Adapting Long Context NLM for ASR Rescoring in Conversational Agents

Ashish Shenoy, Sravan Bodapati, Monica Sunkara, Srikanth Ronanki, Katrin Kirchhoff

Amazon AWS AI, USA

{ashenoy, sravanb, sunkaral, ronanks, katrinki}@amazon.com

Abstract

Neural Language Models (NLM), when trained and evaluated with context spanning multiple utterances, have been shown to consistently outperform both conventional n-gram language models and NLMs that use limited context. In this paper, we investigate various techniques to incorporate turn based context history into both recurrent (LSTM) and Transformer-XL based NLMs. For recurrent based NLMs, we explore context carry over mechanism and feature based augmentation, where we incorporate other forms of contextual information such as bot response and system dialogue acts as classified by a Natural Language Understanding (NLU) model. To mitigate the sharp nearby, fuzzy far away problem with contextual NLM, we propose the use of attention layer over lexical metadata to improve feature based augmentation. Additionally, we adapt our contextual NLM towards user provided on-the-fly speech patterns by leveraging encodings from a large pre-trained masked language model and performing fusion with a Transformer-XL based NLM. We test our proposed models using N-best rescoring of ASR hypotheses of task-oriented dialogues and also evaluate on downstream NLU tasks such as intent classification and slot labeling. The best performing model shows a relative WER between 1.6% and 9.1% and a slot labeling F1 score improvement of 4% over non-contextual baselines.

Index Terms: Neural language models, speech recognition, conversational chat bots.

1. Introduction

Conversations in goal or task oriented conversational interfaces, such as digital personal assistants or chatbots, typically span multiple turns of back and forth between a user and a bot [1, 2]. These interactions by definition involve accomplishing a specific task in a particular domain and mostly adhere to a dialogue structure that can be determined beforehand. Hence, most real-time task oriented chatbot systems allow users to provide a dialogue grammar that tries to capture usage patterns, intents, slots, and the conversation structure of the interactions [3, 4, 5].

A conventional NLM rescorer for ASR is typically trained and evaluated on context that is limited to single turns [6, 7] and is therefore sub-optimal in task oriented dialogue systems. A number of studies recently explored incorporating cross utterance context into both recurrent and non-recurrent NLMs. In particular, recurrent neural network (RNN) and LSTM based NLMs are trained and evaluated without resetting hidden states across sentences [8, 9, 10]. The work proposed in [11] uses a multi-head dot-product attention over LSTM hidden states to better exploit the contextual information. However, context carry over models with either RNNs or LSTMs each has its own limitations. The wide spread adoption of transformer architecture based on self attention [12] and pretrained masked language models [13] paved the way for use of transformers for NLMs. In transformer NLMs, long span context modeling is achieved through longer or adaptive attention spans [14]. While the initial work used modified attention masks to handle longer input sequences more efficiently, Transformer-XL (TXL) [15] made use of segment wise recurrence, making it well suited for long span decoding. More recently, [16] further tried to improve cross utterance decoding using TXL by adding a LSTM fusion layer, where the hidden states of LSTM are carried over multiple sentences. The work in [17] explored the use of a conversational context embedding from recent history to improve the contextualization of end-to-end ASR models where as some other works [18, 19, 20, 21] used an explicit topic vector or a neural cache and a domain classifier for domain and contextual adaptation.

While vast majority of the previous work focused on using context spanning multiple utterances, they do not take into account other contextual signals from NLU such as dialogue act, predefined dialogue grammars or any user-provided speech patterns. In this work, we investigate a number of ways to improve contextualization of LSTM and TXL based NLMs to transcribe task-oriented dialogue audio. Specifically, we use long span context that spans across all the utterances in the same dialogue session and system dialogue acts as classified by a NLU model. Additionally, in order to adapt the NLM towards user provided speech patterns in the pre-defined dialogue grammar, we use semantic embeddings derived from a large pretrained masked language models such as BERT [13]. We use perplexity (PPL) and word error rate (WER) as our ASR evaluation metrics and also evaluate on the natural language understanding (NLU) metrics such as intent classification (IC) and slot labeling (SL) F1 scores to measure the impact on end to end task success rate. The overall contributions of this work can be summarized as follows :

- For speech recognition in task-oriented conversations, we show that utilizing long span context from past utterances in the same dialogue session along with system dialogue act, provides significant improvements in WER reduction (WERR).
- We propose a new architecture that lets us leverage user provided speech patterns by using embeddings derived from a pretrained masked language model, such as BERT, to perform on-the-fly adaptation of a neural rescorer.
- By combining the different forms of contextual information, we successfully train NLMs that achieve a WERR ranging from 1.6% to 9.1%, IC F1 improvement ranging from 0.3% to 1.2% and SL F1 ranging from 0.4% to 4.5% over a non-contextual LSTM LM baseline.

2. Approach

2.1. Recurrent neural language models

2.1.1. Baseline LSTM-based NLM

A typical language model in an ASR system computes the probability of a sequence of words $W = w_0, ..., w_N$ auto-

regressively as:

$$p(W) = \prod_{i=1}^{N} p(w_i|w_{<i}) \tag{1}$$

We use a standard LSTM-based neural network that can model the probability of a word given its history as described in equation 1. If $w_0, w_1.., w_n$ be a sequence of words in a dialogue turn, the probability of $p(w_i|w_{<i})$ can be summarized as below

$$\begin{aligned} embed_i &= E_{ke}^T w_{i-1} \\ c_i, h_i &= LSTM(h_{i-1}, c_{i-1}, embed_i) \\ p(w_i|w_{<i}) &= Softmax(W_{ho}^T h_i) \end{aligned} \tag{2}$$

where $embed_i$ is a fixed size lower dimensional word embedding, the output from the LSTM is projected to word level outputs using W_{ho}^T and then a $Softmax$ layer converts the word level outputs into final word level probabilities.

2.1.2. *Long-span LSTM-based NLM with context carry over*

One way of achieving contextual language modeling in LSTM-LMs is to simply carry over the fixed size context vectors h_n and c_n after the last time step in the previous turn to the initial state in the current turn. For this purpose, we use (a) system dialogue act, and (b) previous bot response to prime the model for the current user turn. The dialogue act (DA) and the bot response (BR) are concatenated and separated with an explicit tag (such as "<dialog_act>" and "<bot_response>") that we define and include in the vocabulary. The gradient is backpropogated only for user turns.

2.1.3. *Feature augmentation using context embeddings*

It is known that LSTM-based NLMs with context carry over (CCO) suffer from sharp nearby, fuzzy far away issues [22] and therefore we propose to investigate input feature augmentation using various context embeddings.

Average word embedding: A standard word embedding module produces word-wise embedding vectors over contextual representations; the final fixed length embedding is obtained by averaging embedding vectors over the entire contextual sequence.

$$\begin{aligned} embed_{ctx} &= \langle E_{ke}^T[ctx_1; ctx_2 ...; ctx_k] \rangle \\ \hat{embed}_i &= [embed_{ctx}; embed_i] \end{aligned} \tag{3}$$

where $embed_i$ is the input word embedding at time step i.

Attention word embedding: Some parts of the contextual information may be more important than the others. To allow for this we add an explicit attention layer over the context. The context representation embeddings are passed through a weighted attention layer. The output embedding from the weighted attention layer is concatenated with the input word embedding. The attention layer can be summarized as follows and is similar to [23] and [24]. Here e_t is the embedding for t-th word in context and u_w is the query word embedding in the utterance being decoded:

$$\begin{aligned} u_t &= tanh(W_w e_t + b_w) \\ a_t &= \frac{exp(u_t u_w)^\top}{\sum_t exp(u_t u_w)^\top} \\ s_i &= \sum_t a_t h_t \\ \hat{embed}_i &= [s_i; embed_i] \end{aligned} \tag{4}$$

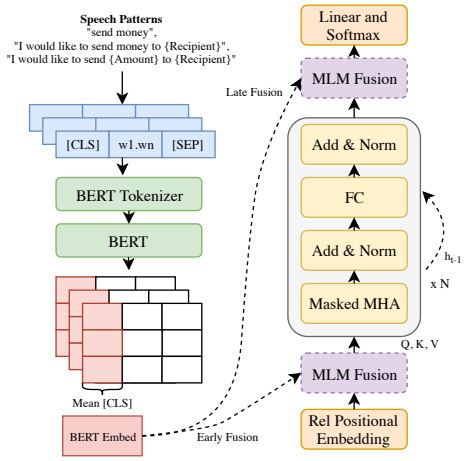

Figure 1: *Transformer-XL model architecture showing the early and late MLM fusion layer.*

Semantic embedding: Pretrained masked language models (MLM) trained on large scale unlabelled data, are capable of learning richer semantic language representations and are powerful language learners [25, 26]. Specifically, at training time, we use BERT to obtain embeddings from the CLS token of all the training sentences separated by domain. The CLS token is a special classification token that is prefixed to every sequence and the final hidden state corresponding to this token is used as the aggregate sequence representation for classification tasks [13, 27]. Then for every domain, we average these embeddings to obtain a single fixed length domain embedding. At inference time, we use these user provided speech patterns or sample utterances to obtain a domain representation for the bot. We then use cosine similarity to pick the closest domain embedding seen at training time to adapt the NLM towards these speech patterns.

2.2. Non recurrent neural language models

2.2.1. *Transformer-XL based NLM*

To address the limitations of using a fixed-length context, Transformer-XL [15] adds a recurrence mechanism to the Transformer architecture. Hence, we directly experiment with the TXL models where we carry over context across sentences. During training, the hidden state sequence computed for the previous segment is reused while processing a new segment as an extended context. The final model consists of a stack of decoder blocks, where each block includes a multi-headed attention (MHA) layer with a residual connection and a set of full-connected (FC) layers followed finally by a linear and softmax layer. The MHA layer uses upper triangular masks in order to ensure uni-directionality while decoding. Formally, if $s_t = [x_{t,1}, ..., x_{t,L}]$ and $s_{t+1} = [x_{t+1,1}, ..., x_{t+1,L}]$ are two consecutive segments of length L and h_t^n is the n-th layer hidden state produced for the t-th segment s_t, then, the n-th layer hidden state for segment s_{t+1} is produced as follows:

$$\begin{aligned} \tilde{h}_{t+1}^{n-1} &= [SG(h_t^{n-1}) \circ h_{t+1}^{n-1}] \\ q_{t+1}^n, k_{t+1}^n, v_{t+1}^n &= \tilde{h}_{t+1}^{n-1} W_q^\intercal \\ h_{t+1}^n &= TransformerLayer(q_{t+1}^n, k_{t+1}^n, v_{t+1}^n) \end{aligned} \tag{5}$$

where $SG(.)$ stands for stop gradient.

Source	# Dialogues	# Utterances
DSTC8 SGD	22825	231642
MultiWOZ 2.1	10420	143048
MultiDoGo	102870	1376816

Table 1: *Goal-oriented datasets and sizes used*

2.2.2. MLM fusion with Transformer-XL

We explore several different techniques of fusing the semantic embeddings derived from pretrained masked language models (MLMs) with TXL. They can be broadly categorized as early and late fusion as shown in Figure 1. In late fusion we explore two different methods : Simple and Cold fusion similar to [28] and [29].

Early Fusion : In this technique, we explore input level fusion by combining the input word embedding with BERT derived embedding at every time step. The computation procedure can be summarized as:

$$g_t = \sigma(W[E_t; e^{MLM}] + b) \tag{6}$$

Simple Fusion : In this form of fusion the hidden state from the last layer of the TXL decoder is concatenated with the BERT derived embedding and is followed by a single projection layer with a non-linear activation function σ, such as $sigmoid$.

$$g_t = \sigma(W[h_t^{TXL}; e^{MLM}] + b) \tag{7}$$

Cold Fusion : This is a more complex fusion that is achieved by introducing gates. The gated fusion allows moderation of information flow between the TXL and the BERT derived embedding during training. Our approach is a modified cold fusion approach from [30] which is as follows:

$$\begin{aligned} \widehat{e^{MLM}} &= \sigma(W[e^{MLM}] + b) \\ g_t &= \sigma(W[\widehat{e^{MLM}}; h_t^{TXL}] + b) \\ h_t^{CF} &= [h_t^{TXL}; (g_t \circ e^{MLM})] \\ r_t^{CF} &= \sigma(W[h_t^{CF}] + b) \end{aligned} \tag{8}$$

Where E_t is input word embedding, h_t^{TXL} is the hidden state from the last transformer decoder, e^{MLM} is the BERT derived embedding from in domain sample utterances and $\circ$ stands for hadamard product. In all the above methods, we pass the output from the fusion layer to a linear layer followed by softmax to predict the next word in the sequence.

3. Experimental Setup

3.1. Dataset

We use a combination of task-oriented user-bot dialogues along with actor, domain and dialogue act annotated for LM training: Schema-Guided Dialogue Dataset [31], MultiWOZ 2.1 [1] [32] and a small sample of MultiDoGo [33]. These are user-agent conversations with an average of 10 turns per dialogue. The average length of a sequence in a single dialogue session where all turns are concatenated was around 120. The statistics of the data sets are listed in Table 1. The merged text only dataset consisted of 260,415 training samples, 51,602 validation samples, 56,091 test samples and around 9.9 million running words. The final vocabulary contained 25,000 most frequent words, which included words from bot responses and user samples. The out-of-vocabulary words were modeled with the `<unk>` token. Since we used a combination of data sources annotated with different tagging schemes, we had to normalize the dialogue acts across the datasources using a heuristic as shown in Table 2. Each of our models was evaluated on two thousand 8kHz anonymized in-house close-talk task-oriented audio conversations. The average number of turns in the audio dataset was 6 and was representative of the real world usage of task-oriented chatbots and followed the same conversation style and distribution as the text-only datasources mentioned above.

Dialogue Act	Normalized DA
confirm, recommend, offerbook	confirm
inform, inform_count	inform
offer, offerbook, offer_intent, select	offer
request	request
general-bye, goodbye	general-bye
general-welcome	general-welcome
general-reqmore	general-reqmore

Table 2: *Normalization of system dialogue acts across data sources*

3.2. N-best rescoring setup

The language model we used for first pass consisted of a standard 4-gram language model, trained on a weighted mix of out-of-domain and in-domain datasets. The weights were determined by minimizing the perplexity on an in-domain dev set. The Kneser-Ney (KN) [34] smoothed n-gram model estimated from the corpora had a final vocabulary of 500k words. All the neural language models had a word embedding size of 512. The BERT derived embeddings had a size of 768. In the LSTM LM models, we used a 3-layer LSTM, each of size 1,150. For transformer LMs, we used 6-layer Tranformer-XL decoder, each of size 512 with 4 attention heads. We used a segment and memory length of 15 while training. Both NLMs were trained with cross entropy objective loss function. During inference, we extract n-best hypothesis with `n<=50` from the lattice generated by the first pass ASR model. We rescored the n-best hypothesis by multiplying the acoustic score with the acoustic scale and adding it to the scores obtained from the second pass NLM.

Model	PPLR	WERR
Non-contextual LSTM	-	-
Context Carry Over	22.53%	1.65%
+ BERT	19.59%	4.58%
Feature Augmentation		
+ Avg(BR)	0.4%	2.06%
+ Avg(BR;DA)	9.61%	6.61%
+ Attn(BR;DA)	15.38%	2.89%
+ Avg(BR;DA) + BERT	2.67%	7.02%
+ Attn(BR;DA) + BERT	16.04%	5.78%

Table 3: *Relative Perplexity Reduction (PPLR) and the relative Word Error Rate Reduction (WERR) for the LSTM models. Attn: Attention, BR: Bot Response, DA: Dialogue Acts.*

4. Results and Discussion

First, we compare the significance of incorporating auxiliary contextual signals and long context into recurrent (LSTM) and non recurrent (TXL) based NLMs. Tables 3 and 4 summarize the overall relative perplexity reduction (PPLR) and the WERR for models with different settings. Then, we evaluate our best

Model	PPLR	WERR
Non-contextual LSTM	-	-
TXL	9.98%	1.67%
+ Context Carry Over (CCO)	18.91%	3.34%
Masked Language Model (BERT)		
+ Early Fusion (EF)	31.63%	5.78%
+ Cold Fusion (CF)	32.93%	5.78%
+ Simple Fusion (SF)	37.53%	8.34%
+ SF + CCO	42.80%	9.16%

Table 4: *Relative Perplexity Reduction (PPLR) and the relative Word Error Rate Reduction (WERR) for the TXL models.*

models on downstream NLU tasks and report the intent classification and slot labeling F1 scores in Table 5. We also determine statistical significance of our WER improvements using matched pairs sentence segment word error test (MAPSSWE).

Significance of System Dialogue Acts: From our experiments in Table 3, we observe that LSTM based LMs with CCO show high perplexity improvements, but WERR improvements are very marginal. We realize better WERR improvements by incorporating feature augmentation, which seems to help mitigate the problem of fuzzy distant context with LSTMs. Our best model was obtained by using system dialogue act as an additional feature. From the results, it is evident that dialogue acts, that are intended towards capturing the function of a dialogue turn, is an important contextual cue that the model learns to utilize effectively.

Effectiveness of BERT embeddings with Recurrent NLMs: We conducted experiments to validate the extent of performance improvements that can be realized by using BERT embeddings extracted from user speech patterns. From results in Table 3, it is clear that using BERT derived embeddings gives additional improvements in terms of WER when used in conjuction with CCO (1.65% to 4.58%) and feature augmentation (average embeddings: 6.61% to 7.02%, attention based embeddings: 2.89% to 5.78%). Results show that the model is able to effectively attend to both the previous historical context and the rich semantic information condensed in the BERT embeddings generated from user provided speech patterns. Also, the observed perplexity gains indicate that the model is able to generate sharper probability distributions over domain specific vocabulary.

Effectiveness of BERT embeddings with Non-Recurrent NLMs : Table 4 compares the improvements in PPL and WER, realized by incorporating BERT derived embeddings using different fusion techniques into a non-recurrent language model (vanilla TXL). In our experiments, we use TXL models with (memory=50) and without (memory=0) context. Our best performing model is obtained with SF + CCO setting, and has a relative improvement of 42.8% and 9.16% in terms of PPL, WERR respectively, over a strong LSTM baseline. This shows that fusion with BERT embeddings extracted from user provided speech patterns is an effective way to adapt NLMs to user specific domain. Also, from the qualitative results presented in Table 6, we can observe that text sampled from our best model (TXL + BERT SF) is salient to the user speech patterns.

Performance on downstream NLU tasks: To capture the impact on the end goals in these conversations, we measure the performance in terms of intent classification F1 score, slot labeling F1 score and content word error rate reduction (CWERR) on the ASR rescored outputs, using an NLU model. For computing CWERR, we remove all the stop words (commonly used function words, such as conjunctions and preposition) from the transcriptions and evaluate only on content words. From results in Table 5, we can observe that our TXL models significantly outperform the LSTM models in terms of content word recognition. Though the LSTM LMs are able to improve on slot carrying auxiliary phrases, these gains fail to translate into actual slot recognition. The slot labeling F1 score improves by 0.48% with the contextual LSTM LM and with a contetxual TXL this further improves significantly by 4.55%.

Model	CWERR	IC F1	SL F1	p-value
Non-contextual LSTM				
+ Avg(BR;DA) + BERT	11.3%	0.33%	1.20%	**0.031**
+ Attn(BR;DA) + BERT	6.08%	0.17%	0.48%	0.195
+ CCO + BERT	6.08%	0.17%	0.48%	0.052
TXL	12.60%	0.46%	3.36%	**0.026**
+ CCO	15.65%	0.46%	3.57%	**0.003**
+ BERT SF	18.26%	1.05%	4.15%	**0.002**
+ BERT SF + CCO	18.26%	1.20%	4.55%	**0.004**

Table 5: *Relative reduction in content WER (CWERR), intent classification (IC) F1 and slot labeling (SL) F1 and MAPSSWE p-value test on WER with significant improvements in bold.*

Input Prompt: "<sos> can you" **BERT Embed:** None **Generated Text:** "send a copy of the details please <eos>"
Input Prompt: "<sos> can you" **BERT Embed:** "travel" **Generated Text:** "tell me what amenities are offered at the hotel <eos>"
Input Prompt: "<sos> can you" **BERT Embed:** "bank" **Generated Text:** "show me all the transfers scheduled <eos>"

Table 6: *Example text sampled from TXL + BERT SF model. The model is primed with an input prompt and a domain specific BERT embedding. When compared to providing an empty BERT embedding, the model adapts to the speech patterns when a domain specific BERT embedding is provided.*

5. Conclusions

In this paper we explored different ways to incorporate auxiliary context information along with cross utterance context to improve LSTM and Transformer-XL based NLMs to rescore n-best list from a hybrid ASR system. We introduced a new method that successfully adapts both recurrent and non-recurrent based NLMs towards user provided speech patterns on-the-fly by using semantic pretrained BERT embeddings. Additionally, we show that NLM rescoring in task oriented dialogue systems can be improved significantly by combining context carry over with feature based augmentation that includes system dialogue acts. Experiments on task oriented audio conversations show substantial WER and PPL gains which also carry over to downstream NLU metrics such as improved intent classification and slot labeling F1 scores. Future work can look at generalization of the improvements on out of domain utterances and utilizing these methods to rescore hypothesis from an end-to-end ASR system.

6. References

[1] P. Budzianowski, T. Wen, B. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic, "Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling," *CoRR*, vol. abs/1810.00278, 2018.

[2] E. Hosseini-Asl, B. McCann, C.-S. Wu, S. Yavuz, and R. Socher, "A simple language model for task-oriented dialogue," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 20 179–20 191.

[3] P. Aleksic, M. Ghodsi, A. Michaely, C. Allauzen, K. Hall, B. Roark, D. Rybach, and P. Moreno, "Bringing contextual information to google speech recognition," in *Interspeech*, 2015.

[4] O. Lemon, K. Georgila, J. Henderson, and M. Stuttle, "An ISU dialogue system exhibiting reinforcement learning of dialogue policies: Generic slot-filling in the TALK in-car system," in *Demonstrations*, 2006.

[5] A. Gandhe, A. Rastrow, and B. Hoffmeister, "Scalable language model adaptation for spoken dialogue systems," in *SLT Workshop 2018, Athens, Greece*. IEEE, 2018, pp. 907–912.

[6] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model." in *Interspeech*, T. Kobayashi, K. Hirose, and S. Nakamura, Eds., 2010, pp. 1045–1048.

[7] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," in *Advances in Neural Information Processing Systems*, T. Leen, T. Dietterich, and V. Tresp, Eds., vol. 13. MIT Press, 2001.

[8] W. Xiong, L. Wu, F. Alleva, J. Droppo, X. Huang, and A. Stolcke, "The microsoft 2017 conversational speech recognition system," in *ICASSP, Calgary, Canada*. IEEE, 2018, pp. 5934–5938.

[9] K. Irie, A. Zeyer, R. Schlüter, and H. Ney, "Training language models for long-span cross-sentence evaluation," in *ASRU Singapore*. IEEE, 2019, pp. 419–426.

[10] W. Xiong, L. Wu, J. Zhang, and A. Stolcke, "Session-level language modeling for conversational speech," in *EMNLP*. Brussels, Belgium: ACL, 2018.

[11] S. Parthasarathy, W. Gale, X. Chen, G. Polovets, and S. Chang, "Long-span language modeling for speech recognition," *CoRR*, vol. abs/1911.04571, 2019.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, 2017, pp. 5998–6008.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT 2019*, Jun. 2019, pp. 4171–4186.

[14] S. Sukhbaatar, E. Grave, P. Bojanowski, and A. Joulin, "Adaptive attention span in transformers," in *ACL*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 331–335.

[15] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," in *ACL*, Florence, Italy, Jul. 2019, pp. 2978–2988.

[16] G. Sun, C. Zhang, and P. C. Woodland, "Transformer language models with lstm-based cross-utterance information representation," 2021.

[17] S. Kim, S. Dalmia, and F. Metze, "Gated embeddings in end-to-end speech recognition for conversational-context fusion," in *ACL*, Florence, Italy, Jul. 2019, pp. 1131–1141.

[18] T. Mikolov and G. Zweig, "Context dependent recurrent neural network language model." in *SLT*. IEEE, 2019, pp. 234–239.

[19] K. Li, H. Xu, Y. Wang, D. Povey, and S. Khudanpur, "Recurrent neural network language model adaptation for conversational speech recognition," in *Interspeech, Hyderabad, India, 2-6 September 2018*, 2018, pp. 3373–3377.

[20] A. Raju, B. Hedayatnia, L. Liu, A. Gandhe, C. Khatri, A. Metallinou, A. Venkatesh, and A. Rastrow, "Contextual language model adaptation for conversational agents," in *Interspeech, Hyderabad, India*. ISCA, 2018, pp. 3333–3337.

[21] X. Chen, T. Tan, X. Liu, P. Lanchantin, M. Wan, M. J. F. Gales, and P. C. Woodland, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition," in *Interspeech 2015, Dresden, Germany, September 6-10, 2015*. ISCA, 2015, pp. 3511–3515.

[22] U. Khandelwal, H. He, P. Qi, and D. Jurafsky, "Sharp nearby, fuzzy far away: How neural language models use context," in *ACL*, Jul. 2018, pp. 284–294.

[23] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR, San Diego, CA, USA*, Y. Bengio and Y. LeCun, Eds., 2015.

[24] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *NAACL-HLT 2016*. San Diego, California: Association for Computational Linguistics, Jun. 2016.

[25] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016.

[26] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[27] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?" *CoRR*, vol. abs/1905.05583, 2019.

[28] M. Kalimuthu, A. Mogadala, M. Mosbach, and D. Klakow, "Fusion models for improved image captioning," in *Pattern Recognition. ICPR International Workshops and Challenges*, ser. Lecture Notes in Computer Science, vol. 12666, 2020, pp. 381–395.

[29] M. Sunkara, S. Ronanki, D. Bekal, S. Bodapati, and K. Kirchhoff, "Multimodal semi-supervised learning framework for punctuation prediction in conversational speech," in *Interspeech 2020, Shanghai, China, 25-29 October 2020*. ISCA, 2020, pp. 4911–4915.

[30] A. Sriram, H. Jun, S. Satheesh, and A. Coates, "Cold fusion: Training seq2seq models together with language models," in *Interspeech, Hyderabad, India*. ISCA, 2018, pp. 387–391.

[31] A. Rastogi, X. Zang, S. Sunkara, R. Gupta, and P. Khaitan, "Towards Scalable Multi-domain Conversational Agents: The Schema-Guided Dialogue Dataset," *arXiv e-prints*, p. arXiv:1909.05855, Sep. 2019.

[32] M. Eric, R. Goel, S. Paul, A. Sethi, S. Agarwal, S. Gao, and D. Hakkani-Tür, "Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines," *CoRR*, vol. abs/1907.01669, 2019.

[33] D. Peskov, N. Clarke, J. Krone, B. Fodor, Y. Zhang, A. Youssef, and M. Diab, "Multi-domain goal-oriented dialogues (MultiDoGO): Strategies toward curating and annotating large scale dialogue data," in *Proc EMNLP-IJCNLP*, 2019, pp. 4526–4536.

[34] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling." in *ICASSP*. IEEE Computer Society, 1995, pp. 181–184.