# Can Noisy Cross-Utterance Contexts Help Speech-Recognition Error Correction?

Seongmin Lee,* Kohki Tamura,*Tomoaki Nakamura,* and Naoki Yoshinaga

**Abstract** Although automatic speech recognition (ASR) is required to apply language technology to ever-increasing multimedia data, the existing ASR methods have been studied mainly on utterance-based datasets and the resulting models are thus not optimized for transcribing the recorded multimedia data. In this study, we investigate the utility of cross-utterance context in ASR error correction, which leverages ASR-based contexts for error correction. To address the data sparseness problem, we utilize a pre-trained text generation model, T5. Experimental results on CORAAL datasets transcribed with NVIDIA STT ASR confirmed that our T5-based context-aware ASR error correction improves the word error rate (WER) of the correction by 2.73 for all utterances and 5.12 for utterances with proper nouns.

## 1 INTRODUCTION

The sudden increase of online communication due to the COVID-19 pandemic accelerates the accumulation of massive multimedia data. Although these multimedia data have valuable information, they accompany a little metadata that describes the content. To improve data accessibility, automatic speech recognition (ASR) is often used to transcribe speech in the data; this will not only help hearing-impaired persons and second-language learners to watch the content but also enable a better search using natural-language queries. Although these multimedia data have valuable information, it requires automatic speech recognition (ASR) to transcribe the speech to perform text-based information retrieval and extraction.

Seongmin Lee, Kohki Tamura, and Tomoaki Nakamura
The University of Tokyo,
e-mail: {lee-s,tamura-k,nakamu-t}@tkl.iis.u-tokyo.ac.jp

Naoki Yoshinaga
Institute of Industrial Science, The University of Tokyo,
e-mail: ynaga@iis.u-tokyo.ac.jp

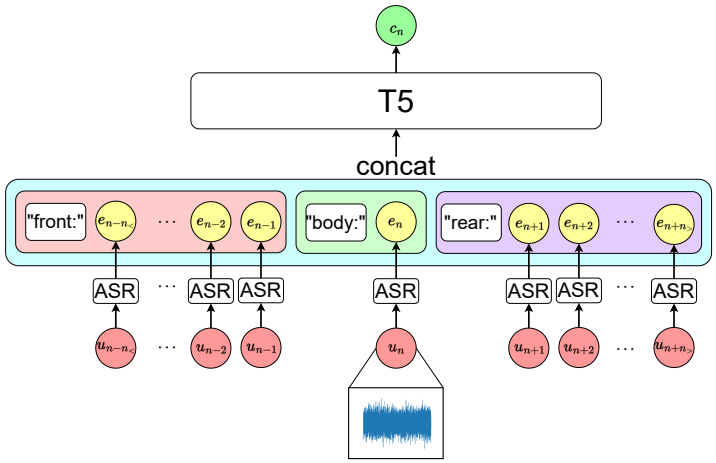


**Fig. 1** Cross-utterance context-aware ASR error correction.

The existing ASR models are, however, not tuned to transcribe these recorded speech data, since common settings take an utterance as a unit of transcription and the existing datasets cover only a limited number of domains. Meanwhile, to aggressively correct ASR errors and perform a quick, lightweight adaptation to a new domain, ASR error correction has been studied [19, 7, 11, 9, 27]. These studies utilize language models trained on massive text to rescore ASR hypotheses or even to generate transcription from erroneous ASR transcription without referring to the acoustic information. However, few studies have explored the effectiveness of noisy but long ASR-based cross-utterance contexts in the text-based ASR error correction.

In this study, to effectively correct erroneous transcriptions of off-the-shelf ASR systems, we propose a method of correcting ASR outputs by leveraging noisy but long cross-utterance ASR-based contexts of the target utterances (Figure 1). Inspired by context-aware neural machine translation using a document-level LM [17], we adopt a Transformer [22]-based pre-trained model, T5 [14], to perform cross-utterance context-aware ASR error correction by generation. We fine-tune T5 to correct ASR errors by giving ASR outputs of past and future utterances of the target utterance along with the ASR output of the target utterance. By training an ASR error correction model independent of ASR, we can not only quickly adjust the black-box ASR to a new domain but also aggressively correct ASR outputs by leveraging longer cross-utterance contexts than ASR inputs for the correction.

We applied the NVIDIA STT Conformer-CTC Large[1] as an off-the-shelf ASR system to transcribe the challenging CORAAL dataset [8] and used the resulting transcriptions to evaluate our ASR error correction model. Experimental results confirmed that noisy ASR-based contexts are yet effective in correcting ASR errors when we used T5-based error correction model.

[1] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_conformer_ctc_large

## 2 RELATED WORK

In this section, we first introduce existing cross-utterance context-aware ASR methods (§ 2.1), and next review existing methods for ASR error correction (§ 2.2). We then mention context-aware text generation (§ 2.3).

### *2.1 ASR Using Language Models*

ASR has been formalized as a task of transcribing a single utterance, and most of the ASR datasets are provided for this setting, since ASR is required to map long acoustic feature sequences to much shorter sequences of letters or words. A few studies have explored the use of language models (LMs) to consider short cross-utterance contexts; past utterances [13] and future utterances [18]. Since these methods use LMs to adjust ASR hypotheses, they are less powerful for obtaining accurate ASR outputs in unseen domains in which ASR cannot guess good hypotheses.

In this work, we focus on a strong off-the-shelf ASR system that is trained on a combination of various ASR datasets, and attempt to correct its outputs by using a powerful pre-trained LM-based ASR error correction while considering noisy but longer ASR-based contexts (transcriptions) into consideration.

### *2.2 ASR Error Correction*

ASR error correction is formulated as a text-to-text task from ASR outputs to gold transcriptions [19, 6]. Recent models [7, 23, 9, 27] exploit powerful neural generation models such as Transformer [22]. These models, however, have rarely utilized only inter-utterance contexts for the correction. Wang et al. [23] indirectly used cross-utterance contexts by extracting person names in past utterances to correct ASR outputs. Zhao et al. [28] and Dutta et al. [4] utilized the pre-trained model, BART [9], for generation-based ASR error correction. Ma et al. [10] leveraged T5 for ASR error correction, using n-best ASR hypotheses as input. All of these studies did not consider cross-utterance contexts, and do not fully exploit the true potential of pre-trained models that can take dozens of utterances as inputs.

In this work, we rely on the strong generation capability of a pre-trained LM trained on massive text, and specifically utilize T5 as a text-based ASR correction model to consider ASR-based contexts into consideration in the text-based ASR error correction.

### *2.3 Context-Aware Text Generation*

Several studies leverage surrounding contexts of inputs in text generation tasks such as machine translation [20] and grammatical error correction [3]. These models are classified into two types; i) 2-to-1 models [20, 2, 16] that simply append context sentences to the input, and ii) multi-encoder models that separately encode the input and its contexts [24, 21, 12]. The most important factor to obtain better performance in the context-aware generation is the size of training data [16]. In the case of machine translation, since most of the existing datasets are parallel sentences without contexts, recent studies resorted to document-level LMs to perform context-aware generation using only parallel sentences without contexts as the training data [26, 17].

In this work, we leverage a pre-trained LM for text generation to compensate for the lack of training data in the ASR error correction, and finetune it to perform the 2-to-1 context-aware generative error correction for ASR.

## 3 PROPOSED MODEL

This section proposes a method to correct an utterance transcribed by off-the-shelf ASR systems while referring to the cross-utterance contexts. In this study, we target situations where we correct existing ASR transcriptions of recorded multimedia data. We therefore develop a text-based generative ASR error correction model without referring to the acoustic information and ASR hypotheses, and evaluate the utility of noisy but long ASR-based contexts for ASR error correction.

Assuming the input (ASR transcriptions) as the source language and the output (correct transcriptions) as the target language, the proposed method translates the input to the output using the framework of 2-to-1 context-aware neural machine translation [20, 16]. Specifically, the model takes the target transcribed utterance $e_n$ for correction, along with $n_<$ past transcribed utterances, $e_{n-n_<:n-1}$, and future transcribed utterances, $e_{n+1:n+n_>}$. It then outputs a corrected transcription, $c_n$, of the target transcribed utterance $e_n$ (Figure 1).

We need the training data of ASR error correction to solve it as a text-to-text task. We can build it from existing ASR datasets by using the target ASR system to transcribe the inputs of ASR datasets, or newly create it for the target domain from scratch by manually correcting the ASR transcriptions. However, since context-aware generation models require a large amount of training data in practice, the size of training data we can prepare for the target domain may not be enough to train an accurate error correction model. One possible workaround for this issue is to use automatic speech synthesis to generate pseudo (silver) training data for the target domain [6]. However, the quality of the silver training data will be affected by the performance of automatic speech synthesis on the target domain; we will need much more silver training data to train an accurate generation model than the gold training data [16].

Therefore, inspired by a context-aware decoder that combines a sentence-level translation model with a document-level LMs [17], we train an error correction model from a pre-trained model for text generation, T5 [14]. Although T5 itself is not pre-trained for error correction, it can be fine-tuned with task-specific downstreaming datasets. The subword-based generation model pre-trained on the diverse and massive amount of text will be expected to have the ability to decode rare words such as proper nouns in various domains.

When we feed cross-utterance contexts along with the target transcription for correction to T5, the position of these contexts may matter, since the relative position embeddings in T5 may not be learned well for distant tokens, due to the diversity in total input length. Since T5 accepts the input text with task-specific prefixes, we formulate the input to the T5 encoder as follows (Figure 1): '`front:` $e_{n-n_<:n-1}$ `body:` $e_n$ `rear:` $e_{n+1:n+n_<}$.' Each utterance $e_*$ is tokenized by the T5 tokenizer. The T5 decoder is then trained to generate the gold transcription of the target transcribed utterance, $e_n$. In preliminary experiments, we compared the position of future transcribed utterances, between before and after the target transcribed utterance, and found that the latter performed consistently better.

## 4 EXPERIMENTAL SETUP

This section describes an experimental setup to evaluate our ASR error correction model. We applied our method to automatic transcriptions of CORAAL (Corpus of Regional African American Language) [8] datasets, in which transcriptions are obtained with ASR.

### *4.1 Data*

We used the CORAAL ASR dataset [8] with 231 sociolinguistic interviews of African Americans from diverse social backgrounds to create an ASR error correction dataset. We split each interview into utterances according to the gold transcriptions. The original CORRAL dataset contains speech data from interviews, which has been split into utterances on a speaker-turn basis. The utterances in this dataset are defined as uninterrupted speech sounds by the same individual, with utterances delimited at pause (60-70ms). Each utterance-based speech data accompanies a timestamp and a human transcription. For each utterance, we extracted past and future utterances by referring to the timestamps, while removing utterances whose transcriptions are annotated in the dataset as "inaudible," and "unintelligible" or can be regarded as fillers (hm, hmm, mm, mhm, mmm, uh, um, huh). Overlapping speeches are not removed, in order to consider contexts properly. We split each interview into train/development/test data in 9:0.5:0.5 rate, as described in Table 1.

| | train | dev. | test |
|---|---|---|---|
| # of interviews | 207 | 12 | 12 |
| # of utterances | 201,032 | 10,846 | 12,455 |
| ave. # of uttr. / interview | 971.17 | 903.83 | 1037.08 |
| ave. # of words / uttr. | 6.20 | 6.09 | 5.91 |
| hours | 122.37 | 5.90 | 7.33 |
| *transcriptions by NVIDIA STT Conformer-CTC Large* | | | |
| ave. # of words / utterances | 5.83 | 5.76 | 5.60 |
| WER | 27.70 | 27.30 | 26.50 |
| CER | 17.32 | 17.35 | 16.70 |

**Table 1** CORAAL-based ASR error-correction dataset.

### *4.2 Baseline ASR and error correction models*

As the target off-the-shelf ASR system for correction, we used NVIDIA STT Conformer-CTC Large. This model is a variation of Conformer [5], which uses CTC loss instead of Transducer in the encoder and its decoder is a linear decoder instead of a single layer of LSTM; it is learned with NeMo ASRSET dataset, which combines multiple English speech datasets (note: the source of our dataset, CORAAL, is not included). We fed each utterance (speech data) in the CORAAL dataset to NVIDIA Conformer-CTC Large to obtain ASR transcriptions for error correction, and then combined the ASR transcription for each utterance, its ASR-based contexts, and the gold transcription corresponding to the ASR transcription to train and test the ASR error correction model (Table 1).

We compared our method to a recent ASR error correction model, ConstDecoder [25] trained with the default settings,[2] and the above ASR with LM fusion,[3] which uses an LM[4] trained on the training data with default settings.

### *4.3 Training*

We adopted the pre-trained T5-base[5] as our base model and fine-tuned this model with the train split of our datasets. We fixed the model's hyperparameters to the same values as the T5-base. In the fine-tuning, we used a batch size of 32 and trained the models with Adafactor optimizer [15], with learning rate of $1e-4$, $\varepsilon_1 = 10^{-30}$, $\varepsilon_2 = 10^{-3}$, clipping threshold of 1.0, decay rate of $-0.8$, and weight decay of 0.0, while disabling relative step, scale parameter, and warmup init (these settings are

[2] https://github.com/yangjingyuan/ConstDecoder

[3] https://docs.nvidia.com/deeplearning/nemo/user-guide/docs/en/main/asr/asr_language_modeling.html

[4] https://github.com/kpu/kenlm

[5] https://huggingface.co/t5-base

| method (train/test contexts) | $n_<$ | $n_>$ | WER | CER |
|---|---|---|---|---|
| (input) | n/a | n/a | 26.50 | 16.70 |
| + LM-fusion | n/a | n/a | 26.33 | **16.58** |
| ConstDecoder | n/a | n/a | 27.95 | 20.80 |
| T5 | 0 | 0 | 24.87 | 17.02 |
| T5 (ASR/ASR) | 5 | 0 | 24.44 | 16.95 |
| T5 (ASR/ASR) | 10 | 0 | 24.31 | 16.83 |
| T5 (ASR/ASR) | 15 | 0 | 24.25 | 16.83 |
| T5 (ASR/ASR) | 0 | 5 | 24.39 | 16.86 |
| T5 (ASR/ASR) | 0 | 10 | 24.42 | 16.86 |
| T5 (ASR/ASR) | 0 | 15 | 24.40 | 16.84 |
| T5 (ASR/ASR) | 5 | 5 | 23.95 | 16.79 |
| T5 (ASR/ASR) | 10 | 10 | 23.89 | 16.82 |
| T5 (ASR/ASR) | 15 | 15 | **23.77** | 16.69 |
| T5 models trained or tested with gold transcriptions | | | | |
| T5 (gold/gold) | 15 | 15 | 22.90 | 16.30 |
| T5 (ASR/gold) | 15 | 15 | 22.72 | 16.07 |
| T5 (gold/ASR) | 15 | 15 | 24.56 | 17.14 |

**Table 2** Results of ASR error correction (CORAAL); the bottom three models are evaluated to measure the impact of using noisy ASR-based contexts for error correction. For example, T5 (ASR/gold) means that T5 trained with ASR-based contexts is evaluated on test datasets with gold contexts.

enabled in training randomly-initialized models in Table 5). We trained several T5 models while varying $n_<$ and $n_>$ from 0 to 15. For $n_< = n_> = 15$, we also trained models with gold transcriptions for contexts to see the upper bound of using contexts in ASR error correction.

## 5 RESULTS

We evaluated the effectiveness of T5 in ASR error correction (§ 5.1), the impact of pre-training (§ 5.2), the effectiveness of T5 in ASR error correction on proper nouns (§ 5.3), processing time (§ 5.4), and the performance in other datasets (§ 5.5). We computed word error rate (WER) and character error rate (CER) averaged over three runs with different random seeds.

### *5.1 Do cross-utterance contexts improve WER?*

Table 2 shows the WER and CER of ASR error correction on our CORAAL-based datasets when inputting the target ASR outputs with and without the ASR outputs

| |
|---|
| *T5 input*<br>`front`: even though i didn't do that with my tenth grade yet but i didn't do that with my tenth grade ...<br>`body`: i show that i can do it my *ninightware* yeaha i end this school with a three point two<br>*T5 output*<br>i showed that i can do it my **ninth grade** yes i ended this school with a three point two |
| *T5 input*<br>`body`: i had a nineteen thirty five *fod* at the time<br>`rear`: brand new car ... it was one of the first veight engines that the ford put out<br>*T5 output*<br>i had a nineteen thirty five **ford** at the time |
| *T5 input*<br>`front`: you can't remember when you pulled the trick on the teacher oh in the classroom ... we set the tash can on fire the teacher come in she took off her coat her coat goll burned up ...<br>`body`: i got *married* with this teacher<br>`rear`: and i asked one of the janitories for some water with some ware in the bucket ... it was a string on where the bucket was attached ... i asked him to pull the string down because it wouldn't come down and the water fell all over<br>*T5 output*<br>i got **mad** with this teacher |

**Table 3** Example outputs of ASR error correction.

| ASR transcription | T5 output |
|---|---|
| *toug of* (tougher) people | **a lot of** people |
| matter of fact she was *misteened* (ms teen) | matter of fact she was **miss** |
| start your *mode* (motor) and you drive | start your **manifest** and you drive |

**Table 4** Examples of over-correction. The results of speech recognition, correct transcription, and overcorrection by T5 are shown in italics, parentheses, and bold, respectively.

of past and future utterances. In the table, the row with $n_< = n_> = 0$ shows results of the T5-based ASR error correction only with the ASR output of the target utterance, which confirms the feasibility of correcting ASR outputs using T5. All the T5-based cross-utterance context-aware ASR error correction models ($n_<, n_> > 0$) outperformed base model $n_< = n_> = 0$. The baseline ConstDecoder failed to improve WER, which shows the difficulty of this dataset. It is also effective to use both past and future utterances and models with a larger number of context utterances perform better, especially for past utterances. We should emphasize that T5 can reduce WER more with longer contexts, even with ASR-based contexts.

Table 3 exemplifies the model's corrections. These three examples show corrections using the past, future, and both utterances, respectively. In the first example, the model successfully corrected *ninightware* into **ninth grade**, despite the poor ASR result. In the second example, the model used future utterances to correct *fod* into **ford**, the automobile company. In the last example, full context is used and the incoherent word *married* is corrected into **mad** correctly.

| method | $n_<$ | $n_>$ | WER | CER |
|---|---|---|---|---|
| (input) | n/a | n/a | 26.50 | 16.70 |
| T5 (random) | 0 | 0 | 27.57 | 18.88 |
| T5 (random) | 5 | 5 | 27.33 | 19.05 |
| T5 (random) | 10 | 10 | 27.38 | 18.99 |
| T5 (random) | 15 | 15 | 27.55 | 18.98 |

**Table 5** ASR error correction with random initialization.

| method (train/test contexts) | $n_<$ | $n_>$ | $\text{WER}_{\text{pronN}}$ |
|---|---|---|---|
| (input) | n/a | n/a | 30.59 |
| T5 | 0 | 0 | 26.48 |
| T5 (ASR/ASR) | 5 | 0 | 26.42 |
| T5 (ASR/ASR) | 10 | 0 | 25.69 |
| T5 (ASR/ASR) | 15 | 0 | 25.74 |
| T5 (ASR/ASR) | 0 | 5 | 26.37 |
| T5 (ASR/ASR) | 0 | 10 | 26.46 |
| T5 (ASR/ASR) | 0 | 15 | 26.50 |
| T5 (ASR/ASR) | 5 | 5 | 25.95 |
| T5 (ASR/ASR) | 10 | 10 | **25.47** |
| T5 (ASR/ASR) | 15 | 15 | 25.51 |

**Table 6** Results of ASR error correction (CORAAL). $\text{WER}_{\text{pronN}}$ lists WER on utterances that contain proper nouns.

Meanwhile, the CER of the correction was almost the same as the ASR outputs. The aggressive correction by T5 may reduce letter-wise similarity since T5 does not utilize acoustic features but only text information to correct transcriptions. We noticed several over-corrections by T5, as shown in Table 4.

## *5.2 Is the pre-training required to improve WER?*

To see the importance of the pre-training, we randomly re-initialized parameters in the pre-trained T5 model and trained the model from scratch on the downstream task. As shown in Table 5, it did not improve WER from the inputs (original ASR outputs). This reconfirms the importance of pre-training reported for BART-based ASR error correction on Chinese datasets [28]. Note that we cannot obtain better results even with cross-utterance contexts. The use of the pre-trained model is crucial to exploit longer cross-utterance contexts to decrease WER.

| $n_<$ | $n_>$ | Inference time per batch (sec.) |
|---|---|---|
| 0 | 0 | 276.502 ($\pm$89.461) |
| 5 | 0 | 301.130 ($\pm$104.892) |
| 10 | 0 | 323.655 ($\pm$103.556) |
| 15 | 0 | 348.090 ($\pm$115.295) |
| 5 | 5 | 323.457 ($\pm$106.181) |
| 10 | 10 | 363.729 ($\pm$111.727) |
| 15 | 15 | 399.280 ($\pm$116.274) |

**Table 7** Inference time of ASR error correction per batch of 32 utterances.

### *5.3 Is T5-based correction effective for proper nouns?*

As stated in § 3, we expect that using the T5 pre-trained model trained with a massive amount of text will contribute to correcting transcriptions of proper nouns rarely or not appeared in ASR training data but in T5 pre-training data. To confirm this, we applied a part-of-speech tagger from Flair [1][6] to gold transcriptions to identify utterances with proper nouns, and computed WER only on these utterances. As a result, T5-based ASR error correction with $n_< = n_> = 15$ greatly improved the WER of the utterances with proper nouns by 5.12 (30.59 $\rightarrow$ 25.47). as shown in Table 6.

### *5.4 How efficient to correct ASR outputs in inference?*

Table 7 lists the average inference time per batch, the size of which is 32. Although there is a clear increase in the inference time as the number of cross-utterance contexts increases, the relative increase ratio against T5 without cross-utterance contexts is moderate. This is because the T5 (Transformer) consumed much time in auto-regressive decoding, and it takes most of the inference time.

### *5.5 Is the proposed method effective for other datasets?*

We have applied our method to ASR transcriptions of the AMI-IHM dataset,[7] which comprises 100 hours of meeting recordings, in the same settings as stated in § 4.3. The details of the ASR error-correction dataset are listed in Table 8. As shown in the results in Table 9, we confirmed that the results show the same trend as the experiment on CORAAL.

[6] `https://github.com/flairNLP/flair`

[7] `https://groups.inf.ed.ac.uk/ami/corpus/`

| | train | dev. | test |
|---|---|---|---|
| # of meetings | 137 | 18 | 16 |
| # of utterances | 99,536 | 12,151 | 11,636 |
| ave. # of uttr. / meeting | 726.54 | 675.06 | 727.25 |
| ave. # of words / uttr. | 7.65 | 7.42 | 7.35 |
| hours | 77.89 | 8.94 | 8.68 |
| *transcriptions by NVIDIA STT Conformer-CTC Large* | | | |
| ave. # of words / utterances | 7.58 | 7.42 | 7.29 |
| WER | 17.61 | 16.92 | 17.44 |
| CER | 9.29 | 9.63 | 9.38 |

**Table 8** AMI-IHM-based ASR error-correction dataset.

| method (train/test contexts) | $n_<$ | $n_>$ | WER | CER |
|---|---|---|---|---|
| (input) | n/a | n/a | 17.44 | 9.38 |
| + LM-fusion | n/a | n/a | 17.26 | 9.14 |
| ConstDecoder | n/a | n/a | 16.80 | 11.03 |
| T5 | 0 | 0 | 15.09 | 9.00 |
| T5 (ASR/ASR) | 5 | 0 | 14.87 | 9.00 |
| T5 (ASR/ASR) | 10 | 0 | 14.90 | 9.01 |
| T5 (ASR/ASR) | 15 | 0 | 14.85 | 8.94 |
| T5 (ASR/ASR) | 0 | 5 | 14.87 | 8.96 |
| T5 (ASR/ASR) | 0 | 10 | 14.90 | 8.99 |
| T5 (ASR/ASR) | 0 | 15 | 14.83 | 8.95 |
| T5 (ASR/ASR) | 5 | 5 | 14.72 | 8.89 |
| T5 (ASR/ASR) | 10 | 10 | 14.69 | 8.91 |
| T5 (ASR/ASR) | 15 | 15 | **14.67** | **8.89** |
| T5 models trained or tested with gold transcriptions | | | | |
| T5 (gold/gold) | 15 | 15 | 14.48 | 8.79 |
| T5 (ASR/gold) | 15 | 15 | 14.51 | 8.78 |
| T5 (gold/ASR) | 15 | 15 | 14.97 | 8.99 |

**Table 9** Results of ASR error correction (AMI-IHM).

## 6 CONCLUSIONS

In this study, we have proposed a cross-utterance context-aware error correction model for off-the-shelf ASR systems. We propose to use the pre-trained Transformer-based text generation model, T5, as a backbone of our model, and feed cross-utterance contexts as additional input to T5. Experimental results on the CORAAL-based ASR error correction datasets transcribed with NVIDIA STT Conformer-CTC confirmed the advantage of our T5-based ASR error correction; $-2.73$ ($26.50 \rightarrow 23.77$) in WER for all utterances and $-5.12$ ($30.59 \rightarrow 25.47$) in WER for the utterances with proper nouns. We observed more improvements in WER when we used longer contexts for error correction. Even though the contexts are ASR-based and

noisy, the fine-tuned T5 successfully utilized those noisy contexts to correct the target ASR description. Since our model is a purely text-based model, it can be used to improve the quality of existing ASR transcriptions in the recorded multimedia data.

In the future, we plan to improve the CER of our error correction model, by using a loss function that considers the acoustic information of the target ASR outputs for correction.

## References


1. Akbik, A., Blythe, D., Vollgraf, R.: Contextual string embeddings for sequence labeling. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1638–1649. Association for Computational Linguistics, Santa Fe, New Mexico, USA (2018). URL https://aclanthology.org/C18-1139
2. Bawden, R., Sennrich, R., Birch, A., Haddow, B.: Evaluating discourse phenomena in neural machine translation. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 1304–1313. Association for Computational Linguistics, New Orleans, Louisiana (2018). DOI 10.18653/v1/N18-1118. URL https://aclanthology.org/N18-1118
3. Chollampatt, S., Wang, W., Ng, H.T.: Cross-sentence grammatical error correction. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 435–445. Association for Computational Linguistics, Florence, Italy (2019). DOI 10.18653/v1/P19-1042. URL https://aclanthology.org/P19-1042
4. Dutta, S., Jain, S., Maheshwari, A., Pal, S., Ramakrishnan, G., Jyothi, P.: Error Correction in ASR using Sequence-to-Sequence Models (2022). DOI 10.48550/arXiv.2202.01157. URL http://arxiv.org/abs/2202.01157. ArXiv:2202.01157 [cs]
5. Gulati, A., Qin, J., Chiu, C.C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., Pang, R.: Conformer: Convolution-augmented Transformer for Speech Recognition. In: Proc. Interspeech 2020, pp. 5036–5040 (2020). DOI 10.21437/Interspeech.2020-3015. URL http://dx.doi.org/10.21437/Interspeech.2020-3015
6. Guo, J., Sainath, T.N., Weiss, R.J.: A Spelling Correction Model for End-to-end Speech Recognition. In: ICASSP 2019. IEEE, Brighton, United Kingdom (2019). DOI 10.1109/ICASSP.2019.8683745. URL https://ieeexplore.ieee.org/document/8683745/
7. Hrinchuk, O., Popova, M., Ginsburg, B.: Correction of Automatic Speech Recognition with Transformer Sequence-To-Sequence Model. In: ICASSP 2020. Barcelona, Spain (2020). DOI 10.1109/ICASSP40776.2020.9053051. URL https://ieeexplore.ieee.org/document/9053051/
8. Kendall, T., Farrington, C.: The Corpus of Regional African American Language. Version 2021.07. Eugene, OR: The Online Resources for African American Language Project (2021). URL http://oraal.uoregon.edu/coraal
9. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (2020). DOI 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703
10. Ma, R., Gales, M.J.F., Knill, K.M., Qian, M.: N-best T5: Robust ASR Error Correction using Multiple Input Hypotheses and Constrained Decoding Space (2023). DOI 10.48550/arXiv.2303.00456. URL http://arxiv.org/abs/2303.00456. ArXiv:2303.00456 [cs, eess]

11. Mani, A., Palaskar, S., Meripo, N.V., Konam, S., Metze, F.: ASR Error Correction and Domain Adaptation Using Machine Translation. In: ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Barcelona, Spain (2020). DOI 10.1109/ICASSP40776.2020.9053126. URL https://ieeexplore.ieee.org/document/9053126/
12. Maruf, S., Martins, A.F.T., Haffari, G.: Selective attention for context-aware neural machine translation. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)), pp. 3092–3102 (2019). DOI 10.18653/v1/N19-1313. URL https://www.aclweb.org/anthology/N19-1313
13. Masumura, R., Ihori, M., Tanaka, T., Saito, I., Nishida, K., Oba, T.: Generalized Large-Context Language Models Based on Forward-Backward Hierarchical Recurrent Encoder-Decoder Models. In: 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) (2019). DOI 10.1109/ASRU46091.2019.9003857
14. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020)
15. Shazeer, N., Stern, M.: Adafactor: Adaptive learning rates with sublinear memory cost. In: J. Dy, A. Krause (eds.) Proceedings of the 35th International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 80, pp. 4596–4604. PMLR (2018). URL https://proceedings.mlr.press/v80/shazeer18a.html
16. Sugiyama, A., Yoshinaga, N.: Data augmentation using back-translation for context-aware neural machine translation. In: Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019), pp. 35–44. Association for Computational Linguistics, Hong Kong, China (2019). DOI 10.18653/v1/D19-6504. URL https://www.aclweb.org/anthology/D19-6504
17. Sugiyama, A., Yoshinaga, N.: Context-aware decoder for neural machine translation using a target-side document-level language model. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5781–5791. Association for Computational Linguistics, Online (2021). DOI 10.18653/v1/2021.naacl-main.461. URL https://aclanthology.org/2021.naacl-main.461
18. Sun, G., Zhang, C., Woodland, P.C.: Cross-Utterance Language Models with Acoustic Error Sampling (2020). DOI 10.48550/arXiv.2009.01008. URL http://arxiv.org/abs/2009.01008
19. Tanaka, T., Masumura, R., Masataki, H., Aono, Y.: Neural Error Corrective Language Models for Automatic Speech Recognition. In: Interspeech 2018. ISCA (2018). DOI 10.21437/Interspeech.2018-1430. URL https://www.isca-speech.org/archive/interspeech_2018/tanaka18_interspeech.html
20. Tiedemann, J., Scherrer, Y.: Neural machine translation with extended context. In: Proceedings of the Third Workshop on Discourse in Machine Translation, pp. 82–92. Association for Computational Linguistics, Copenhagen, Denmark (2017). DOI 10.18653/v1/W17-4811. URL https://aclanthology.org/W17-4811
21. Tu, Z., Liu, Y., Shi, S., Zhang, T.: Learning to remember translation history with a continuous cache. Transactions of the Association of Computational Linguistics (TACL) **6**, 407–420 (2018). DOI 10.1162/tacl_a_00029. URL https://www.aclweb.org/anthology/Q18-1029
22. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 5998–6008 (2017). URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf
23. Wang, H., Dong, S., Liu, Y., Logan, J., Agrawal, A.K., Liu, Y.: ASR Error Correction with Augmented Transformer for Entity Retrieval. In: Interspeech 2020, pp. 1550–1554. ISCA (2020). DOI 10.21437/Interspeech.2020-1753. URL https://www.isca-speech.org/archive/interspeech_2020/wang20p_interspeech.html

24. Wang, L., Tu, Z., Way, A., Liu, Q.: Exploiting cross-sentence context for neural machine translation. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2826–2831 (2017). DOI 10.18653/v1/D17-1301. URL https://www.aclweb.org/anthology/D17-1301
25. Yang, J., Li, R., Peng, W.: ASR Error Correction with Constrained Decoding on Operation Prediction. In: Interspeech (2022). DOI 10.21437/Interspeech.2022-660. URL https://www.isca-speech.org/archive/interspeech_2022/yang22g_interspeech.html
26. Yu, L., Sartran, L., Stokowiec, W., Ling, W., Kong, L., Blunsom, P., Dyer, C.: Better document-level machine translation with Bayes' rule. Transactions of the Association for Computational Linguistics **8**, 346–360 (2020). DOI 10.1162/tacl_a_00319. URL https://aclanthology.org/2020.tacl-1.23
27. Zhang, F., Tu, M., Liu, S., Yan, J.: ASR Error Correction with Dual-Channel Self-Supervised Learning. In: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE (2022). DOI 10.1109/ICASSP43922.2022.9746763
28. Zhao, Y., Yang, X., Wang, J., Gao, Y., Yan, C., Zhou, Y.: BART based semantic correction for Mandarin automatic speech recognition system. In: Interspeech 2021 (2021). DOI 10.21437/Interspeech.2021-739