of performance improvements from our model are seen on utterances with inaccurate ASR output. In future work, we plan to extend our proposed architecture to other speech processing tasks, such as speech translation.

6. Acknowledgements

This work used PSC Bridges2 and NCSA Delta through allocation CIS210014 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

7. References

[1] D. Yu, M. Cohn, *et al.*, "Gunrock: A social bot for complex and engaging long conversations," in *Proc. EMNLP-IJCNLP - System Demonstrations*, 2019.

[2] A. Coucke *et al.*, "Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces," *CoRR*, vol. abs/1805.10190, 2018.

[3] P. Tomasello *et al.*, "Stop: A dataset for spoken task oriented semantic parsing," in *Proc. SLT*, 2022, pp. 991–998.

[4] S. Shon *et al.*, "SLUE phase-2: A benchmark suite of diverse spoken language understanding tasks," *CoRR*, vol. abs/2212.10525, 2022.

[5] E. Bastianelli *et al.*, "SLURP: A spoken language understanding resource package," in *Proc. EMNLP*, 2020.

[6] S. Shon *et al.*, "SLUE: new benchmark tasks for spoken language understanding evaluation on natural speech," in *Proc. ICASSP*, 2022, pp. 7927–7931.

[7] D. Jurafsky and J. H. Martin, *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition, 2nd Edition* (Prentice Hall series in artificial intelligence). 2009.

[8] S. Arora *et al.*, "Token-level sequence labeling for spoken language understanding using compositional end-to-end models," in *Findings of the EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds., 2022, pp. 5419–5429.

[9] L. Zhai *et al.*, "Using n-best lists for named entity recognition from chinese speech," in *Proceedings of HLT-NAACL 2004: Short Papers, Boston, Massachusetts, USA, May 2-7, 2004*, 2004.

[10] D. D. Palmer and M. Ostendorf, "Improving information extraction by modeling errors in speech recognizer output," in *Proceedings of the First International Conference on Human Language Technology Research*, 2001.

[11] J. Horlock and S. King, "Discriminative methods for improving named entity extraction on speech data," in *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, 2003, pp. 2765–2768.

[12] F. Béchet *et al.*, "Detecting and extracting named entities from spontaneous speech in a mixed-initiative spoken dialogue context: How may I help you?$^{\text{sm, tm}}$," *Speech Commun.*, vol. 42, no. 2, pp. 207–225, 2004.

[13] T. Tran *et al.*, "Parsing speech: A neural approach to integrating lexical and acoustic-prosodic information," in *Proc. NAACL*, M. A. Walker, H. Ji, and A. Stent, Eds., 2018, pp. 69–81.

[14] J. Godfrey, E. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Proc. ICASSP*, vol. 1, 1992, 517–520 vol.1.

[15] S. Arora *et al.*, "ESPnet-SLU: Advancing spoken language understanding through espnet," in *Proc. ICASSP*, 2022, pp. 7167–7171.

[16] S. Ghannay *et al.*, "End-to-end named entity and semantic concept extraction from speech," in *Proc. SLT*, 2018, pp. 692–699.

[17] Y. Peng *et al.*, "A study on the integration of pre-trained ssl, asr, lm and slu models for spoken language understanding," in *Proc. SLT*, 2023, pp. 406–413.

[18] L. Lugosch *et al.*, "Speech model pre-training for end-to-end spoken language understanding," in *Proc. Interspeech*, G. Kubin and Z. Kacic, Eds., 2019, pp. 814–818.

[19] P. Wang *et al.*, "Large-scale unsupervised pre-training for end-to-end spoken language understanding," in *Proc. ICASSP*, 2020, pp. 7999–8003.

[20] C.-I. Lai *et al.*, "Semi-supervised spoken language understanding via self-supervised speech and language model pretraining," in *Proc. ICASSP*, 2021, pp. 7468–7472.

[21] A. Pasad *et al.*, "On the use of external data for spoken named entity recognition," in *Proc. NAACL*, M. Carpuat, M. de Marneffe, and I. V. M. Ruiz, Eds., 2022, pp. 724–737.

[22] E. Morais *et al.*, "End-to-end spoken language understanding using transformer networks and self-supervised pre-trained features," in *Proc. ICASSP*, 2021, pp. 7483–7487.

[23] B. Agrawal *et al.*, "Tie your embeddings down: Cross-modal latent spaces for end-to-end spoken language understanding," in *Proc. ICASSP*, 2022, pp. 7157–7161.

[24] Y.-S. Chuang *et al.*, "SpeechBERT: An Audio-and-text Jointly Learned Language Model for End-to-end Spoken Question Answering," in *Proc. Interspeech*, 2021.

[25] Y. Chung, C. Zhu, and M. Zeng, "SPLAT: speech-language joint pre-training for spoken language understanding," in *Proc. NAACL*, K. Toutanova *et al.*, Eds., 2021, pp. 1897–1907.

[26] M. Kim *et al.*, "St-bert: Cross-modal language model pre-training for end-to-end spoken language understanding," in *Proc. ICASSP*, 2021, pp. 7478–7482.

[27] Z. Zhang *et al.*, "A joint learning framework with bert for spoken language understanding," *IEEE Access*, vol. 7, pp. 168 849–168 858, 2019.

[28] C.-J. Hsu *et al.*, *T5lephone: Bridging speech and text self-supervised models for spoken language understanding via phoneme level t5*, 2022.

[29] Y. Huang *et al.*, "Leveraging unpaired text data for training end-to-end speech-to-intent systems," in *Proc. ICASSP*, 2020, pp. 7984–7988.

[30] S. Seo, D. Kwak, and B. Lee, "Integration of pre-trained networks with continuous token interface for end-to-end spoken language understanding," in *Proc. ICASSP*, 2022, pp. 7152–7156.

[31] S. Arora *et al.*, "Two-pass low latency end-to-end spoken language understanding," in *Proc. Interspeech*, H. Ko and J. H. L. Hansen, Eds., 2022, pp. 3478–3482.

[32] D. Le *et al.*, "Deliberation model for on-device spoken language understanding," in *Proc. Interspeech*, 2022, pp. 3468–3472.

[33] Y. Xia *et al.*, "Deliberation networks: Sequence generation beyond one-pass decoding," *Proc. NeurIPS*, pp. 1784–1794, 2017.

[34] K. Hu *et al.*, "Deliberation model based two-pass end-to-end speech recognition," in *Proc. ICASSP*, 2020, pp. 7799–7803.

[35] S. Dalmia *et al.*, "Searchable hidden intermediates for end-to-end models of decomposable sequence tasks," in *Proc. NAACL*, 2021, pp. 1882–1896.

[36] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Proc. NeurIPS*, vol. 32, 2019.

[37] M. Lewis *et al.*, "BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. ACL*, D. Jurafsky *et al.*, Eds., 2020, pp. 7871–7880.

[38] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Proc. EMNLP: System Demonstrations*, 2020, pp. 38–45.

[39] D. S. Park *et al.*, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[40] N. Srivastava *et al.*, "Dropout: A simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[41] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" *Advances in neural information processing systems*, vol. 32, 2019.