*INTERSPEECH 2020*
October 25–29, 2020, Shanghai, China

# Discriminative Transfer Learning for Optimizing ASR and Semantic Labeling in Task-oriented Spoken Dialog

*Yao Qian, Yu Shi and Michael Zeng*

Microsoft Speech and Dialogue Research Group, Redmond, WA, USA
{yaoqian,yushi,nzeng}@microsoft.com

## Abstract

Spoken language understanding (SLU) tries to decode an input speech utterance such that effective semantic actions can be taken to continue meaningful and interactive spoken dialog (SD). The performance of SLU, however, can be adversely affected by automatic speech recognition (ASR) errors. In this paper, we exploit transfer learning in a Generative pre-trained Transformer (GPT) to jointly optimize ASR error correction and semantic labeling in terms of dialog act and slot-value for a given user's spoken response in the context of SD system (SDS). With the encoded ASR output and dialog history as context, a conditional generative model is trained to generate transcripts correction, dialog act, and slot-values successively. The proposed generation model is jointly optimized as a classification task, which utilizes the ground-truth and N-best hypotheses in a multi-task, discriminative learning. We evaluate its effectiveness on a public SD corpus used in the Second Dialog State Tracking Challenge. The results show that our generation model can achieve a relative word error rate reduction of 25.12% from that in the original ASR 1-best result, and a sentence error rate (SER) lower than the oracle result from the 10-best ASR hypotheses. The proposed approach of generating dialog acts and slot-values, instead of classification and tagging, is promising. The refined ASR hypotheses are critical for improving semantic label generation.

**Index Terms**: speech recognition, human-computer interaction, semantic labeling

## 1. Introduction

Spoken Dialog System (SDS) supports human/machine interactions by using speech. Spoken language understanding (SLU) is a technology to interpret the semantic meaning conveyed in spoken input for taking appropriate actions in task-oriented SDS. It generally consists of two key components: automatic speech recognition (ASR) to convert input speech into recognized text and a natural language understanding (NLU) to transform the ASR word string into semantic labels that can drive subsequent SDS responses. SLU performance can be significantly degraded if ASR is suffering from mismatched acoustic/language models between training and test, e.g., ambient noise, speaker variation in accented pronunciations and out-of-vocabulary words (OOV).

Many approaches have been proposed to reduce the impact of ASR errors to NLU. ASR N-best list or Word Confusion Networks (WCNs) / word lattice that preserves more accurate hypotheses than one-best output was employed as inputs to NLU systems [1–4]. Recently, substantial work has shown that using pre-trained transform-based language models like BERT [5] and tuning it for the downstream NLP tasks outperforms the conventional methods. NLU is one of such downstream tasks [6]. The pre-trained models on a vast text corpus can learn a universal representation for the common-sense knowledge and lexical semantics hiding in the data. Adapting pre-trained transformer to ASR lattice was proposed to SDS in [7], where lattice reachability masks and lattice positional encoding were utilized to enable lattice inputs during fine-tuning. In [8], ASR-robust contextualized embeddings were learned by using word acoustic confusion. The studies on ATIS corpus [9] have shown that both approaches could improve the robustness of NLU to ASR errors.

Meanwhile, many researchers tried to skip ASR entirely or use only partial information, e.g., phone sequence instead of word sequence, extracted from its modules for SLU [10–12]. In [10], it has studied techniques for building call routers from scratch without any knowledge of the application vocabulary or grammar. With the popularity of end-to-end modeling approach, which utilizes as little a prior knowledge as possible, for speech recognition [13], language recognition [14], and etc., ASR-free end-to-end SLU has also been exploited in [15–20], where either the raw waveforms or the acoustic features like filterbank features are directly used as the inputs of SLU models for inferring semantic meaning in order to address the issues caused by ASR.

Inspired by the transfer Learning with a unified text-to-text transformer (T5) [21] where the input and output are always text strings for NLP tasks, we exploit using transfer learning based on a Generative pre-trained Transformer (GPT2) [22, 23] language model to jointly correct ASR errors and label semantics (dialog act and slot-value) for the spoken response in SDS.

The major contribution of this paper is two-fold:

1. We formulate semantic labeling, which is conventionally regarded as a classification or tagging problem, as a conditional generation problem, given the erroneous recognition hypothesis and dialog history. It is jointly optimized with generating corrective ASR hypotheses in a sequential way during the supervised training.
2. A multi-task loss combining generative training for causal LM with discriminative training for reranking N-best list is proposed to transfer learning/fine-tuning based on a pre-trained model for ASR error correction and semantic labeling.

## 2. Related Work

Neural/transformer-based language models have been generally used to rescore the ASR N-best list to improve the performance of speech recognition, i.e., the same objective as that of ASR error correction. The approaches proposed in [24, 25], are similar to our work. They use the recognition output as the context to generate a corrective hypothesis or as one language to translate into another language. Our approach differs from them in two ways: 1) the N-best list is only used to augment the training data



http://dx.doi.org/10.21437/Interspeech.2020-1962

size during the model training and not required in the inference stage; 2) ASR output correction is jointly optimized with the downstream semantic labeling task under the context of SDS.

Fine-tuning a pre-trained LM for NLP tasks like intent classification, slot filling, natural language generation, or end-to-end task completion have been explored for dialog systems [6, 26–28]. However, most of them aim at processing the user's written responses or spoken responses with human transcriptions. Our work tackles the issue caused by ASR for the semantic labeling in SDS, e.g., correct slot tag but erroneous value made by ASR in a slot- filling task.

The highest relevant work to ours has been presented in [29], where multi-task neural approaches were proposed to contextual modeling for ASR correction and language understanding. Their methods and our approaches are all tested on the same corpus, i.e., DSTC-2 [30]. The significant difference between ours and theirs is that they use a classification model to predict dialog act and an IOB tagging model to tag each token in the utterance for slot-filling, while we employ a generation model to generate all semantic labels. Our approach can predict multiple dialog acts and correct the erroneous values for slots for a given spoken response in the sense of generation.

# 3. Joint ASR Error Correction and Semantic Labeling

Transfer learning for a specific task based on a generative model pre-trained on a diverse corpus usually achieves the better generalizability than the model trained only on task-specific data. Inspired by it, we employ transfer learning to GPT-2 for our tasks. GPT-2, a stacked decoder Transformer, is a generative LM trained on a massive unlabeled text from the web, where multi-head self-attention over the context is used to generate distribution for output sequence. We add two additional modeling heads on the top of GPT-2. One head will be trained with an auto-regressive generation process for generating the corrective ASR hypothesis and semantic label sequentially, while the other head will be discriminatively trained to rerank N-best list from both ASR and NLU.

## 3.1. Training

We use a weighted sum of multi-tasks losses, i.e., generation task and classification task, in the training stage.

*Generation Task* Given a dialog corpus of turns $\{(s_1, u_1), ..., (s_K, u_K)\}$ where $s_i$ and $u_i$ represent $i$-th system prompt and user response, respectively. Each $u_i$ contains words or tokens $\{w_i^1, ..., w_i^T)\}$, the objective of training a conditional LM is to maximize the log-likelihood over the entire dialog corpus as following,

$$\mathcal{L}_g = \sum_i \sum_t \log P(w_i^t | w_i^{<t}; (s_i, \hat{u}_i), ..., (s_1, u_1); \theta) \quad (1)$$

where $u_i$ and $\hat{u}_i$ are human transcription and recognition outputs for $i$-th user response, respectively. $w_i^{<t}$ represents all the tokens before $t$. Equation 1 can be changed to the following for jointly training with semantic labeling.

$$\begin{aligned}\mathcal{L}_g =& \alpha \sum_i \sum_t \log P(w_i^t | w_i^{<t}; (s_i, \hat{u}_i), ..., (s_1, u_1); \theta) \\ &+ \beta \sum_i \sum_l \log P(m_i^l | m_i^{<l}; u_i; (s_i, \hat{u}_i), ..., (s_1, u_1); \theta)\end{aligned} \quad (2)$$

where $m_i^l$ is the $l$-th semantic label for $i$-th user response. It can be either dialog act or slot-value. $\alpha$ and $\beta$ are the weights for the conditional LM losses of transcriptions and semantic labels. For generation model head training, we shift each input sequence, $u_i$, to GPT-2, project the hidden state on word embedding matrix and calculate a conditional LM loss on them with cross-entropy. To minimize a gap between training and inference, i.e., the training of semantic label generation is conditioned on ground truth, $u_i$, while ground truth is missing during inference, scheduled sampling proposed in [31] is applied during the training.

*Classification Task* We use $\{\hat{u}_i^1, ..., \hat{u}_i^N\}$ to indicate multiple input sequences, e.g., N-best hypotheses for $i$-th user response. We replace the oracle hypothesis in the N-best list of training data with the ground truth and annotate them with one and other N-1 best hypotheses with zero. The objective of training a discriminative classifier is to distinguish among the various possible classes, i.e., the ground truth can be distinguished from the N-1 best hypotheses in our case,

$$\begin{aligned}\mathcal{L}_c = \sum_i (& \log P(u_i^j | (s_i, \hat{u}_i), ..., (s_1, u_1); \theta) \\ & - \frac{1}{N-1} \sum_{k \neq j} \log P(\hat{u}_i^k | (s_i, \hat{u}_i), ..., (s_1, u_1); \theta))\end{aligned} \quad (3)$$

where $u_i^j$ is the ground truth and $\hat{u}_i^k$ is the $k$-th N-1 best hypotheses. The ground truth and N-1 best hypotheses can also be the concatenation of utterance transcription and semantic labels. For classification model head training, we input every sequence, $u_i^j$ or $\hat{u}_i^k$, into the GPT-2, pass the hidden-state of the last token (the end of sequence token), through a linear layer with one output to get a score and apply a cross-entropy loss to classify correctly ground truth among N input sequences.

## 3.2. Inference

The weighted sum of multi-tasks losses is employed in the training stage, but only the generation task is used to generate outputs in the inference stage. The classification task also can be applied to rerank N-best list of testing data. However, its performance is more interior than that of generation. A detailed analysis will be given in the next section. A schematic diagram of the inference stage in our approach is illustrated in Figure 1. Given the dialog history and the ASR transcription for the current user response, it encodes the tokens, input types and position, passes through a fine-tuned transformer, and generates corrective transcription, semantic labels or the concatenation of them, which depends on the outputs/goals of the fine-tuned transformer in the training stage.

# 4. Experiments and Results

Our approach to ASR error and semantic labeling is evaluated on the corpus named DSTC-2 [30]. The models are constructed using PyTorch [32] and HuggingFace's Transformers [33].

## 4.1. Corpus

DSTC-2 was the corpus collected using Amazon Mechanical Turk in the domain of restaurant information and released for the second state tracking challenge. During the data collection phase, the spoken dialog system provided information about the restaurant and the Turkers were asked to find restaurants that matched their preference on the area, price range and food type. The corpus includes 10-best ASR recognition hypotheses for

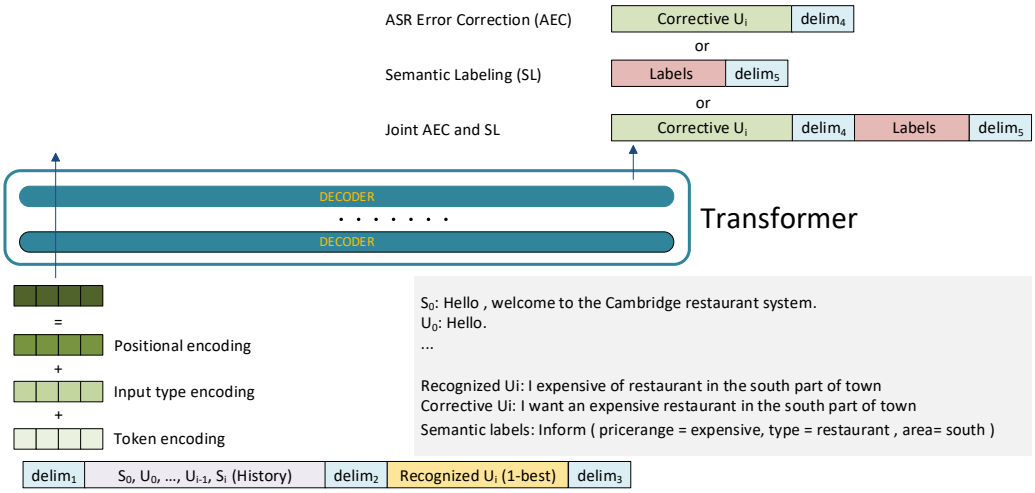


Figure 1: *A schematic diagram of the inference stage in our approach and an example of dialog history, ASR transcription of the current user response, generated corrective transcription and semantic labels.*

each utterance from users instead of audio files. Our goal of this study is the same as in [29], i.e., How to leverage the generative pre-trained transformer-based LM, which employs attention mechanism to consider contextual information, to improve the performance of ASR error correction and semantic labeling, rather than to outperform the state-of-the-art approaches on DSTC-2 corpus. Unlike modifying the original labels for dialog act, converting the slot annotation into IOB tagging format and deleting N-best hypotheses with the transcriptions: "noise", "unintelligible", "silence", "system", "inaudible", and "hello and welcome" in [29], we keep all original labels for slot-value, dialog act and ASR transcription based on the considerations:

1. It is not a straight-forward way to convert the slot annotation into IOB format for many cases in DSTC-2 corpus, e.g., for the transcription, "I am looking for a moderately priced restaurant in the south part of town.", the original labels are dialog act: inform, and slot-values: (area = south, pricerange = moderate, type = restaurant, task = find). The slot values: "find" and "moderate" in the above case do not exactly match the words in the transcription.
2. To investigate the contributions of contextual turns to the performance of proposed approaches, it is better not to delete any turns. Besides, the ASR transcriptions, e.g., empty, noise, and silence are also quite common in a real scenario of SDS.

In total, there are 11,677 train, 3,934 development, and 9,890 test utterances extracted from 1,612 train, 506 development, and 1, 117 test calls/dialogs used in this study. Please refer to [30] for the detailed annotations of the dialog act and slot-value. An example is shown in Figure 1.

### 4.2. Experimental Setup

Word error rate (WER) and sentence error rate (SER) were used to evaluate the performance of ASR error correction. The performance metrics for semantic labeling are dialog act accuracy (DA-Acc), slot-value F1 (slot-F1), and sentence-level semantic frame accuracy (Frame-Acc), where the sentence with all correct labels including both dialog act and slot-value is counted as a correct sentence. We use byte-level Byte-Pair-Encoding [34] for tokenizer and GPT-2 small that consists of 12-layer, 768-hidden, 12-heads with 124M parameters as the pre-trained model for fine-turning to adapt it to our tasks. The preliminary results of transfer learning based on GPT-2 small (124M), GPT-2 medium (355M) and GPT-2 large (774M) for semantic label generation given ground-truth transcription, which are shown in Table 1, indicate that the larger GPT-2 model and the better performance, but it is out of the scope of this article, and the detailed comparison of model size vs performance will be considered as the future work.

We consider 1-best ASR output and N-best list rescoring with transformer-based LM as the baseline of ASR error correction, and the semantic labeling based on 1-best ASR output as the baseline of semantic labeling. Instead of training a transformer-based LM from scratch, we fine-tune GPT-2 (small) LM with the ground-truth (human transcription) of the training data set. To have a fair comparison with our approaches, which consider long contexts, we delimit the dialog history, i.e., the previous system prompts/user responses and the current system prompt, with a special token and concatenate them together as inputs to the LM modeling. The resultant GPT-2 LM is employed to rescore the N-best list of the testing set. For the generation tasks, the training data set is augmented by using the ASR N-best list of training data. There are a total of 114,690 pairs of ASR transcription and human transcription and the corresponding semantic label sequences used in the training stage.

Adam optimizer with weight decay [35] and a schedule with a learning rate that decreases linearly are used to train models. The weight of score interpolation in baseline N-best rescoring, weights for joint generation, weights for multi-tasks losses, the number of epochs, and the starting point of the learning rate for model training are optimized by using the validation/development data set. Top-k sampling that samples from the next-token distribution by keeping only the top k tokens is employed to generate hypotheses in the inference stage.

### 4.3. Results and Discussion

Table 1 lists the performance on the testing set in terms of WER and SER for ASR error correction and DA-acc, Slot-F1, and Frame-Acc for semantic labeling obtained by using the base-

Table 1: *Performance(%) of different systems*

| Tasks | Experimental Settings | WER | SER | DA-Acc | Slot-F1 | Frame-Acc |
|---|---|---|---|---|---|---|
| Semantic | ground truth (small/medium/large) | | | 97.75/98.22/98.64 | 97.63/98.17/98.58 | 94.30/95.91/96.87 |
| Semantic | 1-best | 32.2 | 66.47 | 87.20 | 83.86 | 75.54 |
| Semantic | oracle (10-best) | 21.83 | 44.98 | 89.26 | 86.54 | 78.90 |
| ASR | 10-best rescoring | 28.83 | 57.02 | | | |
| ASR | 10-best reranking | 29.61 | 55.23 | | | |
| ASR | generating | 25.64 | 44.22 | | | |
| ASR | generating (multi-task learning) | 24.5 | 42.56 | | | |
| Joint | generating | 24.71 | 42.90 | 88.89 | 83.94 | 76.74 |
| Joint | generating (multi-task learning) | 24.11 | 42.21 | 89.01 | 84.19 | 77.25 |

line and the proposed approaches. The semantic labeling results from ground-truth/human transcriptions and the oracle outputs from ASR N-best lists are also included in Table 1 as the upper-bound for the comparisons.

As the performance of the semantic labeling model by using the ground truth as the input shown in Table 1, it can be observed that our approach of using generation model to produce a sequence of semantic labels is very promising, or to be precise, it achieves the DA-acc of 97.75%, the Slot-F1 of 97.63%, and the Frame-acc of 94.30%. Although it is not completely fair to compare our results with those achieved by using a classification model in [29] since the data size and labeling strategies used in the experiments are not exactly the same (They have not released the modified annotations for DSTC-2), a close result indicates that the performance of our approach is on a par with the conventional approach. The same observation shown in [29] is that the performance of the semantic labeling model is significantly degraded if the ASR 1-best hypotheses are used as the inputs to the models.

The baseline that uses fine-tuned GPT LM to rescore N-best list can achieve relative WER reduction of 10.5% for the testing set, i.e., WER is decreased from 32.2% to 28.83%. This result is very close to that of the same approach presented in [29], wherein the relative WER reduction of 10.6% was obtained. We have tried to use the training data to train a conventional statistical LM and an LM with the same architecture of GPT2 from scratch but both observed the poor performance. A ranker trained by using GPT-2 can achieve the relative 8% reduction in WER, which is worse than that of rescoring. We suspect that it is caused by mismatched training and testing data. The ranker, i.e., classification task, was trained by using N-best list of training data wherein the oracle hypotheses were replaced by the human transcription while human transcription is not available for N-best list of testing data. The ranker was used as an auxiliary learning task to aid the generation task. This is why we consider replacing the oracle hypothesis with the ground truth.

The generation model can attain a significant improvement in ASR error correction. It reduces the WER from 32.2% to 25.64%, i.e., a relative WER reduction of 20.4%. By joint optimization with classification task, semantic labeling and both in the training stage, its performance can be further improved in terms of relative WER reduction of 23.91%, 23.30% and 25.12%, respectively. A noticeable result observed is that the 42.21% SER of the generated hypotheses by our approach is lower than the 44.98% SER of oracle hypotheses. These results are superior to those shown in [29]. An example of the hypotheses produced by 1-best ASR, 10-best rescoring, and generating is shown in Table 2. It indicates that the generation model can produce a much more meaningful sentence than that of rescoring.

Table 2: *An example of the hypotheses produced by 1-best ASR, 10-best rescoring and generating*

| |
|---|
| **Ground truth:** i want to find a moderately priced restaurant and it should be in the west part of town |
| **1-best:** i would like moderately priced restaurant you should be in the west part of the cow |
| **Re-scoring:** i would like a moderately priced restaurant you should be in the west part of the cow |
| **Generating:** i would like a moderately priced restaurant in the west part of town |

Our approach of multi-task learning for ASR error correction and semantic labeling can improve the performance of semantic labeling as well. Table 1 shows that the model trained by jointly generating transcription and semantic label sequence can improve the DA-Acc from 87.20% to 88.89%, Slot-F1 from 83.86% to 83.94% and Frame-Acc from 75.54% to 76.74% and the model trained by joint generation and classification tasks can further improve DA-Acc, Slot-F1, and Frame-Acc to 89.01%, 84.19%, and 77.25%, respectively. The performance in terms of DA-Acc is close to that of using oracle hypotheses from the N-best list.

We analyze the results produced by our approach and find that our approach has advantages over the conventional approach of IOB tag prediction for semantic labeling, 1) It can produce multiclass naturally, e.g., A label sequence of *ack () | reqalts (food=thai)* for the utterance "okay how about thai food", which contains two acts: "acknowledge" and "requesting for alternative suggestions"; 2) It can also flexibly produce the empty for slot type or slot value, e.g, the labels of *inform (=dontcare)* for the user response "any" to the system prompt "What part of town do you have in mind?". However, it will occasionally produce duplicated slot-values and nonsense labels.

## 5. Conclusions

In this study, transfer learning based on GPT-2 is proposed to correct ASR errors and to improve the corresponding semantic labels in SDS. Experimental results show that our approach is effective in improving the performance of both tasks. In the future, we will generalize the proposed approach to multi-domain SDS corpora and to build a universal model and test it in unseen domains that are short of adequate training data.

# 6. References


[1] G. Tur, J. Wright, A. Gorin, G. Riccardi, and D. Hakkani-Tür, "Improving spoken language understanding using word confusion networks," in *Seventh International Conference on Spoken Language Processing*, 2002.

[2] D. Hakkani-Tür, F. Béchet, G. Riccardi, and G. Tur, "Beyond ASR 1-best: Using word confusion networks in spoken language understanding," *Computer Speech & Language*, vol. 20, no. 4, pp. 495–514, 2006.

[3] M. Henderson, M. Gašić, B. Thomson, P. Tsiakoulis, K. Yu, and S. Young, "Discriminative spoken language understanding using word confusion networks," in *2012 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 176–181.

[4] F. Ladhak, A. Gandhe, M. Dreyer, L. Mathias, A. Rastrow, and B. Hoffmeister, "LatticeRnn: Recurrent neural networks over lattices." in *Interspeech*, 2016, pp. 695–699.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[6] Q. Chen, Z. Zhuo, and W. Wang, "Bert for joint intent classification and slot filling," *arXiv preprint arXiv:1902.10909*, 2019.

[7] C.-W. Huang and Y.-N. Chen, "Adapting pretrained transformer to lattices for spoken language understanding," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 845–852.

[8] ——, "Learning ASR-robust contextualized embeddings for spoken language understanding," *arXiv preprint arXiv:1909.10861*, 2019.

[9] C. T. Hemphill, J. J. Godfrey, and G. R. Doddington, "The ATIS spoken language systems pilot corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania*, 1990.

[10] Q. Huang and S. Cox, "Task-independent call-routing," *Speech Communication*, vol. 48, no. 3-4, pp. 374–389, 2006.

[11] A. L. Gorin, D. Petrovska-Delacretaz, G. Riccardi, and J. H. Wright, "Learning spoken language without transcriptions," in *IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU*, vol. 99, 1999.

[12] H. Alshawi, "Effective utterance classification with unsupervised phonotactic models," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 1–7.

[13] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.

[14] W. Geng, W. Wang, Y. Zhao, X. Cai, B. Xu, C. Xinyuan *et al.*, "End-to-end language identification using attention-based recurrent neural networks," in *Interspeech*, 2016, pp. 2944–2948.

[15] Y. Qian, R. Ubale, V. Ramanaryanan, P. Lange, D. Suendermann-Oeft, K. Evanini, and E. Tsuprun, "Exploring ASR-free end-to-end modeling to improve spoken language understanding in a cloud-based dialog system," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 569–576.

[16] Y.-P. Chen, R. Price, and S. Bangalore, "Spoken language understanding without speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6189–6193.

[17] P. Haghani, A. Narayanan, M. Bacchiani, G. Chuang, N. Gaur, P. Moreno, R. Prabhavalkar, Z. Qu, and A. Waters, "From audio to semantics: Approaches to end-to-end spoken language understanding," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 720–726.

[18] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," *arXiv preprint arXiv:1904.03670*, 2019.

[19] D. Serdyuk, Y. Wang, C. Fuegen, A. Kumar, B. Liu, and Y. Bengio, "Towards end-to-end spoken language understanding," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5754–5758.

[20] P. Wang, L. Wei, Y. Cao, J. Xie, Y. Cao, and Z. Nie, "Understanding semantics from speech through pre-training," *arXiv preprint arXiv:1909.10924*, 2019.

[21] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[22] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, 2018.

[23] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[24] T. Tanaka, R. Masumura, H. Masataki, and Y. Aono, "Neural error corrective language models for automatic speech recognition." in *Interspeech*, 2018, pp. 401–405.

[25] O. Hrinchuk, M. Popova, and B. Ginsburg, "Correction of automatic speech recognition with transformer sequence-to-sequence model," *arXiv preprint arXiv:1910.10697*, 2019.

[26] B. Peng, C. Zhu, C. Li, X. Li, J. Li, M. Zeng, and J. Gao, "Few-shot natural language generation for task-oriented dialog," *arXiv preprint arXiv:2002.12328*, 2020.

[27] C.-S. Wu, A. Madotto, E. Hosseini-Asl, C. Xiong, R. Socher, and P. Fung, "Transferable multi-domain state generator for task-oriented dialogue systems," *arXiv preprint arXiv:1905.08743*, 2019.

[28] P. Budzianowski and I. Vulić, "Hello, it's gpt-2–how can i help you? towards the use of pretrained language models for task-oriented dialogue systems," *arXiv preprint arXiv:1907.05774*, 2019.

[29] Y. Weng, S. S. Miryala, C. Khatri, R. Wang, H. Zheng, P. Molino, M. Namazifar, A. Papangelis, H. Williams, F. Bell *et al.*, "Joint contextual modeling for ASR correction and language understanding," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6344–6348.

[30] M. Henderson, B. Thomson, and J. D. Williams, "The second dialog state tracking challenge," in *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, 2014, pp. 263–272.

[31] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," *ArXiv preprint arXiv:1506.03099*, 2015.

[32] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[33] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv preprint arXiv:1910.03771*, 2019.

[34] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *ArXiv preprint arXiv:1508.07909*, 2015.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ArXiv preprint arXiv:1412.6980*, 2014.