DialogStitch: Synthetic Deeper and Multi-Context Task-Oriented Dialogs

Satwik Kottur[*1], **Chinnadhurai Sankar**[*1], **Zhou Yu**[2†], **Alborz Geramifard**[1]
[1]Facebook AI [2]Columbia University
{skottur,chinnadhurai,alborzg}@fb.com, zy2461@columbia.edu

Abstract

Real-world conversational agents must effectively handle long conversations that span multiple contexts. Such context can be interspersed with chitchat (dialog turns not directly related to the task at hand), and potentially grounded in a multimodal setting. While prior work focused on the above aspects in isolation, there is a lack of a unified framework that studies them together. To overcome this, we propose *DialogStitch*, a novel framework to seamlessly 'stitch' multiple conversations and highlight these desirable traits in a task-oriented dialog. After stitching, our dialogs are provably *deeper*, contain *longer-term dependencies*, and span *multiple contexts*, when compared with the source dialogs— all by leveraging existing human annotations! Though our framework generalizes to a variety of combinations, we demonstrate its benefits in two settings: (a) multimodal, image-grounded conversations, and, (b) task-oriented dialogs fused with chit-chat conversations. We benchmark state-of-the-art dialog models on our datasets and find accuracy drops of (a) 12% and (b) 45% respectively, indicating the additional challenges in the stitched dialogs. Our code and data are publicly available[1].

1 Introduction

Task-oriented dialog agents have become increasingly popular in the recent years due to their ready deployment to several real-world applications. For such agents to be effective, they need to carryout long conversations spanning multiple contexts, interspersed with social chit-chat, and potentially grounded in multimodal settings.

Though prior works propose several datasets and task formulations to model these desired traits, we believe that they fall short on two counts. They either study these traits in isolation or in a simplified setting that does not cover the spectrum of requirements for real-world applications. The well-known task-oriented datasets MultiWOZ (Budzianowski et al., 2020) and Google Schema Guided (Rastogi et al., 2020) datasets contain only 13.4 and 20.4 turns respectively, on an average. While adequate for their intended purposes (*e.g.*, find a restaurant or book a flight), these datasets do not support modeling task-oriented agents that need to go beyond and handle longer conversations (also argued by Roller et al. (2020)). For instance, a real world customer service task might require conversations that last for hours, thus requiring more than 20 turns.

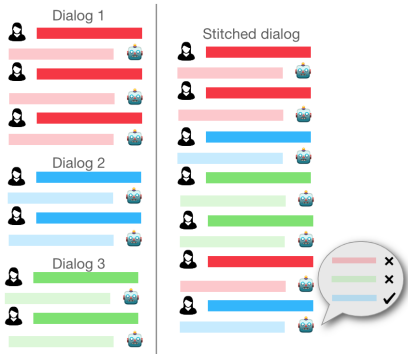

Figure 1: *DialogStitch* combines multiple dialogs together making them *longer*, contain *longer term dependencies*, and span *multiple contexts*—desirable for a task-oriented, multimodal conversational agent—without any additional annotation cost.

As a step to bridge these gaps, we propose *DialogStitch*, a novel framework that takes existing dialog dataset and creates dialogs that comparatively are *longer*, contain *longer-term* dependencies, and span *multiple contexts*. Unlike existing works that either combine dialogs using human annotators (Smith et al., 2020; Moirangthem and Lee, 2018), our framework imparts these desirable traits to task-oriented dialogs by using the available human annotations without collecting any additional ones and thus free of cost, due to its synthetic

[*] Joint first authors
[†] Work done with ZY was visiting Facebook AI
[1] github.com/facebookresearch/DialogStitch

Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 21–26
July 29–31, 2021.

(a) DialogStitch on CLEVR-Dialog. (b) DialogStitch on MultiWOZ 2.2 + WoW

Figure 2: Examples of dialogs generated by DialogStitch, spanning multiple contexts (red, blue, green) for both our settings (Sec. 3, 4). (a) Images (left) denote contexts in the stitched dialog. Context switch happens with the introduction of a new context (U-9, U-21) or at a context recaller question that typically refers back to an object in the scene (U18: *aforementioned green thing*, U-30: *earlier cylinder*). Though there could be similar objects (*other cylinders*) in other contexts, the object mention is unique and unambiguous in the dialog, making the DialogStitch output consistent and coherent task-oriented dialogs. (b) Context switch between task-oriented and chit-chat turns.

nature. As shown in Fig. 1, DialogStitch takes multiple dialogs and interleaves them carefully to ensure the resultant dialog is coherent, consistent, and more closely resembles the real-world scenarios. As the cherry on top, DialogStitch allows for the construction of dialog tasks analogous to the *copying memory task* (Hochreiter and Schmidhuber, 1997), a synthetic task to benchmark model's capability to retain information over many time steps, *i.e.*, modeling long-term dependencies.

To summarize our contributions:

- We propose *DialogStitch*, a novel framework to create task-oriented dialogs that are *longer*, contain *longer-term* dependencies, and handle *multiple contexts* by leveraging existing annotations.
- We show the effectiveness of our approach in two settings: stitching multimodal (image-grounded) conversations, and task-oriented with open-domain conversations.
- We benchmark the state-of-the-art models on our datasets to serve as baselines for future research.

2 Our Approach

Consider a set of K dialogs $\{\mathbf{D_i}\}^{\mathbf{1:K}}$ where each dialog $\mathbf{D_i}$ consists of n_i turns with each turn $T_i^j = (u_i^j, s_i^j)$ containing a *user* and a *system* utterance respectively. Each dialog can also have a turn-independent[2] multimodal context M_i, for example, an image in which the dialog is grounded. As shown in Fig. 1, DialogStitch interleaves dialogs by inserting turns from one dialog into another. The exact strategy to interleave dialogs is domain-specific and uses the additional annotations accompanying the source datasets. However, care is taken to ensure that: (a) the user and system utterance in a turn are not separated, though the turns themselves are interleaved, (b) after stitching, the ordering among the turns in each dialog is preserved in the final dialog to avoid inconsistencies, and (c) no ambiguity (*e.g.*, multiple referents for coreference, values for slots) results from this process of stitching. Hence the resulting dialog is meaningful and coherent.

The stitched dialog $\text{DS}(\{\mathbf{D_i}\}^{\mathbf{1:K}})$ has the following properties: (a) it has $\sum_{i=1}^{K} n_i$ turns, *deeper* than each of the individual source dialogs $\mathbf{D_i}$, (b) the gap between the turns of any dependency (*e.g.*, coreference, slot carryover) in a dialog D_i will only increase on an average since new turns from other dialogs would separate them further, thus making the dependencies *longer-term*, (c) it spans *multiple contexts* $\{M_i\}^{1:K}$. Note that there is no additional human annotation required and all the above benefits are solely due to our novel framework, and thus free of cost. We demonstrate the effectiveness of DialogStitch by instantiating it in two settings: multimodal, image-grounded conversations (Sec. 3), and, task-oriented dialogs fused with chit-chat conversations (Sec. 4).

3 Stitching Multimodal Dialogs

We showcase the ability of DialogStitch to handle and stitch dialogs with complex multi-round

[2]Our framework readily extends to turn-dependent multimodal context M_i^j. For brevity, we only discuss the simpler scenario here.

reasoning spanning across different multimodal contexts using the CLEVR-Dialog dataset (Kottur et al., 2019). CLEVR-Dialog is a visually-simple yet reasoning-wise complex visual dialog (Das et al., 2017) dataset, which contains a series of related question-answers pairs as dialog turns. These questions are grounded in an image, set in the abstract CLEVR world (Johnson et al., 2017), and is made of spatially arranged objects (with shape, size, material, color attributes) against a plain background (see Fig. 2a). By design, dialogs in CLEVR-Dialog have strong multi-turn dependencies. In addition, these dialogs also come with complete state annotations like type of question, objects/attributes of interest, and coreferences, for each turn. These two reasons make CLEVR-Dialog a perfect testbed for DialogStitch.

DialogStitch on CLEVR-Dialog. Each dialog $\mathbf{D_i}$ in CLEVR-Dialog starts off with a caption C_i that partially describes the image, followed by 10 question-answer pairs $(Q_i^j, A_i^j)^{1:10}$, as illustrated in Fig. 1. To align with our framework in Sec. 2, we treat the caption as the first turn with an empty assistant utterance $T_i^0 = (C_i, \emptyset)$, and the question-answer pairs as following turns, *i.e.*, $T_i^j = (u_i^j, s_i^j) = (Q_i^j, A_i^j)$.

To stitch K different dialogs together, we: (a) identify the *recaller* questions that can help us recall their corresponding multimodal context (image) in the stitched dialog, using the question type annotations. These questions (with `early` tag) typically contain a reference to previously mentioned objects in the dialog, for example, *'What size is the earlier cylinder?'*. Refer (Kottur et al., 2019) for a full list of question types and tags in CLEVR-Dialog. (b) breakdown each dialog into 2–3 chunks at randomly selected *recaller* question pivots. For each of these chunks, we note all the objects and attributes mentioned in the dialog so far. Note that this is possible only due to the available annotations. (c) starting with the first chunk of a randomly selected dialog, we select a chunk from dialogs different from the one previous selected as a candidate. We then check for stitch compatibility by ensuring that there is no overlap of objects and attributes mentioned in both the stitched dialog and the candidate. If compatible, we append the candidate at the end and repeat the process, else discard and re-select a new one. Note that when selecting chunks from a dialog, priority is given to the one that appear earlier. This ensures that the resultant stitched dialog respects the turn ordering from all the source dialogs and is coherent.

Model	Source	DS (Ours)
VB-Q	39.1	39.3
VB-QI	52.7	53.0
VB-QH	45.8	50.2
VB-QIH	68.2	56.5

Table 1: Accuracy of VisDial-BERT on CLEVR-Dialog (source), CLEVR-Dialog+ (DS).

Stitched Dataset. CLEVR-Dialog comprises $85k$ images x 5 dialogs per image x 10 question-answer pairs per image = $4.25M$ question-answer pairs, split into `train` (82%) and `val` (18%). We set $K = 3$ and run DialogStitch to obtain CLEVR-Dialog+. For a fair comparison, we keep the number of question-answer pairs constant between the datasets. As a result, CLEVR-Dialog+ contains $142k$ dialogs x 30 question-answer pairs per dialog = $4.25M$ question-answer pairs, split proportionally into `train` and `val`. Note that stitching is performed without cross data contamination, *i.e.*, dialogs for `train` of CLEVR-Dialog+ are sampled from CLEVR-Dialog `train`, and similarly for `val`. CLEVR-Dialog+ dialogs are trivially $3\times$ deeper, contain $3\times$ the number of multimodal contexts, and most importantly, have longer range dependencies ($2\times$ mean coreference distance of 5.6 vs. 3.2), when compared with CLEVR-Dialog.

Experiments and Metrics. To benchmark performance on CLEVR-Dialog+, we select the state-of-the-art visual dialog model, VisDial-BERT (Murahari et al., 2020), and adapt it to our setting. Following Kottur et al. (2019), we ablate VisDial-BERT (VB) to model different valid combinations of the question (Q), history (H), and image (I) for the given dialog. We use answer accuracy, similar to CLEVR-Dialog, to compare the these models. Implementation and adaption details are in supp.

Results. Tab. 1 shows the performance of VB (and ablations) on both CLEVR-Dialog (source) and CLEVR-Dialog+ (DS). Key observations are:

- As expected, Q models perform the worst on both the source and DS datasets, followed by QH models that are also blind (no access to image).
- Surprisingly, the gap between Q and QH models is larger for DS (10% vs 6.7%) than source, even though DS has irrelevant turns in its history. A possible explanation is that since dialogs are stitched together ensuring there is no overlap of attributes/objects, it gives away information that the models are able to leverage.
- As DialogStitch reorganizes the dialog history, history-agnostic models (Q, QI) have similar performances on both source and DS.

Corpus	#Turns(Avg)	JGA w/o	Slot-P/R w/o	JGA w/	Slot-P/R w/
MWOZ-2.2	13.4	$55.3_{\pm 0.1}$	$95.2_{\pm 0.2}$ / $0.93.8_{\pm 0.1}$	-	-
MWOZ-2.2 + DailyDialog	21.3	$53.3_{\pm 1.0}$	$91.2_{\pm 0.2}$ / $87.4_{\pm 0.4}$	$45.4_{\pm 2.0}$	$92.0_{\pm 1.3}$ / $82.1_{\pm 1.3}$
MWOZ-2.2 + WoW	22.5	$51.3_{\pm 0.7}$	$91.3_{\pm 0.6}$ / $88.0_{\pm 0.8}$	$45.7_{\pm 1.9}$	$91.8_{\pm 1.5}$ / $82.6_{\pm 1.5}$
MWOZ-2.2 + PersonaChat	28.2	$48.3_{\pm 1.7}$	$88.3_{\pm 1.3}$ / $83.2_{\pm 1.9}$	$44.4_{\pm 1.5}$	$88.2_{\pm 1.2}$ / $80.9_{\pm 1.0}$
MWOZ-2.2 + WoW + DailyDialog	30.4	$38.7_{\pm 3.1}$	$83.2_{\pm 4.0}$ / $75.3_{\pm 2.9}$	$15.5_{\pm 2.5}$	$44.7_{\pm 5.6}$ / $29.3_{\pm 4.7}$
MWOZ-2.2 + WoW + PersonaChat	37.3	$30.6_{\pm 1.0}$	$77.7_{\pm 1.2}$ / $69.5_{\pm 2.6}$	$22.4_{\pm 2.3}$	$69.2_{\pm 3.2}$ / $63.9_{\pm 3.4}$
Schema	20.4	$53.0_{\pm 0.6}$	$93.8_{\pm 0.7}$ / $74.4_{\pm 0.3}$	-	-
Schema + WoW	29.5	$49.8_{\pm 1.5}$	$91.2_{\pm 0.4}$ / $73.0_{\pm 2.2}$	$46.6_{\pm 0.1}$	$89.2_{\pm 0.3}$ / $71.1_{\pm 0.9}$

Table 2: Joint Goal Accuracy (JGA) (%) & Slot-Precision/Recall (%) of various stitched datasets with the SimpleTOD (Hosseini-Asl et al., 2020) model. We report mean and std-dev across 3 runs. JGA w/ $\rightarrow$ model trained to generate both dialog states and chit-chat responses & JGA w/o $\rightarrow$ only dialog states. With Dialog Stitch, the avg. dialog-state dependency (turn-id of the utterance corresponding to each dialog-state) increased from 6.33 to 8.97).

- Performance improves when models have access to H and I, confirming importance for the task.
- QIH outperforms all other models in both the cases. However, the lead is only 6.3% for DS vs 15.5% for source. Further, QIH model on DS is inferior to that of source by a huge 11.7% points. This shows the additional challenges in the stitched dialog that are deeper, have longer dependencies, and span multiple contexts.

4 Stitching Open-Domain Dialogs

Being socially engaging is a desirable trait for task-orientated dialog agent as it facilitates a wider adoption in everyday applications. To achieve this, agents must additionally handle chit-chit about social topics. We emulate these scenarios to synthetically stitch task-oriented and open-domain dialogs.

Datasets. We adopt the ParlAI framework (Miller et al., 2017) as a testbed for DialogStitch, since it grants a unified access to a vast repository of both open-domain and task-oriented dialog datasets. Though DialogStitch is easily extendable to all these datasets within ParlAI, we consider the following datasets (see supp. for dataset statistics):

- **Task-Oriented:** MultiWOZ 2.2 (Zang et al., 2020) and Schema Guided (Rastogi et al., 2020)
- **Open-Dialog:** Wizard Of Wikipedia (WoW) (Dinan et al., 2019), PersonaChat (Zhang et al., 2018), and DailyDialog (Li et al., 2017)

Stitched Datasets. Similar to multimodal Stitched datasets described in Sec. 3, we divide the dialogs into multiple chunks (2-5) at randomly selected *pivot* turns and take the following precautions while fusing them into a single conversation.

- The context switch at the *pivot* turns is always initiated by the user utterance.
- For coherency, we use conversational cues to indicate a context-switch turn (*e.g.*, *'getting back to the restaurant booking'*) from task-oriented to open-domain, and vice-versa.
- Additionally, we re-sample a pivot if the open-domain assistant turn preceding asks a question. This avoids dialogs where the user changes context instead of responding to the question asked by the assistant, thus improving naturalness.

To generate longer conversations and multiple contexts, we can configure DialogStitch to stitch a task-oriented dialog with multiple open-domain dialogs within the same conversation.

Human Evaluation. To evaluate the quality, we compare 50 stitched dialogs with corresponding human stitched dialogs (where human annotators manually stitch the task-oriented and a chit-chat dialog chosen from three options). Overall, humans found our stitched dialogs to be 54% coherent and 66% natural compared to the human stitched dialogs (74% coherence, 72% naturalness). This indicates that our stitched dialogs trade coherence and naturalness reasonably with annotation cost.

Experiments and Metrics. We benchmark the stitched datasets using the SimpleTOD model (Hosseini-Asl et al., 2020). to generate the dialog states (SlotType-SlotValue, *e.g.*, *Cuisine-Italian, Time-5pm*) and the next utterance given the conversation history. We track dialog states using Slot-precision & recall (Slot-P/R) and joint goal accuracy (JGA). JGA computes the percentage of the turns in which the model correctly predicts all the dialog states corresponding to that turn. Following (Hosseini-Asl et al., 2020), we truncate the dialog history to 1024 tokens. See supp. for more details.

Observations. We observe that the JGA consistently drops with increasing dialog length (Tab. 2). For instance, JGA drops from 55.3% to 30.6% when fused with WoW and PersonaChat datasets. It drops further when the model is also tasked to engage in open-domain dialogs. When trained to additionally generate responses for a dialog context, JGA drops from 53.3% to 45.4% (DailyDialog).

Conclusion. *DialogStitch* generates dialogs that are *longer*, involve *multiple contexts*, and contain *longer term dependencies* compared to prior work. Performance of state-of-the-art models drops when benchmarked on our datasets, thus suggesting a need to better model multiple-contexts and longer-

term dependencies. We hope it stimulates research in designing architectures and training techniques adept at deep conversations amid the dearth of crowd-sourced datasets with longer contexts.

A Implementation Details

Multimodal Dialogs. Our DialogStitch is implemented entirely in Python, without any other significant package dependencies. To train Visdial-BERT (Murahari et al., 2020), we use the provided open source implementation[3] built on PyTorch (Paszke et al., 2019). Visdial-BERT uses bottom-up, top-down (BUTD) image features (Anderson et al., 2018) for images. We use publicly available BUTD features[4] for CLEVR images, thanks to (Shrestha et al., 2019). Similar to (Kottur et al., 2019), we set aside a subset (500 images) of the `train` and use it to pick the best performing models via early stopping. We follow the steps below to adapt Visdial-BERT to CLEVR-Dialog+:

- VisDial-BERT augments the question at a particular turn with image features and dialog history, and then concatenates with ground-truth answer to predict a binary positive class for the alignment. Negative instances are selected by randomly pairing the question + image + dialog history with other answers in a given batch of training. In our work, we replace this binary classifier and replace it with a N_A-way classifier head, where $N_A = 29$ is the size of the output answer space for CLEVR-Dialog.
- Since CLEVR-Dialog contains templated language, the weight for the masked language prediction loss is reduced by 50% each epoch.
- Due to the longer nature of CLEVR-Dialog+, a small percent of the dialogs (1%) were longer than 512 tokens. In these cases, we simply remove an equal number of tokens from the start of the dialog to clip the total length to 512 tokens.

Rest of the hyperparameters are kept similar to (Murahari et al., 2020). We perform all our experiments on 8 NVIDIA Tesla V100 GPUs.

B Further Details: Stitching Open Dialogs

Model Details. SimpleTOD (Hosseini-Asl et al., 2020) builds a dialog model by fine-tuning GPT2 (Radford et al., 2019), a large pre-trained language model. It combines dialog history, previous dialog states and user utterance into a single sequence as input and let the language model learn to generate a sequence, containing dialog states and system response.

[3] https://github.com/vmurahari3/visdial-bert

[4] https://github.com/erobic/ramen

Corpus	Dialogs	#turns	Turns(Avg.)	Domain/Topics
MultiWOZ-2.2	10,420	71,410	13.4	7
Schema	22,825	463,284	20.4	17
DailyDialog	13,118	103,632	7.9	10
WoW	21,343	193,217	9.1	1,247
PersonaChat	10,907	162,064	14.8	1,155

Table 3: Statistics for the datasets used in this work.

Experimental Setup. We perform all our experiments using a single NVIDIA P100 16GB GPU. We train with a batch-size of 8 with a learning rate of $1e-4$, adam optimizer with hyper-parameters in (Radford et al., 2019) and set the training time to 6000 secs with validation performed every epoch. Following (Hosseini-Asl et al., 2020), we truncate in the input and output sequences to 1024.

Human Evaluation Setup We compiled a list of 60 stitching tasks where the annotator manually stitches a task-oriented (MultiWOZ 2.2) and chit-chat (Wizard of Wikipedia). The annotators could either start the conversation with either a task-oriented or chit-chat turn but need to exhaust all turns while maintaining order of the turns. In the second part of the experiment, the human stitched dialogs and our stitched dialogs were compared by three independent annotators with respect to naturalness and coherency.

Approach to Retrieving Relevant Open-Domain Dialogs. Certain open-domain dialogs like WoW and PersonaChat are annotated with the topic of the conversation. We also have the option in *DialogStitch* to only fuse open-domain dialogs with topics relevant to the task-oriented domain. See supp. for details. We curate a set of relevant keywords (e.g., italian cuisine) related to the task-oriented dialog domain (e.g., restaurant) and use them filter the open-domain dialog based on overlapping keywords and topics. In our human evaluation experiment where human annotators picked the relevant dialog based on the technique mentioned above 55% (random 33%) times when presented with four chit-chat dialogs to blend with the task-oriented dialog. We leave the task exploring more techniques of finding in-domain open-dialog conversations from a given dataset to the future work.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2020. Multiwoz – a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M.F. Moura, Devi Parikh, and Dhruv Batra. 2017. Visual Dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*.

Satwik Kottur, José M. F. Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. 2019. CLEVR-dialog: A diagnostic dataset for multi-round reasoning in visual dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 582–595, Minneapolis, Minnesota. Association for Computational Linguistics.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset.

A. H. Miller, W. Feng, A. Fisch, J. Lu, D. Batra, A. Bordes, D. Parikh, and J. Weston. 2017. Parlai: A dialog research software platform. *arXiv preprint arXiv:1705.06476*.

Dennis Singh Moirangthem and Minho Lee. 2018. Chat discrimination for intelligent conversational agents with a hybrid CNN-LMTGRU network. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 30–40, Melbourne, Australia. Association for Computational Linguistics.

Vishvak Murahari, Dhruv Batra, Devi Parikh, and Abhishek Das. 2020. Large-scale pretraining for visual dialog: A simple state-of-the-art baseline. In *The European Conference on Computer Vision (ECCV)*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset.

Stephen Roller, Y-Lan Boureau, Jason Weston, Antoine Bordes, Emily Dinan, Angela Fan, David Gunning, Da Ju, Margaret Li, Spencer Poff, Pratik Ringshia, Kurt Shuster, Eric Michael Smith, Arthur Szlam, Jack Urbanek, and Mary Williamson. 2020. Open-domain conversational agents: Current progress, open problems, and future directions.

Robik Shrestha, Kushal Kafle, and Christopher Kanan. 2019. Answer them all! toward universal visual question answering models. In *CVPR*.

Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents' ability to blend skills.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. Multiwoz 2.2 : A dialogue dataset with additional annotation corrections and state tracking baselines.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too?