Contrastive and Consistency Learning for Neural Noisy-Channel Model in Spoken Language Understanding

Suyoung Kim, Jiyeon Hwang, Ho-Young Jung
Department of AI, Kyungpook National University
{swimai, exo1926, hoyjung}@knu.ac.kr

Abstract

Recently, deep end-to-end learning has been studied for intent classification in Spoken Language Understanding (SLU). However, end-to-end models require a large amount of speech data with intent labels, and highly optimized models are generally sensitive to the inconsistency between the training and evaluation conditions. Therefore, a natural language understanding approach based on Automatic Speech Recognition (ASR) remains attractive because it can utilize a pre-trained general language model and adapt to the mismatch of the speech input environment. Using this module-based approach, we improve a noisy-channel model to handle transcription inconsistencies caused by ASR errors. We propose a two-stage method, *Contrastive and Consistency Learning (CCL)*, that correlates error patterns between clean and noisy ASR transcripts and emphasizes the consistency of the latent features of the two transcripts. Experiments on four benchmark datasets show that CCL outperforms existing methods and improves the ASR robustness in various noisy environments. Code is available at https://github.com/syoung7388/CCL

1 Introduction

Understanding a conversation is critical for reaching a user's goal in a spoken dialogue system. Spoken Language Understanding (SLU) has been investigated by applying an end-to-end approach with a singular speech input model for training. The end-to-end approach requires an extensive volume of labeled speech data, which can be challenging to collect. As dialogue systems are commonly used in many real-world scenarios, the end-to-end SLU models are susceptible to unseen instances that mismatch the training data due to limited language expression and acoustic conditions. Utilizing a module-based approach, which leverages the capabilities of pre-trained language models, is an appealing option for developing general spoken dialogue systems. However, this option depends on the premise that the model pre-trained on clean text can effectively handle errors caused by the Automatic Speech Recognition (ASR) module. Consequently, it is necessary to solve the problem of inconsistencies in noisy ASR transcripts that contain speech recognition errors.

Recent studies have addressed the challenge of discrepancies between clean and noisy ASR transcripts. Ruan et al. (2020) minimized the Kullback-Leibler (KL) divergence so that the prediction distributions of both transcripts become similar in a noisy environment. In contrast, Chang and Chen (2022) demonstrated a technique that utilizes a contrastive learning to align the implicit features of paired clean and noisy ASR transcripts. However, most previous studies still need to consider the alignment for fine-grained errors such as insertion, deletion, and substitution caused by ASR modules.

We expect that using an improved noisy-channel model will mitigate the sensitivity of the pre-trained language model to discrepancies between clean and noisy ASR transcripts. The noisy-channel approach aims to identify the target word, even if the input becomes scrambled or distorted. We present the neural noisy-channel model to predict error-free intent by (i) finding the error portions of given noisy input relative to clean transcript and (ii) matching the latent representations from two transcripts. In this paper, we introduce *Contrastive and Consistency Learning (CCL)* to perform this two-stage.

As shown in Figure 1 (a), reference and inference networks receive clean and noisy ASR transcripts as inputs, respectively. During the CCL, two networks simultaneously perform a token-based contrastive learning followed by a consistency learning. The purpose of token-based contrastive learning is to correlate errors in the noisy ASR transcript with the corresponding clean transcript at both word and utterance token-levels. We select the word tokens

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 5698–5711
June 16-21, 2024 ©2024 Association for Computational Linguistics

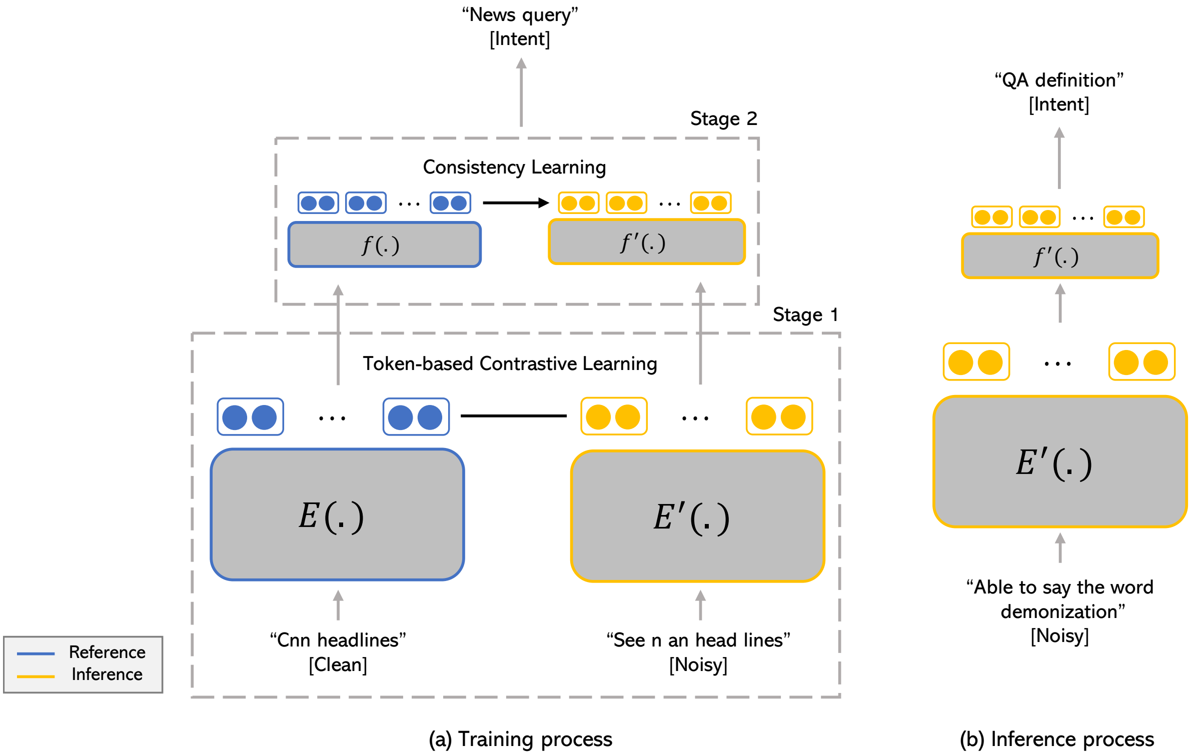


Figure 1: An overview of CCL. (a) Given a noisy ASR transcript, the inference network correlates error portions through token-based contrastive learning and then maintains coherence with the latent feature of the clean transcript through consistency learning. (b) In evaluation, inference network takes only noisy ASR transcript as input.

as a positive pair by aligning them with an edit distance algorithm when two transcriptions do not match sequentially. In addition, consistency learning makes the inference network emphasize the coherence between clean and noisy latent features to avoid misclassifying the noisy ASR transcriptions. In Figure 1 (b), we only use the inference network during the evaluation process, and the reference network plays a supplementary role in inference network training. This allows us to solve the ASR error problem without any increase in inference time.

To demonstrate the effectiveness of the proposed CCL, we experiment with the SLU benchmark datasets: SLURP (Bastianelli et al., 2020), Timers (Lugosch et al., 2021), FSC (Lugosch et al., 2019), and SNIPS (Coucke et al., 2018). On the SLURP benchmark dataset, the proposed CCL method improves the intent classification performance by 2.59% under severe ASR errors. We also observe that our method outperforms previous module-based or end-to-end approaches on SNIPS. To summarize, our contributions are as follows:

- We investigate the error rates and the error portion that occurs during speech recognition.
- We propose two-stage CCL method to enhance the noisy-channel model. This method correlates the error parts between two transcripts and emphasizes the consistency of latent features.
- We demonstrate that the CCL method improves the performance for noisy ASR transcripts through extensive experiments on various benchmark datasets. In addition, we study the robustness of ASR errors by conducting ablation studies and visualizing the representations to investigate the effectiveness of our proposed method.

2 Related Work

In this section, we introduce the related work from two aspects.

2.1 Spoken Language Understanding

SLU can be implemented by two main approaches: (i) a module-based approach, wherein speech is converted to text through ASR followed by natural language understanding, and (ii) an end-to-end approach, wherein the system is a single network from input to task performance. In the

Dataset	SLURP	Timers	FSC	SNIPS
Google API	0.24	0.15	0.08	-
HuBERT	0.44	0.86	0.17	-
Wav2vec2.0	0.56	0.57	0.27	0.41

Table 1: Word Error Rate (WER) of noisy ASR transcripts on the benchmark datasets.

Clean transcript	Noisy ASR transcript	WER	ASR errors
"turn up"	"turn off"	0.5	up → off
"brighten the lights a little bit"	"brighten the night for little bit"	0.33	lights → night
"please let me know the alarm kept for tuesday's meeting"	"please let me know the alarm cap for tuesday's meet"	0.2	kept → cap, meeting → meet

Table 2: Some examples of clean and noisy ASR transcripts from the SLURP dataset. For ASR errors, the transcripts are tokenized, and the errors between the tokens are shown.

module-based approach, attention-based SLU studies (Liu and Lane, 2016; Goo et al., 2018) achieved good performances with the introduction of the RNN-based attention technique (Bahdanau et al., 2015). Liu and Lane (2016) used an attention-based bidirectional LSTM structure for simultaneous intent classification and slot filling. In a follow-up study, Goo et al. (2018) added slot gates to BLSTM to correlate intents and slots. In contrast, the end-to-end approach has the advantage of being managed as a single network. In end-to-end approaches, a combined CNN and LSTM structure (Chen et al., 2018; Cao et al., 2021) and a GRU-only structure (Serdyuk et al., 2018) were proposed. Although the studies (Seo et al., 2022; Saxon et al., 2021; Qian et al., 2021) employing the Transformer architecture have demonstrated promising performance, there are limitations in training with large-scale data to develop a general dialogue system.

2.2 ASR Errors in a Module-based Approach

By utilizing the pre-trained language models, such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019), numerous studies in the field of SLU have successfully enhanced their generalization capabilities. Chen et al. (2019) combined intent classification and slot filling with the output of BERT for clean transcripts and achieved better performance than existing attention-based models. Qin et al. (2021) proposed the BERT structure with the addition of a co-interactive module that considered slot and intent relationships. However, their exclusive training on large-scale clean datasets made the pre-trained language model susceptible to ASR errors. Ruan et al. (2020) endeavored to mitigate these recognition errors by utilizing the KL divergence between the predicted probabilities for clean and noisy ASR transcripts. In contrast, Kim et al. (2022) used the combined cross-entropy loss of two transcripts to predict intent. Chang and Chen (2022) used contrastive learning to learn the error relationship between two transcripts during additional pre-training, and the noisy problem was mitigated by supervised contrastive learning and self-distillation during the fine-tuning.

3 Method

In this section, we analyze the ASR error problems. We also introduce the noisy-channel model and the CCL method to address these issues. Our method performs the token-based contrastive learning followed by the consistency learning.

3.1 Analyze ASR Errors

We analyze the ASR errors on the four benchmark datasets: SLURP, Timers, FSC, and SNIPS. We introduce the details of datasets in Appendix A. The SLURP and FSC datasets have many classes, which can lead to ambiguities (such as `alarm_set` and `alarm_remove`) between them. We expect these datasets to be more sensitive to ASR errors than simpler ones.

Table 1 shows the error rates according to various ASR modules. We use the noisy ASR transcripts provided by Chang and Chen (2022) and Kim et al. (2022) for SLURP and SNIPS, respectively. In addition, we extract the noisy ASR transcripts directly using Google API, HuBERT (Hsu et al., 2021), and Wav2vec 2.0 (Baevski et al., 2020) speech recognizers on the remaining datasets. FSC has the lowest Word Error Rate (WER), whereas the SLURP and Timers datasets recorded by speakers of various accents have higher WER. Table 2 lists some examples of the ASR errors such as (up, off), (light, night), etc. Unless these ASR errors are trained to the language model, a complete understanding of utterances is still limited.

3.2 Noisy-channel Model

We improve intent classification performance for the analyzed ASR errors based on the noisy-

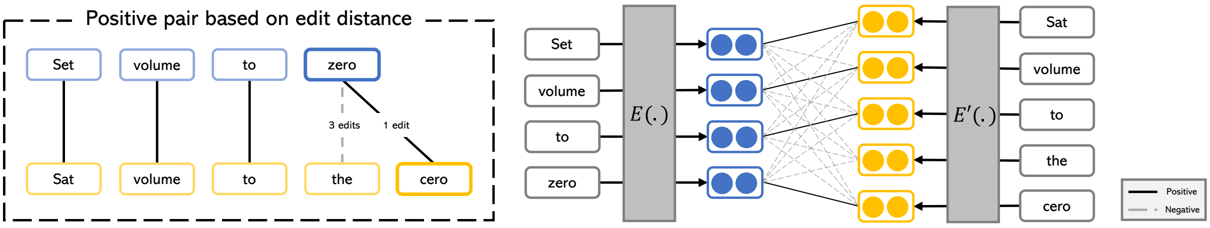


Figure 2: Selective-token contrastive learning finds a positive pair based on the edit distance for the given clean transcript "`Set volume to zero`" and noisy ASR transcript "`Sat volume to the cero`", and conducts contrastive learning as shown on the right side.

channel model. In the enhanced noisy-channel model, an intent probability given the noisy ASR transcript can be computed as a conditional probability for clean and noisy ASR transcripts as follows:

$$\begin{aligned} P(I|\hat{X}) &= \sum_i P(I, X_i|\hat{X}) \\ &\approx P(I, X_c|\hat{X}) \\ &= P(I|X_c, \hat{X})P(X_c|\hat{X}) \end{aligned} \tag{1}$$

where among entire clean transcripts X, X_c is the clean transcript corresponding to the noisy ASR transcript $\hat{X}$, and I is the intent label. We compute an approximation by identifying the correlation between the noisy ASR transcript and the corresponding clean transcript rather than all clean transcripts. This shows that the intent probability for noisy ASR transcript can be maximized by the token alignment of two scripts and the similarity between two scripts. The reparameterized approximation identifies the corresponding clean transcript for the noisy ASR transcript and utilizes the associated clean transcript as a reference to comprehend the intent.

3.3 Token-based Contrastive Learning

We introduce the token-based contrastive learning that aligns clean and noisy ASR transcripts at the word and utterance token-levels. Token-based contrastive loss maximizes the probability $P(X_c|\hat{X})$ by aligning the corresponding word and entire utterance tokens. Given mini-batch B of paired (X_c, $\hat{X}$), features are extracted by the respective encoders E and E'. Here, we use the pre-trained language model RoBERTa as the encoder. Inspired by (Oord et al., 2018; Chen et al., 2020; Gao et al., 2021), we define the contrastive loss as follows:

$$\ell(h, h') = -\frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} \log \frac{e^{(h_i^\top h_i')/\tau}}{\sum_{j \in \mathcal{N}} e^{(h_i^\top h_j')/\tau}} \tag{2}$$

where h and h' are outputs of the encoders, e is an exponential function, and τ is a hyperparameter. $\mathcal{P}$ is the set of the indices of the positive pairs, and $\mathcal{N}$ is the set of indices of all non-matching pairs.

Selective-token contrastive learning. Unlike previous token-based contrastive learning (Su et al., 2022), since ASR errors are caused by insertion, deletion, and substitution, the noisy ASR transcript is not aligned with the clean transcript on the word token-level. We propose the selective-token contrastive learning to create pairs between word tokens via an edit distance algorithm. As shown in Figure 2, we select the positive pairs as the tokens with the minimum number of edits and the negative pairs as non-matching tokens in the batch. Given clean tokens x_c (tokenized X_c) and noisy tokens $\hat{x}$ (tokenized $\hat{X}$), the proposed method is performed with respective outputs z_c and $\hat{z}$ passing through the encoder. Selective-token contrastive loss that correlates the ASR errors for a token representation pair $(z_c, \hat{z})$ is defined as:

$$\mathcal{L}_{sel} = \ell(z_c, \hat{z}) + \ell(z_c, z_c) \tag{3}$$

The representation z_c for clean input contains semantic information for understanding the intent and thus should preserve the clean embeddings when pulling the positive pair close together. Therefore, by adding $\ell(z_c, z_c)$, we train the reference network to maintain its expressiveness when receiving the clean transcript as input. Note that $\ell(z_c, z_c)$ is not equal to zero; this is attributable to the influence of the negative samples in the contrastive loss. According to Wang and Isola (2020), the contrastive loss minimizes the similarity between the features of the negative pair when positive pair features are perfectly aligned.

Utterance contrastive learning. To align the error part of a given $(X_c, \hat{X})$ as the entire utterances, two transcripts are configured as positive pair and trained to increase similarity compared to negative pairs in the batch. Utterance contrastive loss is computed as follows using the [CLS] token representations $(Z_c, \hat{Z})$ that carry the implied meaning of the entire utterance:

$$\mathcal{L}_{utt} = \ell(Z_c, \hat{Z}) \tag{4}$$

Finally, the token-based contrastive objective $\mathcal{L}_{ctr}$ is the weighted sum of the selective-token contrastive loss $\mathcal{L}_{sel}$ and the utterance contrastive loss $\mathcal{L}_{utt}$:

$$\mathcal{L}_{ctr} = \lambda_{ctr}\mathcal{L}_{sel} + (1 - \lambda_{ctr})\mathcal{L}_{utt} \tag{5}$$

where λ_{ctr} is the hyperparameter to adjust the ratio of each loss.

3.4 Consistency Learning

We propose the consistency learning that maintains consistency for clean transcripts' latent features to understand better the intent of noisy ASR transcripts with severe recognition errors. In Figure 1, the consistency learning is used by adding a linear projection $f(.)$ and $f'(.)$ to $E(.)$ and $E'(.)$, respectively. Given the paired sentences $(X_c, \hat{X})$, the final outputs for the two samples are defined as:

$$\begin{aligned} v_c &= f(E(X_c)) \\ \hat{v} &= f'(E'(\hat{X})) \end{aligned} \tag{6}$$

where all tokens for the output of the encoder are flattened. The purpose of consistency learning is to prevent misclassification by additionally utilizing referenced X_c when $\hat{X}$ is given as input. Increasing the consistency between referenced X_c and $\hat{X}$ leads to the probability $P(I|X_c, \hat{X})$ being maximized. We reduce the distance between the noisy latent feature $\hat{v}$ and the referenced latent feature v_c to mitigate the discrepancy. We are also consistent about the target probabilities of p_c and $\hat{p}$ for v_c and $\hat{v}$, respectively, to understand the target intent more precisely. By taking $\hat{X}$ as input and emphasizing the consistency of the latent features and target probabilities for X_c, we expect to avoid making incorrect predictions due to speech recognition errors. Formally, we derive the overall loss of consistency learning as:

$$\mathcal{L}_{con} = \lambda_{con}(v_c - \hat{v})^2 + (1-\lambda_{con})(\log p_c - \log \hat{p})^2 \tag{7}$$

where λ_{con} is hyperparameter.

Reference network. The reference latent feature used in the consistency learning should have a relatively better representation than the noisy latent feature. Therefore, the reference network is trained maximizing the probability $P(y|X_c)$ of label y over input X_c at every step to be consistent with better latent features. After optimizing the reference network with cross-entropy loss at each step, the inference network bootstraps the optimized latent features.

4 Experiment

4.1 Experiment Settings

The proposed CCL is compared with the following three baseline methods:

- SpokenCSE: To solve the ASR error problem of pre-trained RoBERTa, Chang and Chen (2022) proposed contrastive learning between clean and noisy ASR transcripts. In the fine-tuning, self-distillation and supervised contrastive learning were applied to regulate the misprediction of noisy ASR transcripts. We conducted further experiments when results were unavailable for SpokenCSE.
- Clean-CE: The language model trained only on clean transcripts is vulnerable to noisy input. To investigate these vulnerabilities, we employ a technique that utilizes clean transcripts as inputs and fine-tunes them using cross-entropy (CE).
- Noisy-CE: Fine-tuning the noisy ASR transcript is the common method to solve the ASR error problem. While this method seems simple, it can be biased toward noisy ASR transcripts and lead to performance degradation of the clean transcript.

We conduct comparison experiments between baseline and CCL methods on SLURP, Timers, and FSC datasets. All models are based on RoBERTa to compare performance under the same conditions as Chang and Chen (2022). In addition, the CCL method is compared with various previous works in the field of SLU using SNIPS. We refer to the noisy ASR transcripts converted by different recognizers as Noisy$_{wer}$, and the clean transcripts as Clean. In the token-based contrastive learning, we use the same hyperparameters: learning rate 1e-6 and λ_{ctr} 0.7. The learning rate is selected for consistency learning from {1e-5, 3e-5, 5e-5, 7e-5, 1e-4}. We

method	Noisy$_{0.24}$		Noisy$_{0.44}$		Noisy$_{0.56}$		Clean	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
Clean-CE	82.6	82.8	64.7	60.51	50.98	52.85	96.35	96.25
Noisy-CE	85.19	85.23	77.72	73.76	70.53	70.65	96.3	96.29
SpokenCSE	85.26	85.66	79.08	**78.91**	70.31	71.03	95.82	96.21
CCL (Ours)	**86.22**	**86.27**	**81.51**	76.83	**73.12**	**73.13**	**96.99**	**96.98**

Table 3: Results for macro F1-score and accuracy performances on SLURP dataset for our model trained with CCL method. We compare the CCL method and other baselines for clean and noisy transcripts.

method	Timers				FSC			
	Noisy$_{0.15}$	Noisy$_{0.86}$	Noisy$_{0.57}$	Clean	Noisy$_{0.08}$	Noisy$_{0.17}$	Noisy$_{0.27}$	Clean
Clean-CE	99.69	94.87	89.15	100	95.83	92.99	87.13	**100**
Noisy-CE	99.97	99.66	99.29	100	97.19	99.24	97.23	93.35
SpokenCSE	99.97	99.71	99.17	100	99.18	99.24	98.5	98.02
CCL (Ours)	**99.98**	**99.73**	**99.45**	**100**	**99.31**	**99.26**	**98.87**	99.16

Table 4: Intent classification accuracy results of Timers and FSC datasets.

tune λ_{con} of best value that is 0.0 to 1.0. The batch size in our experiments is 64, the same for all. All experiments are performed with NVIDIA TITAN RTX 24GB GPU.

4.2 Main Results

Table 3 lists the results of CCL and baseline methods on the SLURP test set. Clean-CE accurately predicts approximately half of the answers to Noisy$_{0.56}$, indicating that the language models pre-trained with only clean transcripts are susceptible to ASR errors. We also achieve the best results when measured by macro F1-score, which can consider the class imbalance of the SLURP dataset. Our method significantly outperforms Noisy-CE by 2.59% (accuracy) and 2.48% (F1-score) in Noisy$_{0.56}$. The performance gain is more significant for higher error rates than lower ones because our method aligns more ASR errors in noisy transcripts with clean ones containing contextual information to predict intent. Similar results are seen when running with four seeds to calculate the mean and standard deviation in Appendix Table 12. Moreover, we attain the highest performance on clean transcripts, highlighting the robustness of the CCL method in handling speech recognition errors without compromising its performance on clean transcripts.

The CCL and compared methods are evaluated in various settings. We employ two distinct datasets: simple Timers and complex FSC. Table 4 lists the accuracy of clean and noisy ASR transcripts. Challenging tasks involving complex classes may exhibit heightened sensitivity to speech recognition errors, leading to performance degradation. However, we observe that the performance gain for complex FSC is more significant than that for simple Timers compared to the highest-performing method among the baselines. This indicates the proposed CCL method is robust to ASR errors under challenging tasks.

Table 5 presents the intent classification accuracy results for the SNIPS dataset. We compare the performance of various approaches in the SLU field. In end-to-end, Lai et al. (2020) and Kim et al. (2021) employed training methods that simultaneously utilized audio and text data. Module-based approaches have attempted to address the ASR error problem using a method that increases the cosine similarity between two transcripts (Huang and Chen, 2020) or approaches that use phoneme sentences (Chen et al., 2021). Among the existing module-based approaches, the method with dual-BERT (Kim et al., 2022) achieved the best performance, although it was 0.03% lower than the end-to-end approach. Whether end-to-end or module-based, our method achieves a higher performance of 98.72%.

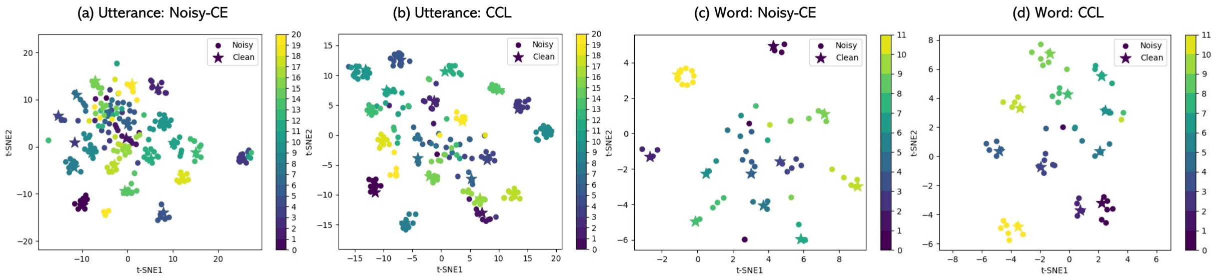


Figure 3: t-SNE visualization of clean and noisy ASR transcripts in word and utterance token-levels for SLURP test set. (a-b) each color represents a sample of clean transcript and associated noisy ASR transcripts. (c-d) each color indicates the tokens of sample, such as clean token and noisy ASR tokens.

approach	Model	Acc.
End-to-end	Lai et al. (2020)	98.65
	Kim et al. (2021)	96.7
Module	Huang and Chen (2020)	89.55
	Kim et al. (2022)	98.62
	Chen et al. (2021)	85.4
	CCL (Ours)	**98.72**

Table 5: Comparison of intent classification accuracy for SNIPS dataset between the end-to-end and the module-based approaches.

4.3 Further Analysis

In this section, we analyze the factors that led the model trained using CCL to outperform the compared models. In addition, we conduct a range of experiments to verify the efficacy of the proposed CCL.

Token-based contrastive learning. To better understand why CCL is robust to ASR errors, we visualize the outputs of encoders trained with CCL and Noisy-CE methods, respectively. We randomly select a sample (clean transcript, noisy ASR transcripts) in the SLURP test set with multiple speakers. Figure 3 (a-b) visualizes the utterance token representations of multiple samples, indicating that the CCL method yields a much better separation than Noisy-CE; since our method can align several noisy ASR transcripts and clean transcript with high performance through the utterance contrastive learning. Figure 3 (c-d) shows the results of sample containing pairs (clean word token, noisy word tokens). In CCL, the distance between clean and noisy tokens is reduced, demonstrating the effectiveness of selective-token contrastive learning. We analyze that the alignment of two transcripts in utterance and word token-levels leads to robust to ASR errors.

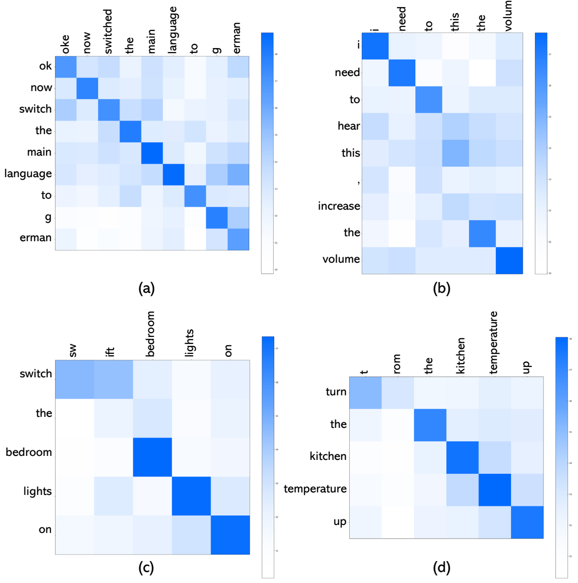


Figure 4: Token-based similarity map visualization results for FSC validation and test sets. X-axis and Y-axis represent clean and noisy ASR transcripts, respectively.

Figure 4 presents token-based similarity maps on the FSC valid and test sets. We observe that substituted tokens and aligned clean tokens have high similarity even when they have different meanings in Figure 4 (a). We can also see that clean and noisy ASR transcripts are semantically aligned when there are different token lengths caused by deletion (Figure 4 (b)). In contrast, Figure 4 (c-d) shows various speech recognition errors such as substitution, insertion, and deletion. The visualizations prove that selective-token contrastive learning works well in correlating the ASR errors, even when the tokens are not aligned or various ASR errors co-occur. Furthermore, we describe the analysis of the utterance-based similarity maps in Appendix B.2.

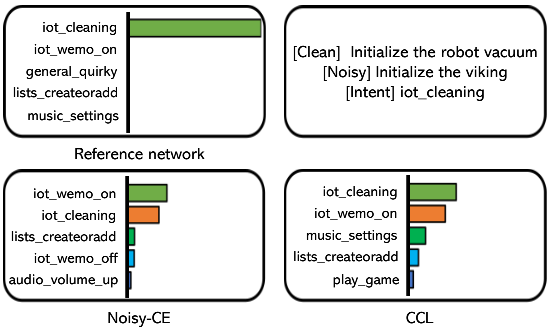


Figure 5: Top-5 intent prediction distributions on SLURP dataset from reference network, inference network (trained with CCL), and baseline model (trained Noisy-CE).

Consistency learning. We explore the advantages of the proposed consistency learning over the cross-entropy loss commonly used in fine-tuning. Figure 5 shows the probability distribution of the SLURP test set for the reference network, inference network (trained with CCL), and baseline model (trained Noisy-CE). The probability distribution of the reference network is the result for the clean transcript as input, and the rest are the results for the noisy ASR transcript. The top-5 probability values out of a set of 60 classes are depicted in all the visualizations. While the reference network accurately predicts the answer `iot_cleaning`, Noisy-CE encounters confusion between `iot_cleaning` and `iot_wemo_on` for the noisy ASR transcript where important keywords were deleted and replaced. On the other hand, the CCL method exhibits a similar level of confusion as Noisy-CE, but this method allows the model to predict the correct answer. We analyze that the CCL method mitigates misclassifications by maintaining consistency with the latent feature of the reference network rather than relying on a fixed target.

Ablation studies. We conduct an ablation study to investigate the impact of the losses used in the CCL method. Table 6 demonstrates that the CCL method achieves the best accuracy. We observe that the higher the speech recognition error rate, the more significant the performance degradation while removing the token-based contrastive loss. This result shows that the proposed token-based contrastive learning is increasingly effective in noisy environments. In token-based contrastive learning, better performance is observed when mismatched tokens are selectively correlated using $\mathcal{L}_{sel}$, rather than $\mathcal{L}_{utt}$. In the fine-tuning, we attain improved results employing the consistency loss rather than the commonly used cross-entropy loss. This highlights the significance of preserving consistency between clean and noisy latent representations. We conduct additional ablation experiments using cross-entropy loss instead of consistency objective in Appendix B.3.

method	SLURP		Timers	
	Noisy$_{0.24}$	Noisy$_{0.56}$	Noisy$_{0.15}$	Noisy$_{0.57}$
$\mathcal{L}_{ce}$	85.19	70.53	99.97	99.29
$\mathcal{L}_{con}$	85.95	72.81	99.88	99.26
$\mathcal{L}_{con}$ $+ \mathcal{L}_{sel}$	86.17	72.3	99.89	99.26
$\mathcal{L}_{con}$ $+ \mathcal{L}_{utt}$	86.13	72.16	99.89	99.24
$\mathcal{L}_{con}$ $+ \mathcal{L}_{sel}$ $+ \mathcal{L}_{utt}$	**86.22**	**73.12**	**99.98**	**99.45**

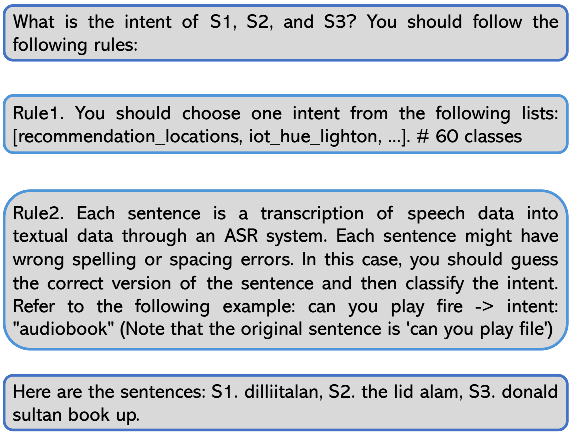
Table 6: Ablation study of different losses.

Noisy ASR transcript	GPT4	CCL	GT
dilliitalan	iot_hue_lightdim	alarm_remove	alarm_remove
the lid alam	alarm_set	alarm_remove	alarm_remove
donald sultan book up	play_audiobook	qa_factoid	qa_factoid

Table 7: Comparative analysis of GPT4 and CCL

4.4 Comparision with Large Language Models

Recent advances in Large Language Models (LLMs) have attracted significant research interest due to their representation power, which has led to notable successes in natural language processing (Ouyang et al., 2022; Achiam et al., 2023; Touvron et al., 2023), multi-modal (Li et al., 2023; Lin et al., 2023), and computer vision (Zhao et al., 2023; Wang et al., 2024) fields. However, it remains unclear whether LLMs trained only on the clean text can identify and contextually understand the ASR errors. We conduct a comparison experiment on GPT4 (Achiam et al., 2023) and our CCL method to verify the ability to understand the errors. Through this experiment, we aim to observe how each method classifies intent labels on the same noisy ASR transcripts from the SLURP dataset.



What is the intent of S1, S2, and S3? You should follow the following rules:

Rule1. You should choose one intent from the following lists: [recommendation_locations, iot_hue_lighton, ...]. # 60 classes

Rule2. Each sentence is a transcription of speech data into textual data through an ASR system. Each sentence might have wrong spelling or spacing errors. In this case, you should guess the correct version of the sentence and then classify the intent. Refer to the following example: can you play fire -> intent: "audiobook" (Note that the original sentence is 'can you play file')

Here are the sentences: S1. dilliitalan, S2. the lid alam, S3. donald sultan book up.

Figure 6: Prompt structure for LLMs.

For GPT4, we provided the prompt as shown in Figure 6. As shown in the Table 7, GPT-4 can understand the intent `alarm` in the transcript containing the minor ASR errors. However, it still has trouble detecting severe ASR errors, which could lead to misclassification. In contrast, CCL detects a wide range of ASR errors, even if they are hard to understand, allowing our method to predict precise intention.

As mentioned above, LLMs demonstrate constraints in understanding the ASR errors without additional training. We fine-tuned the decoder-based and encoder-based LLMs using either the cross-entropy loss or the CCL method. Due to the resource limitation, we use the GPT-Neo (Black et al., 2021) and GPT-2 (Radford et al., 2019) models. Table 8 shows the results of decoder-based and encoder-based models on the SLURP test set. The encoder-based language model consistently outperforms decoder-based language models, even with fewer parameters (110M). We believe the encoder structure is apt for classification tasks because the model can capture contextual information in bi-directions. Furthermore, our CCL method enhances the performance of the encoder- and decoder-based language models in extremely noisy environments.

5 Conclusion

We have introduced CCL, a novel method to solve the ASR error problem of pre-trained language models. We designed the two-stage method: (i) token-based contrastive learning to correlate the ASR errors at both the word and utterance token-levels and (ii) consistency learning to maintain coherence with the latent feature of relatively better clean transcripts. Experiments and analysis on four benchmark datasets have demonstrated that our model significantly outperforms previous models in various speech recognition error rates.

model	Noisy$_{0.24}$	Noisy$_{0.56}$
GPT-Neo (125M)	76.55	57.44
GPT-Neo+CCL	75.99	64.48
GPT-2 (355M)	83.17	65.94
GPT-2+CCL	80.33	70.23
RoBERTa (110M)	85.19	70.53
RoBERTa+CCL	**86.22**	**73.12**

Table 8: Performance of various models on the SLURP benchmark dataset.

6 Limitations

Although our work makes further progress in the ASR problem of SLU, it is subject to two potential limitations. First, when ASR errors seriously occur that the original meaning is unrecognizable, the model trained with CCL predicts wrong answers as in Appendix B.4. In the future work, we plan to develop a method to address the mismatch between noisy ASR transcripts and labels. Then, there has been a growing interest in the field of multilingual, but most ASR error studies have centered on English. We need to expand the ASR error problem to multilingual in the future work.

7 Acknowledgements

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2024-2020-0-01808) supervised by the IITP(Institute of Information & Communications Technology Planning & Evaluation).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

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, Online. Association for Computational Linguistics.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Yiran Cao, Nihal Potdar, and Anderson R Avila. 2021. Sequential end-to-end intent and slot label classification and localization. *arXiv preprint arXiv:2106.04660*.

Ya-Hsin Chang and Yun-Nung Chen. 2022. Contrastive Learning for Improving ASR Robustness in Spoken Language Understanding. In *Proc. Interspeech 2022*, pages 3458–3462.

Qian Chen, Wen Wang, and Qinglin Zhang. 2021. Pre-Training for Spoken Language Understanding with Joint Textual and Phonetic Representation Learning. In *Proc. Interspeech 2021*, pages 1244–1248.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Yuan-Ping Chen, Ryan Price, and Srinivas Bangalore. 2018. Spoken language understanding without speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6189–6193. IEEE.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

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*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

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.

Chao-Wei Huang and Yun-Nung Chen. 2020. Learning asr-robust contextualized embeddings for spoken language understanding. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8009–8013. IEEE.

June-Woo Kim, Hyekyung Yoon, and Ho-Young Jung. 2022. Improved spoken language representation for intent understanding in a task-oriented dialogue system. *Sensors*, 22(4):1509.

Seongbin Kim, Gyuwan Kim, Seongjin Shin, and Sangmin Lee. 2021. Two-stage textual knowledge distillation for end-to-end spoken language understanding. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7463–7467. IEEE.

Cheng-I Lai, Jin Cao, Sravan Bodapati, and Shang-Wen Li. 2020. Towards semi-supervised semantics understanding from speech. *arXiv preprint arXiv:2011.06195*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech 2016*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Loren Lugosch, Piyush Papreja, Mirco Ravanelli, Abdelwahab Heba, and Titouan Parcollet. 2021. Timers and such: A practical benchmark for spoken language understanding with numbers. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks.*, pages 1–11.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. 2019. Speech model pre-training for end-to-end spoken language understanding. *arXiv preprint arXiv:1904.03670*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

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 *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7458–7462. IEEE.

Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. 2021. A co-interactive transformer for joint slot filling and intent detection. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8193–8197. IEEE.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Weitong Ruan, Yaroslav Nechaev, Luoxin Chen, Chengwei Su, and Imre Kiss. 2020. Towards an asr error robust spoken language understanding system.

Michael Saxon, Samridhi Choudhary, Joseph P. McKenna, and Athanasios Mouchtaris. 2021. End-to-End Spoken Language Understanding for Generalized Voice Assistants. In *Proc. Interspeech 2021*, pages 4738–4742.

Seunghyun Seo, Donghyun Kwak, and Bowon Lee. 2022. Integration of pre-trained networks with continuous token interface for end-to-end spoken language understanding. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7152–7156. IEEE.

Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio. 2018. Towards end-to-end spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5754–5758. IEEE.

Yixuan Su, Fangyu Liu, Zaiqiao Meng, Tian Lan, Lei Shu, Ehsan Shareghi, and Nigel Collier. 2022. Tacl: Improving bert pre-training with token-aware contrastive learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2497–2507.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR.

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. 2024. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *Advances in Neural Information Processing Systems*, 36.

Qi Zhao, Shijie Wang, Ce Zhang, Changcheng Fu, Minh Quan Do, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. 2023. Antgpt: Can large language models help long-term action anticipation from videos? In *The Twelfth International Conference on Learning Representations*.

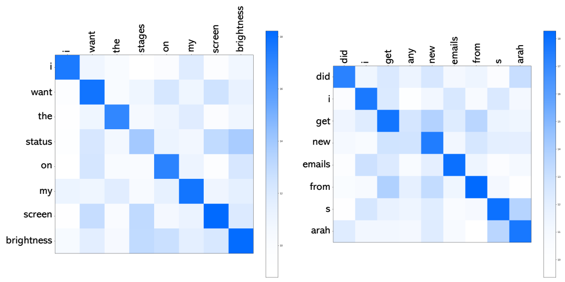


Figure 7: Visualization results of further similarity maps between tokens.

A Dataset Description

We experiment with the SLU benchmark datasets:

- SLURP: The SLURP dataset, a challenging SLU benchmark dataset, consists of house robot voice commands from 177 speakers. Since SLURP contains inaccurate labels, we use a dataset from Chang and Chen (2022) where the labels have been denoised. We use 60 intent labels and divide each dataset into 50,627, 13,078, and 10,992 for train, valid, and test, respectively.

- Timers: The Timers dataset is simple data recorded from 95 speakers, smaller than SLURP. This dataset consists of four labels: SetAlarm, SetTimer, SimpleMath, and UnitConversion. We split the data into training, validation, and testing of 7,000, 1,000, and 12,000, respectively.

- FSC: The FSC dataset consists of commands for smart-home or voice assistants of 97 speakers. As in Lugosch et al. (2019), we use 31 intents, 23,132 training, 3,118 validation, and 3,793 testing.

- SNIPS: A well-known text-based intent classification SNIPS dataset consists of 13,084 training, 700 validation, and 700 evaluation. We classify seven intents using an ASR transcript from Kim et al. (2022), which transcribed synthetic speech files into text.

B Additional Analysis

B.1 Selective-token Contrastive Learning

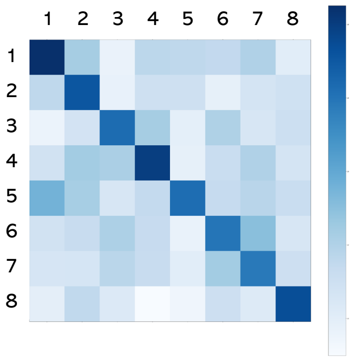
Figure 7 shows the further examples of similarity maps that visually represents the similarity between the outputs of the inference and reference networks trained using the token-based contrastive loss. The visualization experiment uses the SLURP test set. In Figure 7 (left), a recognition error occurs for clean transcript “`I want the status on my screen brightness`” and noisy ASR transcript “`I want the stages on my screen brightness`”, where `status` is replaced by `stages`. Despite `status` and `stages` being distinct tokens, their similarity scores toward each other are high. In Figure 7 (right), we can observe that the insertion error token `any` is not aligned with the other tokens. These visualizations demonstrate that we can identify the recognition errors even while handling unfamiliar data by employing the selective-token contrastive learning.

method	SLURP		Timers	
	Noisy$_{0.24}$	Noisy$_{0.56}$	Noisy$_{0.15}$	Noisy$_{0.57}$
$\mathcal{L}_{ce}$	85.19	70.53	99.97	99.29
$\mathcal{L}_{ce}$ + $\mathcal{L}_{sel}$	84.61	70.52	99.98	**99.73**
$\mathcal{L}_{ce}$ + $\mathcal{L}_{utt}$	84.66	70.61	99.97	99.66
$\mathcal{L}_{ce}$ + $\mathcal{L}_{sel}$ + $\mathcal{L}_{utt}$	84.92	71.02	99.98	99.67
$\mathcal{L}_{cons}$	85.95	72.81	99.88	99.26
$\mathcal{L}_{cons}$ + $\mathcal{L}_{sel}$	86.17	72.3	99.89	99.26
$\mathcal{L}_{cons}$ + $\mathcal{L}_{utt}$	86.13	72.16	99.89	99.24
$\mathcal{L}_{cons}$ + $\mathcal{L}_{sel}$ + $\mathcal{L}_{utt}$	**86.22**	**73.12**	**99.98**	99.45

Table 9: Total ablation studies of different losses.

B.2 Utterance Contrastive Learning

We compare the contextual representations of clean and noisy ASR transcripts to demonstrate that the utterance contrastive objective works well. To quantify contextual similarity, we calculate the similarity score between [CLS] tokens in each of the reference and inference networks. Figure 8 shows the similarity map for samples, where the x-axis and y-axis represent noisy ASR and clean transcripts, respectively. Overall, the diagonal sim-



	Clean transcript	Noisy ASR transcript
1	“do i have anything booked for tomorrow”	“do I have anything booked for to morrow”
2	“add valentine’s day with her tomorrow”	“at valentine’s day with her to morrow”
3	“can you able to say the word demonetisation”	“able to say the word demonization”
4	“if you could talk about three things what would they be”	“could talk about three things what would davy”
5	“book me a train ticket to chicago for tomorrow morning before ten”	“to chikovofor to morrow morning before death”
6	“what is the name of the lead singer in this band”	“the name of lee singer in this ban”
7	“can you recommend any pub in mg road”	“do you recommend any hobby in m butchy mo”
8	“show me delivery near me”	“show me tdelivery near”

Figure 8: Left side shows the similarity map visualization between randomly sampled examples from the SLURP test set. Details of the sampled example are on the right side.

Clean transcript	**Noisy ASR transcript**	**PR**	**GT**
health	naws	news_query	general_quirky
please turn lights off	you send the baxle	transport_taxi	iot_hue_lightoff
create a new list	read in yolist	lists_query	lists_createoradd
delete alarm	dilliitalan	alarm_remove	alarm_remove
delete alarm	the lid alam	alarm_remove	alarm_remove
what time is it	what’s im he say	datetime_query	datetime_query

Table 10: Intent classification results of the CCL method for noisy ASR transcripts on the SLURP dataset.

ilarity scores are higher than the other parts. In particular, the clean transcript “`Book me a train ticket to chicago for tomorrow morning before ten`” and the noisy ASR transcript “`To chikovofor to tomorrow morning before death`” have high similarity scores, even though the speech recognition error has lost essential keywords. This indicates that contrastive learning at the utterance level is effective in implicitly detecting contextual disparities between two transcripts.

B.3 Ablation studies

Table 9 shows the results of additional ablation experiments with cross-entropy loss while retaining token-based contrastive learning. In most cases, token-based contrastive learning typically demonstrates enhanced performance by employing the consistency loss instead of cross-entropy loss. This highlights the significance of preserving the consistency between clean and noisy latent representations, particularly in environments where speech recognition errors are prevalent. On the other hand, the best performance is achieved on Noisy$_{0.57}$ when the cross-entropy loss and the selective-contrastive loss are used together. We analyze that cross-entropy loss works well in the simple task with four intents. However, we need the consistency learning that works well on complex tasks to capture the diverse intents of users in the real world.

method	Noisy$_{0.24}$	Noisy$_{0.56}$
CN-CE	84.86	71.12
SimCSE	83.51	70.71
CCL (Ours)	**86.22**	**73.12**

Table 11: Performance of further baselines on SLURP benchmark.

B.4 Qualitative results

Table 10 lists the intent prediction results on the SLURP test set. From rows 1-3, the proposed CCL predicts intent that is different from the ground truth; however, this is matched when considered in the context of a noisy ASR transcript. In other words, our method tends to predict incorrect answers when the original meaning is lost in the speech recognition process. Although such results with 1.0 WER are rare in the SLU benchmark dataset, we need to improve these issues for real-world use in noisy environments. For future work, we will develop a method that addresses the mismatch between noisy ASR transcripts and their corresponding labels. As shown in rows 4-6, our method accurately predicts intent class when the original meaning is partially preserved.

C Comparison with further baselines

We compare the CCL method with the following baselines:

- CN-CE: A simple baseline is to train a model on the combination of clean and noisy ASR transcripts. This might make it more reasonable than another baseline since it utilizes both

method	SLURP			Timers			FSC		
	Noisy$_{0.24}$	Noisy$_{0.44}$	Noisy$_{0.56}$	Noisy$_{0.15}$	Noisy$_{0.86}$	Noisy$_{0.57}$	Noisy$_{0.08}$	Noisy$_{0.17}$	Noisy$_{0.27}$
Clean-CE	$83.43_{\pm 0.55}$	$64.67_{\pm 0.03}$	$52.13_{\pm 0.79}$	$99.70_{\pm 0.07}$	$93.37_{\pm 4.05}$	$90.09_{\pm 3.05}$	$93.84_{\pm 1.33}$	$91.36_{\pm 1.82}$	$86.87_{\pm 1.18}$
Noisy-CE	$84.97_{\pm 0.20}$	$79.04_{\pm 0.88}$	$70.44_{\pm 0.08}$	$\mathbf{99.98}_{\pm 0.01}$	$99.69_{\pm 0.02}$	$99.09_{\pm 0.15}$	$96.01_{\pm 0.92}$	$99.09_{\pm 0.14}$	$98.28_{\pm 0.70}$
SpokenCSE	$85.21_{\pm 0.24}$	$78.72_{\pm 0.43}$	$69.72_{\pm 0.40}$	$99.97_{\pm 0.01}$	$\mathbf{99.73}_{\pm 0.02}$	$99.24_{\pm 0.08}$	$96.16_{\pm 3.44}$	$99.19_{\pm 0.07}$	$98.53_{\pm 0.05}$
CCL (Ours)	$\mathbf{86.28}_{\pm 0.10}$	$\mathbf{81.45}_{\pm 0.13}$	$\mathbf{72.69}_{\pm 0.30}$	$99.97_{\pm 0.01}$	$99.64_{\pm 0.12}$	$\mathbf{99.33}_{\pm 0.32}$	$\mathbf{99.23}_{\pm 0.06}$	$\mathbf{99.32}_{\pm 0.05}$	$\mathbf{98.77}_{\pm 0.11}$

Table 12: Performance comparison results with the average and standard deviation on the noisy ASR transcripts from SLURP test set. Each result is calculated across 4 random seeds. We denote optimal performance in bold.

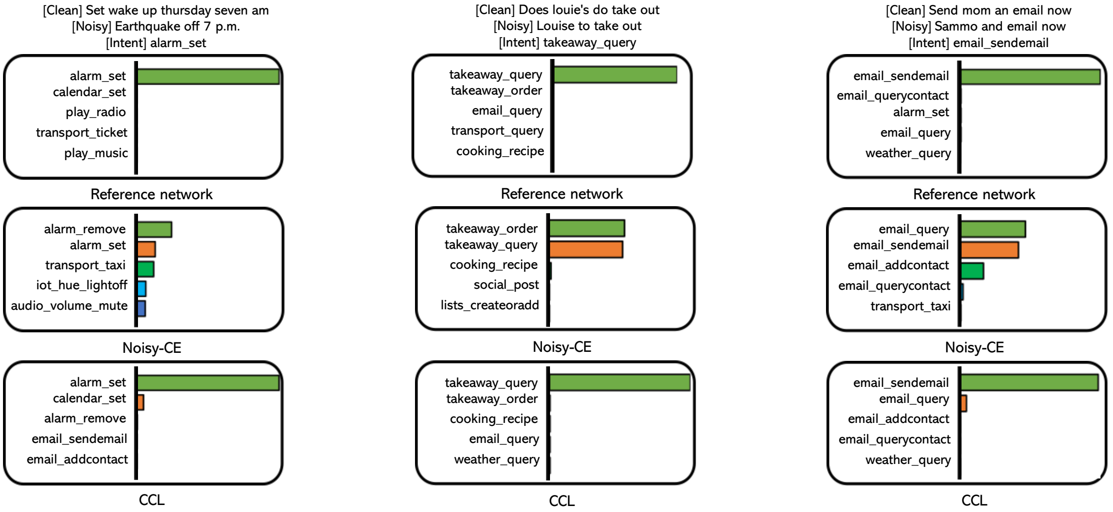


Figure 9: Comparison of prediction distributions on the SLURP test set from reference networks, inference networks, and Noisy-Net.

clean and noisy data, which is consistent with the CCL method.

- SimCSE (Gao et al., 2021): By applying contrastive learning, the method significantly advances the state-of-the-art in sentence embedding. The method maximizes the similarity between the original and randomly dropped features. We employ the identical noisy ASR transcripts to serve as a positive pair.

Table 11 shows that the CCL and CN-CE methods using both clean and noisy ASR transcripts achieve better results than SimCSE only trained with noisy ASR transcripts. Furthermore, our method, which focuses on correlating errors and emphasizing consistency between two transcripts, is better suited to noisy environments than the simple CN-CE method that employs the cross-entropy loss to train on a fixed target.