SIGNIFICANT ASR ERROR DETECTION FOR CONVERSATIONAL VOICE ASSISTANTS

John Harvill[1*], *Rinat Khaziev*[2], *Scarlett Li*[2], *Randy Cogill*[2], *Lidan Wang*[2], *Gopinath Chennupati*[2], *Hari Thadakamalla*[2]

[1]University of Illinois Urbana-Champaign, USA
[2]Amazon Alexa AI, USA

ABSTRACT

Modern Automatic Speech Recognition (ASR) systems are evaluated with respect to Word Error Rate (WER). While WER is a useful metric for training and evaluation of speech models, it does not fully reflect the difference in semantics between predicted and ground truth transcriptions. In conversational voice assistants, the ability to sufficiently understand semantic meaning of the user request is often more important than recognizing all words correctly. In this work, we propose a system that can determine, to a high degree of accuracy, whether the semantics of a predicted and reference transcript are significantly different. This knowledge is used to identify ASR errors that can result in downstream failure in conversational voice assistants. Reliable identification of these errors can be used to inform design choices for ASR systems targeting improvement on the most harmful errors.

***Index Terms*—** ASR, Voice Assistants, Weak Labeling, Semantics, Large Language Models

1. INTRODUCTION

Conversational Voice Assistants (CVAs) became ubiquitous in everyday life and are capable of performing a variety of tasks for a user through simple voice commands. These include setting timers and reminders, providing weather reports, sending emails/texts, shopping and answering questions. Automatic Speech Recognition (ASR) [1, 2, 3] is an important component of the CVA that processes a user request. Significant defects in the speech recognition system are challenging to recover from in the downstream systems and often lead to defective interactions between user and a CVA [4].

In this work, we focus on evaluating performance of the ASR system in terms of its ability to produce a semantically sufficient transcription of the user speech. We introduce a new task of identifying semantically significant errors in the ASR subsystem only, which we call Significant ASR Error Detection (SASRED). For the SASRED task, a positive label indicates that an ASR error has been made such that the semantics of the predicted transcript are significantly different from that of the intended speech.[1] The negative label indicates either no error or a semantically insignificant ASR error (See Figure 1).

Our proposed approach takes advantage of several recent advances from both the fields of Natural Language Processing (NLP) and Machine Learning (ML). The first component in our system is an encoder (SimCSE) that produces vector representations of semantics at the sentence level [5]. We use SimCSE to measure the semantic difference between predicted and reference transcripts for a given audio input, which is then used to compute weak labels for the SASRED task in a self-supervised fashion. We construct a training dataset using a combination of weak and human labels to distill knowledge to a Large Language Model (LLM) via Parameter-Efficient Fine-Tuning (PEFT) [6, 7, 8]. We find that the combination of both weak labeling and transfer learning are critical for the best performance on the SASRED task.

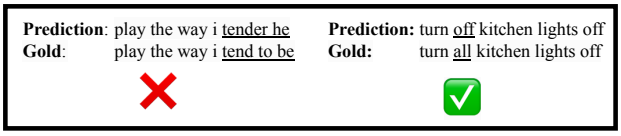

Fig. 1: Significant (left) vs. insignificant (right) ASR errors. The example on the left is a significant error, because the title of the requested song is transcribed incorrectly, making it impossible for the CVA to do what the user intended. The example on the right is insignificant, because the semantics of the intent to turn off the kitchen lights remains intact despite the error in the predicted transcript.

Such an approach is different from evaluating ASR models using deterministic metrics, like Word Error Rate (WER) or Sentence Error Rate (SER), since those approaches do not directly capture semantic differences between predicted and ground truth transcriptions. We aim to build a system that can reliably identify incorrectly predicted transcriptions that need the most improvements. Our model can be used for evaluating ASR models based on ground truth transcription or transcription provided by a larger and more performant model than that used for the original transcript prediction.

In this paper, we make several contributions: **1)** Introduce the SASRED task and motivate its significance. **2)** Provide experimental results on SASRED using a previously published Failure Point Isolation [4] approach. **3)** Propose an effective semi-supervised machine learning system that significantly outperforms previous methods on SASRED and requires only a few hundred annotated data. **4)** Perform a variety of ablation studies that give insight into why each component of our proposed system is necessary for best performance.

2. RELATED WORK

Detection of the ASR defects was first proposed by Khaziev et al. [4] for the generalized defect attribution system in Conversational Voice Assistants as part of the Failure Point Isolation (FPI) task. In their work, the authors proposed an approach that demonstrated promising results across a variety of subsystem classes including detection defects in trigger word (wake word), ASR, NLU and Entity Recognition/Resolution (ERR) systems using a transformer-based model. FPI model uses the text from the dialog session between the

∗Work completed as part of an internship with Amazon Alexa AI.

[1]In this paper, a semantically significant difference is one where the difference is judged by human annotators to be extremely likely to result in a downstream error in the CVA.

user and CVA, as well as various categorical and numerical features extracted from session metadata as input features. The entire system is trained end-to-end on a dataset of several million dialog sessions and achieves performance close to that of non-expert human annotators. Unlike Khaziev et al. [4], we focus on detecting significant errors in the ASR system only without considering overall success of the interaction with the CVA. While the methods proposed in this paper do not generalize to FPI detection for other CVA subsystems, our work demonstrates significant advances over previous techniques for the ASR class and thus deserves its own dedicated discussion.

The ASR error correction task [9, 10, 11] is well-studied and related to our proposed task in that it processes predicted ASR transcripts using only text features as input. The goal of ASR error correction is to further improve transcription quality by reducing Word Error Rate (WER) directly with dedicated domain-specific language models. In contrast, the goal of our proposed task is to identify those dialog sessions whose transcript errors are significant for the purpose of motivating design changes in the next iteration of the production ASR model.

Another related task is that of ASR error detection [12], where the goal is to determine where errors occur in the predicted transcript. Gekhman et al. [12] integrate word-level confidence scores from the ASR system with a bi-directional language model using confidence embeddings and find notable improvement compared to using the bi-directional language model alone. While ASR error detection can be useful for detecting potential errors in the transcript, it gives no indication as to whether the error is semantically significant and could lead to downstream failure in a CVA.

3. METHODS

We propose a three-step approach for the SASRED task that takes advantage of self-supervised and supervised systems, and find that all parts are necessary to achieve the best performance. The first step is weak labeling, where we create confident weak labels using a pretrained sentence embedding model. The second step is the deterministic construction of error explanations that are used for chain-of-thought prompting [13]. The final step involves transfer learning to an LLM by training with weakly labeled data and a small dataset with human annotations. The only input features used by our approach are the predicted and reference ASR transcripts.

3.1. Weak Labeling

We rely on a critical assumption for weak labeling: As the semantic difference between the predicted and reference transcript grows, so does the likelihood of a fatal ASR error (See Figure 1). To measure the semantic difference between the two transcripts, we use the sentence embedding model called SimCSE[2] [5]. This model is trained to map sentences to an embedding space such that the distance between two sentences with similar meaning is minimized. We use cosine distance d between the SimCSE embeddings of the predicted and reference transcripts as a basis for creating weak labels. The weak labels are then chosen such that sessions with d above a threshold T are classified as significant ASR errors and those with $d \leq T$ are insignificant. We determine the optimal value of T by searching in increments of 0.01 from 0 to 1 on each fold of the labeled training data and computing F1 scores.

Label Confidence. By empirical examination of some examples, we found that the confidence of the weak label is correlated with d for a dialog session. In cases where $d < 0.05$, almost all sessions have insignificant errors. Likewise, for $d > 0.20$, most sessions contain significant errors. We use this information to split training data into the set of Confident Weak Labels (CWL) and Unconfident Weak Labels (UWL). Those weakly labeled sessions with $0.05 \leq d \leq 0.20$ fall into UWL and the rest fall into CWL (see validation of the labels in Section 5). These two sets are used for different experimental settings to demonstrate the importance of using confident weak labels and human annotated data.

[2]https://huggingface.co/princeton-nlp/sup-simcse-roberta-large

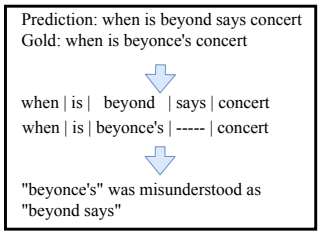

Fig. 2: Process for constructing ASR explanations. First, transcripts are aligned at the word level. Then, aligned sections whose strings do not match are collected. The sections are converted to strings by joining each element of a given section by spaces, where blanks ("——") are removed, i.e. [beyonce's,——] → "beyonce's" and [beyond,says] → "beyond says" in the above example. Finally, the strings for each collected section are placed into the template: "<REF>" was misunderstood as "<PRED>".

3.2. Construction of Explanations

Chain-of-Thought (CoT) prompting [13] is a recently proposed technique that improves the ability of LLMs to solve math word problems and perform commonsense and symoblic reasoning. Instead of directly prompting an LLM to predict a label, CoT predictions provide logical steps leading up to the decision and then produce the label prediction. In this paper, we explored what effect CoT prompting has on the SASRED task. Given that ASR errors can be explained perfectly by noting discrepancies between the predicted and reference transcripts, we can construct explanations for ASR errors algorithmically using a template. To extract discrepancies automatically, we perform Needleman-Wunsch alignment at the word level between predicted and reference transcripts, and pull out sections that are aligned but whose strings are not equal. We then construct CoT explanations by copying extracted sections into a template (See Figure 2).

3.3. Transfer Learning to LLM

Given the labels and error explanations, we can fine-tune an LLM to perform classification via text generation [14] using the PEFT technique called Low Rank Adaptation [8] (LoRA). We do so by constructing a string with inputs followed by outputs, where the inputs are the predicted and reference transcripts, and the outputs are the explanation and label (See Figure 3). We fine-tune the LLM using cross-entropy loss with teacher forcing on the entire string. At inference time, only the predicted and reference transcripts are provided to the model and generation is continued auto-regressively by choosing the most likely token at each timestep (argmax decoding) until the end-of-sentence token is produced. The structure of generating the explanation first and then the label is learned perfectly during finetuning such that parsing the label from the generated output is straightforward (See Figure 3).

Original transcript: when is beyond says concert
Correct transcript: when is beyonce's concert

Misheard speech: "beyonce's" was misunderstood as "beyond says"

Label prediction: ASR

Fig. 3: String input and output format for LLM training/prediction. The predicted ASR transcript is preceded by the prompt text "Original transcript:" and the reference transcript is preceded by "Correct transcript:". If the above example were used for training, the entire string (black + blue text) would be used to compute the cross-entropy loss. If the example above were held out for evaluation, only the black text is provided at inference time, and the blue text would ideally be generated. The label is then parsed from the generated text by splitting on the string "Label prediction:" and choosing the last element in the split.

4. BASELINES

We use ASR defect predictions from the system proposed by Khaziev et al. [4] as one of the baselines in this work. We evaluate this pre-trained model on the test set used in our experiments as is. While our proposed methods only use a small input feature set, FPI baseline uses all features discussed in the original paper [4].

As a second baseline, we use a classifier similar to the weak labeler (see Section 3.1) based on Word Error Rate (WER). WER is evaluated between the predicted and reference transcripts as the semantic distance feature d. Though, this is a crude measurement of semantic similarity, this baseline demonstrates importance of supervised training for learning semantic representations for our task.

5. DATASET

We construct our train and test datasets from a sample of approximately 94k anonymized (de-identified) user interactions (or dialog sessions) with Amazon Alexa. Each interaction contains the predicted ASR transcript and a reference transcript. We create high-quality labels for 746 dialog sessions (gold labels) using an internal annotation pipeline for three data bins based on d: $d < 0.05$ - Low, $0.05 \leq d < 0.20$ - Medium, and $0.20 \leq d$ - High. Additionally, we add 300 "no-difference" data $d = 0$ (see Table 1) to ensure that fine-tuned models correctly identify "no-error" cases, since we find empirically that the LLM-based methods incorrectly classify a few examples in this category.

The data with $d > 0$ was annotated for two labels: "Significant ASR Error" and "No Significant Error". In this work, we considered semantically different errors in action words, mis-recognition of entities, interrogative adverbs, etc, as "Significant ASR Errors". "Insignificant Error" include mistakes in recognizing plural vs singular nouns, missing articles, insignificant nouns, and semantically similar action verbs, etc. We refer to "Significant ASR Errors" as Fatal (significant or fatal difference) and "No Significant Error" as "Non-fatal" throughout the paper.

Based on annotated data, we observe that label confidence of the weak labeler (SimCSE) is heavily correlated with the distance feature d [3] (See Figure 4). We confirm that the weak labelling scheme described in 3.1 has relatively high performance: Recall of 1.0 in low and high confidence bins, Precision of 0.88 for Non-fatal errors in

[3]Dialog sessions with small or large d are almost always non-fatal or fatal, respectively, and are on average less difficult datapoints.

Label	Dataset bin			
	No Diff.	Low	Medium	High
Fatal ASR Error	0	17	225	124
Non-fatal ASR Error	300	121	251	8
Total	300	138	476	132

Table 1: Annotated dataset label distribution by bin. For each bin, data are randomly sampled from the bin and then annotated such that the low percentanges of Fatal and Non-fatal labels in Low and High, respectively, represent the natural distribution.

Low distance bin and 0.95 for fatal errors in High distance bin. Based on this split, the data in Medium is, on average, most difficult to classify properly using d and the models trained in this work, which is validated in our experiments.

5.1. Cross-validation

Our proposed methods rely on both weakly and human (gold) labeled data for training. We construct a 3-fold cross-validation split by balancing both human and weakly labeled data in each fold. For each training split, we use 2/3 of the human labeled data[4] along with a balanced split of Fatal/Non-fatal weakly labeled sessions, where no test data are included in the weakly labeled training data. Since our random data sample contains many other FPI labels (Wakeword, NLU), ASR errors do not make up a large portion of the dataset. Balancing Fatal (ASR) and Non-fatal labels evenly thus results in a reduction in training dataset size. Of the several training configurations we explore in this paper, the largest training dataset consists of approximately 10k sessions. Our test set for each fold contains the human labeled data not used in training such that our evaluation is only performed with respect to human labels (1/3 of annotated data per fold). The results in this paper are reported across all three folds such that every human annotated session is included in evaluation.

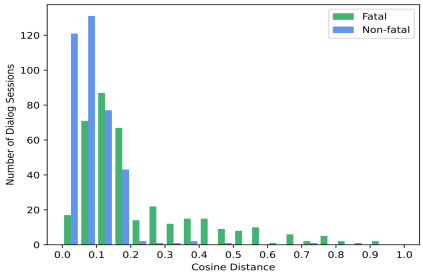

Fig. 4: Histogram of weak label distances for annotated dialog sessions (excluding "No Difference" data).

6. EXPERIMENTS

We evaluate our proposed approach with a variety of training settings that highlight the importance of weakly labeled data, human annotations and error explanations. We compare to the performance of the baselines ([4] and WER-based classifier) and the weak labeler (SimCSE) on all three folds of the test set. We describe the various training settings and discuss the purpose of each setting below.

1) Confident Weak Labels and Gold Labels. This setting highlights the benefit of excluding sessions where d is close to the decision boundary (T) and thus produces a noisy label.

2) Removing Error Explanations. This ablation is identical to Setting #1 but without chain-of-thought error explanations.

[4]We split equally per distance bin (2/3 No Diff., 2/3 Low, etc.).

Method	Training Data	Precision / Recall / F1							
		Low		Medium		High		All	
		Fatal	Non-fatal	Fatal	Non-fatal	Fatal	Non-fatal	Fatal	Non-fatal
Baselines									
FPI [4]	N/A	.67/.24/.35	.90/.98/.94	.78/.27/.40	.59/**.93**/.72	.96/.44/.61	.08/**.75**/.14	.84/.33/.47	.73/**.97**/.83
WER	GL	.12/.71/.20	.86/.26/.39	.47/**.92**/.62	.50/.07/.12	.94/.98/.96	0.0/0.0/0.0	.51/**.93**/.66	**.93**/.51/.66
Weak Labeler									
SimCSE	GL	0.0/0.0/0.0	.88/**1.0**/.93	.57/.63/.60	.64/.59/.61	.94/**1.0**/**.97**	0.0/0.0/0.0	.70/.73/.72	.85/.84/.84
Fine-tuned Models									
1. Falcon$_{7B}$	CWL + GL	.56/.29/.38	.91/.97/.94	.70/.77/.74	**.78**/.71/.74	**.97**/.94/.96	**.36**/.50/.42	.77/.81/.79	.89/.87/.88
2. - explan.	CWL + GL	.43/.53/.47	.93/.90/.92	.64/.73/.68	.72/.63/.67	.94/.96/.95	0.0/0.0/0.0	.72/.80/.76	.88/.83/.86
3. Falcon$_{7B}$	CWL	.22/.24/.23	.89/.88/.89	.58/.72/.64	.68/.53/.59	.94/.94/.94	.13/.13/.13	.67/.77/.72	.87/.80/.83
4. Falcon$_{7B}$	GL	.35/**.82**/.49	**.97**/.79/.87	.62/.83/.71	**.78**/.55/.65	.95/.82/.88	.12/.38/.18	.56/.83/.67	.87/.66/.75
5. Falcon$_{7B}$	CWL + UWL	.63/.29/.40	.91/.98/.94	.68/.67/.67	.71/.71/.71	.96/.94/.95	.27/.38/.32	.76/.74/.75	.86/.87/.87
6. Falcon$_{7B}$	CWL (PT), GL (FT)	**.80**/.47/**.59**	.93/.98/**.96**	**.79**/.75/**.77**	**.78**/.82/**.80**	**.97**/.93/.95	**.36**/.63/**.45**	**.85**/.80/**.82**	.89/.92/**.91**

Table 2: Results organized by bin, where each cell in the table shows precision/recall/F1. Setting numbers from Section 6 are given for fine-tuned models. Abbreviations: CWL - Confident Weak Labels, GL - Gold Labels, UWL - Unconfident Weak Labels, PT - Pretrain, FT - Finetune, WER - Word Error Rate. "All" includes data from "No Difference" bin (not shown in this table, see Section 7).

3) Confident Weak Labels Only. The goal of this experiment is to demonstrate the performance gap from Setting #1 when no human annotated data is available for training and we rely solely on transfer learning from the weak labeler.

4) Gold Labels Only. This experiment shows the importance of having a large amount of weakly labeled data available for training.

5) Confident and Unconfident Weak Labels. This setting demonstrates the importance of using a small amount of human labeled data instead of noisy unconfident weak labels for training.

6) Separate Pre-training and Fine-tuning Steps. For this experiment, we first pretrain with weakly labeled data only (CWL) and then finetune only on human annotated data, i.e. Gold Labels (GL). This uses the same training data as Setting #1 but separates training stages by label type (weak label vs. human annotation).

6.1. Models and Hyperparameters

We use the 7B-parameter version of Falcon [15] for our experiments. For each finetuning run, we use a batch size of eight and learning rate of 3e-4. We find that validation loss after one epoch plateaus and that some instability can occur when training longer, so we report results for all models after training for one epoch.[5] Due to our use of three-fold cross-validation and the expensive computational budget of finetuning LLMs, we run each fold once.

7. RESULTS

The experimental results[6] are given in Table 2. We discuss the key takeaways below.

Chain-of-Thought. When comparing Setting #1 (with error explanations) to Setting #2 (without explanations), we see that using explanations leads to noticeable improvement of F1 scores for both Fatal and Non-fatal classes overall ("All" columns) and for the most difficult examples (Medium bin). This demonstrates that chain-of-thought prompting is effective for the SASRED task.

Optimal Training Data. The best overall F1 scores for both classes come from Settings #1 and #6, which both use the combination of confident weak labels and gold labels for training. When comparing to Settings #3 and #4, which remove either the gold labels or confident weak labels, respectively, we see a degradation in performance. This shows that a combination of a large amount of weakly labeled data and a small amount of high-quality annotated data is necessary for optimal performance.

Separate Fine-tuning Stages. The only difference between the best two methods is whether all data are combined during one training stage (Setting #1), or if training is separated by first pretraining on weakly labeled data and then finetuning on human annotated data (Setting #6). The best overall F1 scores come from Setting #6, which demonstrates that it is best for the final training steps to be performed with high-quality data only.

Optimal Method. Our proposed transfer learning approach significantly outperforms the weak labeler (SimCSE) and the baselines. The best method, Setting #6, outperforms the weak labeler by close to 0.2 F1 on Medium data and close to 0.1 F1 overall for Fatal and Non-fatal classes. When comparing to the FPI baseline (Khaziev et al. [4]), the biggest difference is that of recall for Fatal errors. While our proposed system outperforms the baseline on precision for both classes, it significantly improves recall for the Fatal class (0.33→0.80 in "All").

8. CONCLUSIONS

In this paper, we introduced the Significant ASR Error Detection (SASRED) task, where the goal is to identify ASR transcriptions whose semantics are significantly different compared to the reference transcription. Unlike Word Error Rate (WER), which does not account for semantics, SASRED makes it possible to focus on ASR errors that are most likely to lead to downstream CVA failure, which could guide design improvements to an existing ASR system. We proposed an effective solution to the SASRED task that uses weak labeling, chain-of-thought prompting, and transfer learning to achieve significant performance gains over several baselines. Across multiple ablation settings, we achieve the best performance when using chain-of-thought prompting, training with a dataset of confident weak labels and human labels, and separating pretraining and finetuning stages based on label quality (weak vs. human labels). Future work could explore the tradeoff of using human transcripts or those from a large, offline ASR model to create training annotations for defective CVA interactions automatically.

[5] We train for three epochs for Setting #4 (Gold Labels Only). The training dataset is significantly smaller than those for the other settings and the validation loss takes longer to flatten.

[6] We exclude the "No Difference" bin from Table 2 due to space constraints and near perfect performance for all but one method. Except for Setting #4, the lowest value of precision, recall, and F1 is 0.97. Setting #4 achieves 1.0/0.70/0.82 for Non-fatal errors (scores are not defined for Fatal because there are no datapoints with that label in the "No Difference" bin.)

9. REFERENCES

[1] Mohamed Benzeghiba, Renato De Mori, Olivier Deroo, Stephane Dupont, Teodora Erbes, Denis Jouvet, Luciano Fissore, Pietro Laface, Alfred Mertins, Christophe Ris, et al., "Automatic speech recognition and speech variability: A review," *Speech communication*, vol. 49, no. 10-11, pp. 763–786, 2007.

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

[3] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[4] Rinat Khaziev, Usman Shahid, Tobias Röding, Rakesh Chada, Emir Kapanci, and Pradeep Natarajan, "Fpi: Failure point isolation in large-scale conversational assistants," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, 2022, pp. 141–148.

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

[6] Brian Lester, Rami Al-Rfou, and Noah Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic, Nov. 2021, pp. 3045–3059, Association for Computational Linguistics.

[7] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang, "P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Dublin, Ireland, May 2022, pp. 61–68, Association for Computational Linguistics.

[8] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[9] Anirudh Mani, Shruti Palaskar, Nimshi Venkat Meripo, Sandeep Konam, and Florian Metze, "Asr error correction and domain adaptation using machine translation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6344–6348.

[10] Haoyu Wang, Shuyan Dong, Yue Liu, James Logan, Ashish Kumar Agrawal, and Yang Liu, "Asr error correction with augmented transformer for entity retrieval," 2020.

[11] Anirudh Mani, Shruti Palaskar, and Sandeep Konam, "Towards understanding asr error correction for medical conversations," in *Proceedings of the first workshop on natural language processing for medical conversations*, 2020, pp. 7–11.

[12] Zorik Gekhman, Dina Zverinski, Jonathan Mallinson, and Genady Beryozkin, "Red-ace: Robust error detection for asr using confidence embeddings," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 2800–2808.

[13] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al., "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24824–24837, 2022.

[14] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi, "The curious case of neural text degeneration," *arXiv preprint arXiv:1904.09751*, 2019.

[15] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo, "Falcon-40B: an open large language model with state-of-the-art performance," 2023.