

# Navigating Connected Memories with a Task-oriented Dialog System

Anonymous EMNLP submission

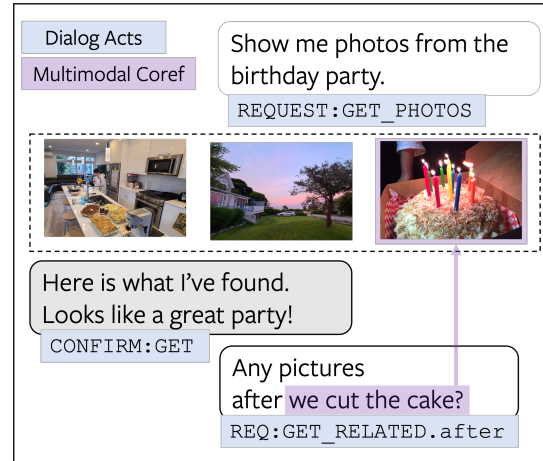
## Abstract

Recent years have seen an increasing trend in the volume of personal media captured by users, thanks to the advent of smartphones and smart glasses, resulting in large media collections. Despite conversation being an intuitive human-computer interface, current efforts focus mostly on single-shot natural language based media retrieval to aid users query their media and re-live their memories. This severely limits the search functionality as users can neither ask follow-up queries nor obtain information without first formulating a single-turn query.

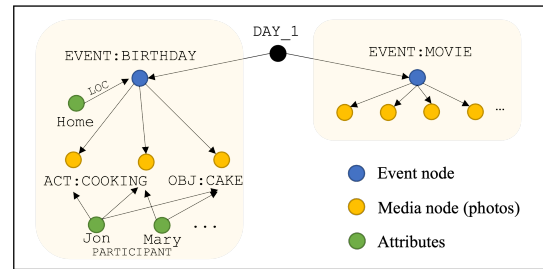
In this work, we propose *dialogs for connected memories* as a powerful tool to empower users to search their media collection through a multi-turn, interactive conversation. Towards this, we collect a new task-oriented dialog dataset COMET, which contains 11.5k user↔assistant dialogs (totalling 103k utterances), grounded in simulated personal memory graphs. We employ a resource-efficient, two-phase data collection pipeline that uses: (1) a novel multimodal dialog simulator that generates synthetic dialog flows grounded in memory graphs, and, (2) manual paraphrasing to obtain natural language utterances. We analyze COMET, formulate four main tasks to benchmark meaningful progress, and adopt state-of-the-art language models as strong baselines, in order to highlight the multimodal challenges captured by our dataset. Our code & data will be made publicly available.

## 1 Introduction

The rise of smartphones and smart glasses has contributed to a surge in the amount of personal media (photos, videos, montages, *etc.*) captured by users on a day-to-day basis in the past decade. For instance, it is estimated that about 1.5 trillion photos would be clicked in the year 2022 (Pantic, 2021). As a result, personal media collections typically grow at an alarming rate, making it cumbersome for users to manually search, retrieve, and re-live



(a) Dialogs for Connected Memories



(b) Memory Graph

Figure 1: Illustration of COMET: **C**ONNECTED **M**EMORIES with a **T**ASK-oriented **D**IALOG. (a) Each dialog turn is fully annotated with dialog acts and multimodal coreference labels, accompanied with photos associated with the request. (b) These media are from the underlying memory graph, a structured collection of personal media.

their captured memories<sup>1</sup>.

To alleviate this situation, solutions that perform natural language query-based media retrieval (Tan et al., 2019; Vo et al., 2019; Tellex and Roy, 2009; Barbu et al., 2013; Li et al., 2017; Guo et al., 2018a; Saha et al., 2018) have been proposed. However, such approaches exhibit two drawbacks. First, they are single-shot interactions without any context carry-over, *e.g.*, *Show me some photos from the beach last week.* This limits the functionality and does not let users ask any follow-up queries like

<sup>1</sup>Memories and media files are used interchangeably.

053 ‘Display photos from the first time I was here?’, 105  
054 since understanding here requires the query history. 106  
055 Second, users cannot seek information without ac- 107  
056 tually formulating the query to retrieve the corre- 108  
057 sponding memory. For instance, there is no easy 109  
058 query to know the first time a user visited the beach 110  
059 in the memory they are reviewing. 111

060 In order to overcome these limitations, we pro- 112  
061 pose *dialogs for connected memories* as a powerful 113  
062 interface where users can interactively query their 114  
063 memory collections. By design, a conversational 115  
064 agent can handle multi-turn interactions enabling 116  
065 several additional queries that require context car- 117  
066 rryover, e.g., ‘When was the first time I was at this 118  
067 beach?’. Though prior efforts have explored the 119  
068 use of dialogs in media retrieval (Wu et al., 2021; 120  
069 Guo et al., 2018b) in other domains (e.g., fashion), 121  
070 there is no existing work focusing on interactive 122  
071 search and query of personal media collections to 123  
072 the best of our knowledge. 124

073 More concretely, we propose COMET, a new 125  
074 multimodal task-oriented dialog dataset aimed at 126  
075 developing conversational assistants that can en- 127  
076 able users to interactively search and query their 128  
077 collection of memories. Working with personal me- 129  
078 dia collections presents two main obstacles: (a) 130  
079 There are no readily available public datasets that 131  
080 contain personal media along with associated me- 132  
081 dia attributes that we could leverage, and, (b) Per- 133  
082 sonal memories constitute sensitive information, 134  
083 thus resulting in privacy and safety concerns. To 135  
084 circumvent these roadblocks, we devise a novel 136  
085 memory graph simulator that can leverage publicly 137  
086 available media datasets and help create several 138  
087 synthetic memory collections. We represent these 139  
088 collections as memory graphs to capture useful 140  
089 relationships between the constituent memories, 141  
090 e.g., memories taken at the same place. We then 142  
091 collect 11.5k user↔assistant task-oriented dialogs 143  
092 (totalling 103k utterances), grounded in 1.1k mem- 144  
093 ory graphs. An example dialog is shown in Fig. 1. 145  
094

095 Our dataset is challenging as it requires rea- 146  
096 soning through both the dialog history and mul- 147  
097 timodal context (memory graphs) to resolve core- 148  
098 ferences, track the dialog state, predict the right 149  
099 API, and generate a meaningful natural language 150  
100 assistant response. As an example, consider the 151  
101 query ‘When was the first time I was here?’. First, 152  
102 the model needs to resolve here using the dialog 153  
103 history and previously viewed memories. Next, it 154  
104 needs to understand that the query is seeking in-

formation about a *connected memory*, and predict 105  
the right API `get_time(resolve(here), first` 106  
`time)`. Finally, it should produce a response like 107  
‘The first time you were here was on August 2, 2019 108  
with Jean’, potentially including some chit-chat. 109

To capture these challenges and benchmark 110  
progress towards assistants that can interactively 111  
handle dialogs for connected memories, we formu- 112  
late four main tasks: Assistant API Call Prediction, 113  
Multimodal Coreference Resolution (MM-Coref), 114  
Multimodal Dialog State Tracking (MM-DST), and 115  
Response Generation. We train baseline models for 116  
these tasks, and discuss future research directions. 117

## 2 Related Work 118

**Task-oriented Dialogs** aim to understand user 119  
queries and accomplish a pre-defined set of tasks 120  
(e.g. booking hotels), which is a popular setting 121  
in consumer-facing virtual assistants. Our work 122  
addresses similar challenges often found in other 123  
task-oriented dialogs, such as natural language un- 124  
derstanding (NLU), dialog state tracking (DST) 125  
(Henderson et al., 2014), etc. Compared to the con- 126  
ventional task-oriented dialog datasets (e.g. Multi- 127  
WoZ (Budzianowski et al., 2018; Eric et al., 2019; 128  
Rastogi et al., 2019)), however, our work involves 129  
a unique multimodal setting where dialogs are 130  
grounded on a memory graph composed of sev- 131  
eral media files, introducing novel challenges such 132  
as Multimodal DST and Multimodal Coreference 133  
Resolution given personal photo collections. 134

The most notable modeling approaches for task- 135  
oriented dialog systems include casting the DST 136  
task as a joint causal language modeling problem 137  
(Hosseini-Asl et al., 2020; Peng et al., 2020; Gao 138  
et al., 2019), by fine-tuning a large pre-trained trans- 139  
formers such as GPT-2 (Radford et al., 2019). We 140  
follow this recent trend and provide baselines by ex- 141  
tending it accommodate for the unique multimodal 142  
contexts that our dataset brings. 143

**Multimodal Dialogs** have become increasingly 144  
more popular, where the dialog model processes 145  
both visual and text input to handle queries. Many 146  
existing literature (Hori et al., 2018; Das et al., 147  
2017; Kottur et al., 2019; de Vries et al., 2017, 148  
2018; Thomason et al., 2019) study multimodal 149  
Q&A dialogs grounded on a single image as multi- 150  
modal context, extending the conventional Visual 151  
Question Answering (Antol et al., 2015) tasks to 152  
multi-turn scenarios. We extend this line of work 153  
by studying the multimodal agent that operates 154

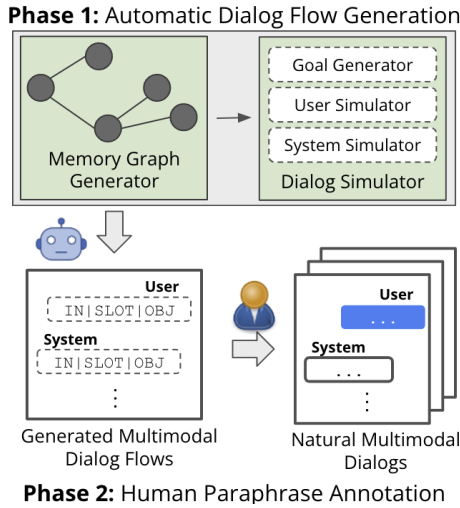


Figure 2: Two-stage pipeline to collect dialogs for COMET. See Sec. 3 for more details.

on a collection of media (memory graph), thus requiring reasoning abilities over multiple grounding contexts. In addition, our focus on task-oriented dialogs extends the previous literature that primarily focuses on retrieval tasks (Guo et al., 2018a; Saha et al., 2018; Firdaus et al., 2020), and studies understanding structured user intents and fine-grained attributes annotated for each multimodal query.

**Memory QA:** Our work is also similar to the Memory QA tasks (Jiang et al., 2018; Moon et al., 2019), where the main task is to answer user QA queries upon a collection of images, extending the Visual QA task (Antol et al., 2015) which operates on a single image. However, the existing literature is limited to a simple single-turn QA interaction, and focuses on the identification of an evidential image to answer a question. While our dataset does include QA queries, we extend the problem domain to the conversational settings which support complex scenarios (e.g. searching for related memories), allowing for rich multimodal interactions.

### 3 The COMET Dataset

COMET is aimed to enable assistant systems that can process interactive queries from users and help navigate their collection of memories through a natural language conversation. Towards this, we collect the COMET dataset using a two-phase approach (shown in Fig. 2): (a) Generating synthetic dialog flows between a user and an assistant that are conditioned on memory graphs, using a novel multimodal dialog simulator (Sec. 3.1), and, (b) Manually paraphrasing the above flows to obtain dialogs with natural language utterances (Sec. 3.2),

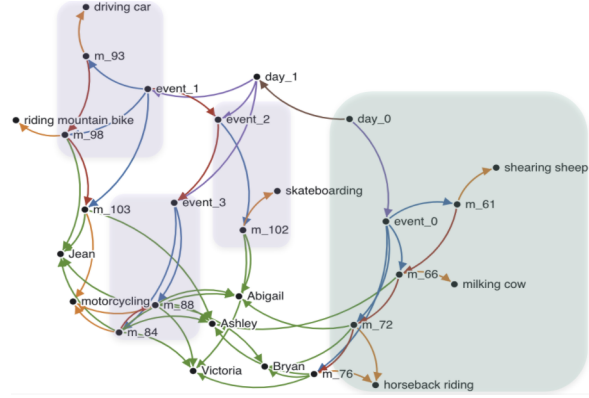


Figure 3: Memory subgraph with constituent memories and their hierarchical relationships. Each memory contains activity (orange), people (green), time, and place (not shown) attributes. Memories are grouped into events (purple box), then days (green box), and finally trips (shown subgraph). Each memory graph contains multiple trips, though only one is shown here for brevity.

thus moving closer to real-world application. This approach is resource-efficient as it reduces the annotation overheads when compared to collecting human↔human dialogs, both in terms of cost and time. In what follows, we describe these two phases in detail and analyze our COMET dataset. See the supplementary (Fig. 7) for an example dialog.

### 3.1 Multimodal Dialog Self-play

We first leverage a multimodal dialog simulator (Sec. 3.1.2) to generate synthetic dialog flows between a user and an assistant. Each of these flows is grounded in a graph connecting the memories of a user from their collection. The memory graphs in our work are simulated by a novel graph simulator (Sec. 3.1.1) and are designed to capture several hierarchical relationships between the user memories.

#### 3.1.1 Memory Graph Simulator

Graphs have been ubiquitously used in various fields to effectively represent a set of entities and relationships between them. Following this trend, we use a graph structure to represent a collection of memories (see Fig. 3 for an example). As mentioned in Sec. 1, to circumvent the lack of readily available datasets for personal photo collections and surrounding privacy issues, we construct a novel graph simulator to synthetically generate memories graphs using public datasets. These memory graphs are then used as an input to the multimodal dialog simulator to generate dialog flows.



**Memories and Attributes.** Memories constitute the atomic units of the graph simulator, and can cover a wide variety of media including photographs, videos, and user-created montages. We limit the scope of memories to represent static images in this work, although most components of our proposed framework readily extend to the broader definition. As photo collection of individuals is sensitive information, we use publicly available image dataset as a proxy to mitigate the risk. Specifically, we use Creative Commons images from MS COCO (Lin et al., 2014) that contains objects and people in everyday contexts as memories.

We then assign four attributes to each of the images as follows: (a) *Activity*: Each image in MS COCO has 5 associated captions. We use sentence-BERT (Reimers and Gurevych, 2019) to find the closest activity label from the taxonomy of the ActivityNet dataset (Heilbron et al., 2015), using average text-similarity to the captions. To ensure a good representation, we only keep those with at least 20 memories resulting in about 138 labels covering wide variety of activities. (b) *Place*: For each activity, we first manually map it to a place type, which then is randomly mapped to an actual place from a manually curated list. For instance, *playing frisbee*  $\rightarrow$  *park*  $\rightarrow$  *Cal Anderson Park, Seattle, USA*. (c) *People*: We use the associated bounding box annotations for MS COCO images and map those labeled as ‘person’, above a threshold size, to a random name from a curated list of 200 names. (d) *Time* attribute is sampled randomly from a constrained time range, depending on the relationship shared with other memories in the graph.

**Hierarchical Relationships.** To closely emulate scenarios in a personal photo collection, we devise the following hierarchy of relations amongst the memories: *memories*  $\rightarrow$  *events*  $\rightarrow$  *days*  $\rightarrow$  *trips*. Using heuristic rules, we sample and group memories into events that are then grouped into days, which are finally grouped into trips. These groupings impose constraints on the attributes of the constituent memories, which can be used to generate interesting conversational flows to query connected memories. For instance, memories from the same event need to happen at the same place type, while those in a day need to happen in the same city. Similar restrictions arise for the time attribute as well, which would be used to sample reasonable times for the corresponding memories, e.g., memories from the same event cannot be separated by more than few hours. These hierarchical relationships en-

able connected queries like ‘*What did we do after this?*’, ‘*Show other pictures with Jane on this trip*’, or ‘*Where did we go the next day?*’.

**Memory Graphs.** Putting everything together, we construct a memory graph for each collection:

- nodes: memory, event, day, trip, person, activity
- edges: memory attributes, hierarchical relations

Note that each memory graph can contain multiple trips. Fig. 3 illustrates a memory subgraph, visualizing only one trip for brevity. We synthetically generate multiple memory graphs which form the input to the dialog flow simulator.

**Applications in the Real-World Setting.** While we use the publicly available image dataset to generate memory graphs, applying the method above in the existing real-world photo album products at large-scale is straightforward as we do not require any additional information (e.g. captions or annotations) other than meta data that are readily associated with the media (e.g. timestamp, locations). This meta data can be rearranged from tables to graphs without additional annotations, only by specifying the relations of interest (e.g., people, place, time, predicted concepts). Memory graphs are not only practical but also desired to enable connected memory search.

### 3.1.2 Multimodal Dialog Simulator

The multimodal dialog simulator takes the generated memory graphs along with the meta information of each node to create user $\leftrightarrow$ assistant dialog flows, following the agenda-based dialog simulator approach (Schatzmann et al., 2007).

**Dialog Flow Generation via Self-play.** The dialog simulator comprises three main components: the *goal generator*, the *user simulator*, and the *assistant simulator*. The goal generator randomly samples an *agenda* for each dialog, which defines a sequence of high-level *goals* for the scenario (e.g., SEARCH  $\rightarrow$  GET\_RELATED\_PHOTOS  $\rightarrow$  GET\_INFO). Given a goal, the user simulator draws an acceptable dialog action based on a probability distribution, which is defined with NLU intents (e.g., REQUEST:GET, CONFIRM:SHARE), slots (e.g., location, time), and memory references. The assistant simulator then takes the output of the user simulator, retrieves the multimodal contexts via the simulation API (e.g. obtaining the information of a memory node from the graph, retrieving related memories), and generates natural language generation (NLG) intents, slots and new memory

references. The process is repeated until the simulator exhausts every goal in the agenda.

**Multimodal Dialog Ontology.** Following other task-oriented dialog datasets (Eric et al., 2019; Rastogi et al., 2019; Moon et al., 2020), for COMET we provide the standard dialog annotations such as the intent (NLU & NLG) and slot labels. To accommodate for the complex multimodal nature of the scenarios, we extend the dialog ontology to include memory reference annotations as their corresponding node IDs, which seamlessly annotates both multimodal contexts and language (e.g. ‘When was our trip to Whistler?’ → `INFORM:GET_INFO.time, memories: [8]`). The same notation can be used to refer the memories that are carried over in the dialog context (e.g. ‘Where did we go after that?’ → `INFORM:GET_RELATED.location, memories: [8]`). This proposed fine-grained and unified ontology will allow a systematic approach to study diverse referring expressions in multimodal dialogs.

### 3.2 Manual Paraphrase

Once the memory graph conditioned dialog flows have been generated, we paraphrase utterances in the dialog flow with the help of human annotators. This allows us to draw utterances from the natural language distribution, thus moving closer to the application. We build an interactive user interface to aid annotators paraphrase utterances from COMET dataset. Specifically, the interface shows the images corresponding to the memories along with the dialog flow and instructs annotators to paraphrase without losing key information such as objects and attributes. See appendix for an example dialog. As paraphrasing utterances is faster, cheaper, and requires little to no domain knowledge on the annotator’s part, our two-phase pipeline is much more resource-effective, when compared to collecting multimodal human↔human dialogs and collecting dialog annotations on top (Moon et al., 2020).

### 3.3 COMET Dataset Analysis

We now analyze the COMET dataset, which contains 11.4k dialogs totalling 103.4k utterances, grounded in 1.1k memory graphs. Tab. 1 presents the overall dataset statistics.

**Analyzing Dialogs.** Dialogs in COMET use a total of 1.1k memory graphs with each containing 100 memories. For every dialog, there are about 3.5

Total # dialogs	11.5k
Total # utterances	103.4k
Total # memory graphs	1.1k
Avg # words (user turns)	10.7 ± 4.4
Avg # words (assistant turns)	15.4 ± 9.8
Avg # utterances / dialog	8.8
Avg # memories mentioned / dialog	3.5
Avg # memories in graph / dialog	100

Table 1: COMET Dataset Statistics

connected memory mentions with the distribution given in Fig. 4b. User and assistant turns average about 10.7 and 15.4 words respectively (distribution in Fig. 4a). It is interesting to note that the assistant responses are significantly longer than the user. As an example, consider the following user utterance ‘U: Are there any similar photos from 2020?’ and the corresponding assistant response ‘A: Here’s one of Laura and Virginia cooking sausages at home, the afternoon of August 26, 2020. It looks like a fun time!’. This illustrates that the annotators paraphrasing the dialog flows included: (a) details about the retrieved memories to give additional context to the user, thus invoking subsequent connected memory queries (e.g., ‘What did we do that evening?’), (b) chitchat about the memories to make the conversational natural sounding.

**Analyzing Dialog Annotations.** Our COMET come with annotations at dialog level for dialog state tracking (NLU intents and slots), necessary API calls for assistant, and multimodal coreference resolution. Following Kottur et al. (2021), our intents follow a hierarchy of *dialog acts* (4: ASK, CONFIRM, INFORM, REQUEST) and *activities* (4: DISAMBIGUATE, GET, REFINE, SHARE). See Fig. 4d for a breakdown distribution over dialog acts and activities. Due to the retrieval nature of our assistant (either memories or associated attributes), a major chunk of the activities are GET. Similarly, there are 5 APIs in our dataset (Fig. 4c):

- SEARCH: Search using input parameters,
- REFINE\_SEARCH: Build on top of search carrying over existing parameters,
- GET\_INFO: Seek information about current or previously viewed memories,
- GET\_RELATED: Explore other memories similar to the current/prior memories, and,
- SHARE: Share it to friends or family,

As expected, SEARCH is the most dominant API call in the dataset. Note that the turns with GET and REFINE\_SEARCH API calls elevate the need for conversation in retrieving connected memories, where the user requests for memories similar to

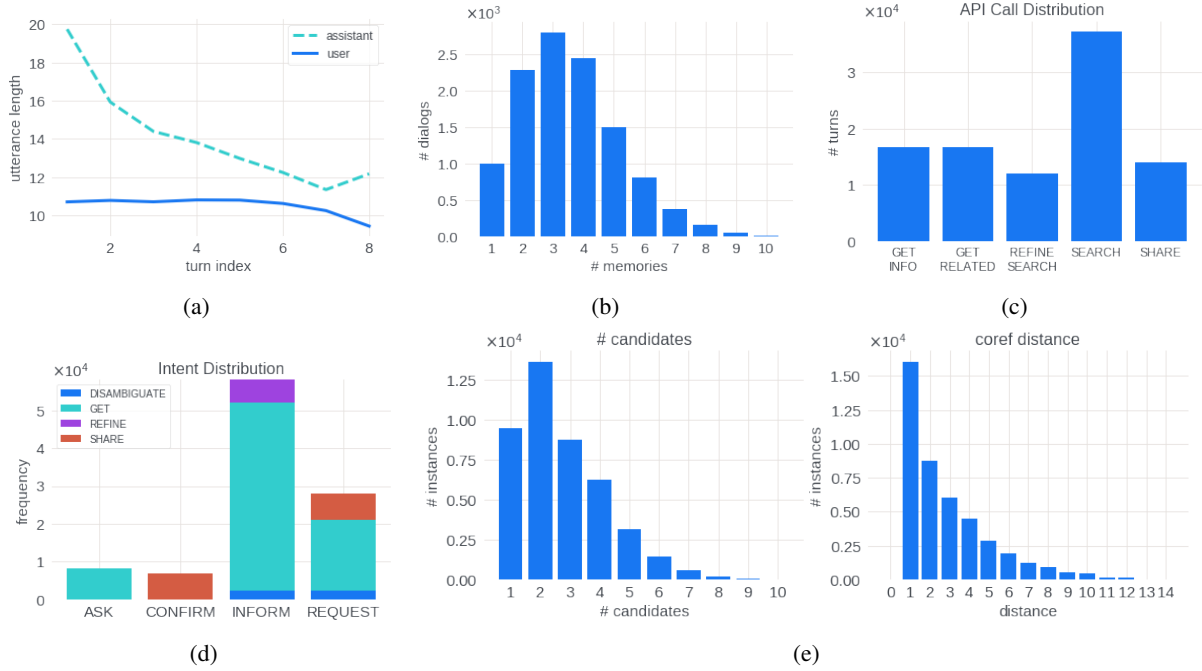


Figure 4: Distribution of (a) utterance lengths with dialog turns, (b) number of memory mentions in each dialog, (c) API calls across the dialogs, (d) dialog acts and activities, and (e) referent candidates (L) and coreference distance (R) between memory mentions.

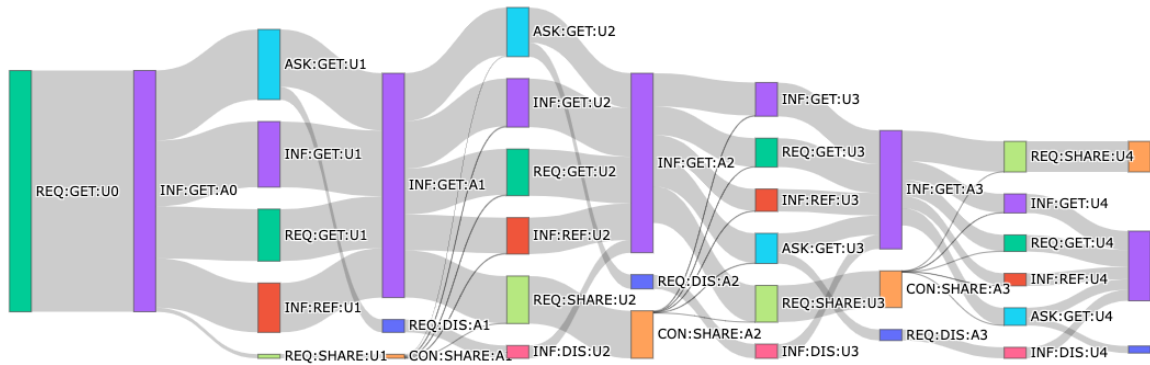


Figure 5: Transition of dialog acts in COMET for the first 4 turns, for dialog flows generated by our novel multimodal dialog simulator for connected memories. Each block is of the form ACT:ACTIVITY:[A|U][turn], to denote dialog act, activity, user or assistant turn, and turn number, respectively. See text for more details.

the ones already viewed or with additional specifications, respectively. Finally, Fig. 4e visualizes the distribution of number of candidates and utterance difference between the current and the one with referent memory (coreference distance). For turns requiring coreference resolution, the average number of candidates is 2.7 at a distance of 2.9 utterances. Though a majority of referents are naturally 1 utterance away (previous turn), the long tail (even up to 10+ utterances) indicates the presence of challenging multimodal coreferences.

**Analyzing Dialog Flows.** As mentioned earlier, the multimodal dialog simulator generates the dialog flows during the first phase of our data generation. We visualize these dialog flows in Fig. 5 for the first four dialog turns, where each block denotes

an intent at a particular turn and the grey stripes denote NLU intent transitions in subsequent turns. The width of the stripe is proportional to the frequency of the transition. For brevity, each block is label as ACT:ACTIVITY:[A|U][turn]. The high branch-off factors for these intents capture the diversity of the dialogs flows in our dataset, which is desirable in building a robust dialog system.

## 4 Task Formulation

To benchmark progress of conversational models towards the goal of assisting users in interactively querying connected memories in a meaningful way, we propose four main tasks for COMET. Tab. 2 outlines the task formulations along with the corresponding evaluation metrics.

Task Name	Goal	Evaluation
1. Assistant API Call Prediction	Given user utterances, predict the right API call necessary to execute the query.	Classification accuracy
2. Multimodal Coreference Resolution (MM-Coref)	Given user utterances, resolve referent memories to their canonical ID(s) as defined by the memory graph.	Coref Precision / Recall / F1
3. Multimodal Dialog State Tracking (MM-DST)	Given user utterances, track user belief states across multiple turns.	Slot Precision / Recall / F1
4. Assistant Response Generation	Given user utterances, ground-truth APIs and ground-truth object IDs, generate Assistant responses or retrieve from a candidate pool.	Generation: BLEU; Retrieval: Accuracy@k, mean reciprocal rank, mean rank

Table 2: Proposed tasks and descriptions on our COMET dataset. Please see Sec. 4 for more details.

#### 4.1 Assistant API Call Prediction

The first step in executing any query on connected memories successfully is to understand the user utterance in the context of the dialog history and multimodal information, and predict the right API call. For instance, a query like ‘When was the last time I was here?’ should result in a GET\_INFO API prediction. Note that errors in API call prediction cascade through the model pipeline resulting in an incorrect or unrelated response from the assistant. Thus, this task tests the ability of the conversational agent to predict the right API call. Evaluation is done per each turn through API call accuracy.

#### 4.2 Multimodal Coreference Resolution

Recall that one of our motivations to use conversations for querying connected memories is the ability to support multi-turn queries. In such scenarios, humans often use short-hands or references when the underlying referred entity (referent) can be usually deduced without any ambiguity. As an example, when looking at a particular memory, a follow-up ‘When was the last time I was here?’ is intuitive and natural, whereas ‘When was the last time I was at Waikiki Beach, Hawaii?’ requires the user to remember the name and use it in the query, making it cumbersome.

Therefore, the model must be able to handle multimodal coreferences in order to field such queries effectively. The input for this task includes the dialog history, multimodal context, and all the memories mentioned so far (as coreference candidates). The models needs to thus resolve the reference to one or more of the candidates. We use coreference precision, recall, and F1 to measure performance.

#### 4.3 Multimodal Dialog State Tracking

Due to the multimodal nature of COMET, we adopt multimodal dialog state tracking (MM-DST) used in (Kottur et al., 2021) as one of our tasks. To elab-

orate, slots in our dataset can be grounded in the multimodal context information and requires reasoning through the current or previously viewed memories. For instance, a query like ‘Where did we go from here?’ requires the slot value to be the currently viewing memory. This implies that the dialog states can contain non-textual tokens (e.g., memories), thus making it multimodal. In order to measure the performance in this task, we use slot recall, precision, and F1 scores. Note that unlike (Kottur et al., 2021), we drop evaluating for dialog act prediction since GET has an overwhelming majority due to the nature of the problem.

#### 4.4 Assistant Response Generation

This task evaluates the ability of the model to either generate a response or retrieve from a pool of candidates, given dialog history, ground-truth APIs & results, belief states, and multimodal contexts. Though the model has access to API results, producing a natural language utterance to describe it within the flow of the dialog is still a difficult task.

We evaluate this task in two different ways: (a) *Generative*, where the model produces the response similar to a conditional language model. We use n-gram overlap based BLEU-4 (Papineni et al., 2002) and more recent neural evaluation metric BERTScore (Zhang\* et al., 2020) to measure performance by comparing the generated response to the ground truth, and (b) *Retrieval*, where the model ranks a list of randomly pooled candidate responses (unique to a turn) along with the ground truth. Retrieval metrics like recall@k ( $k = \{1, 5, 10\}$ ), mean rank, and mean reciprocal rank are used.

### 5 Modeling & Empirical Analysis

We now perform preliminary empirical evaluation and analysis for the proposed tasks by training base-lines. Detailed modeling work is left as future work.



Model	1. API	2. Coref	3. DST	4. Gen.	
	Acc $\uparrow$	Coref F1 $\uparrow$	Slot F1 $\uparrow$	BLEU $\uparrow$	BERTS $\uparrow$
Text	<b>88.4</b> $\pm$ 0.4	76.5 $\pm$ 0.7	<b>90.0</b> $\pm$ 0.4	0.332 $\pm$ 0.003	0.884 $\pm$ 0.001
MM-BUTD	<b>89.0</b> $\pm$ 0.4	<b>84.4</b> $\pm$ 0.6	67.0 $\pm$ 0.6	<b>0.391</b> $\pm$ 0.004	<b>0.904</b> $\pm$ 0.001
MM-CLIP	82.9 $\pm$ 0.5	37.0 $\pm$ 0.8	65.9 $\pm$ 0.7	<b>0.385</b> $\pm$ 0.004	0.901 $\pm$ 0.001

Table 3: Baseline performances for GPT-2 models: text-only (text) and multimodal image features (MM). (1) **API Call Prediction (API)**, via classification accuracy, (2) **Multimodal Coreference Resolution (Coref)**, via coref prediction F1, (3) **Dialog State Tracking (DST)**, via slot F1, (4) **Response Generation** via BLEU and BERTScore.  $\uparrow$ : higher is better. **Bold** denotes best performance with statistical significance.

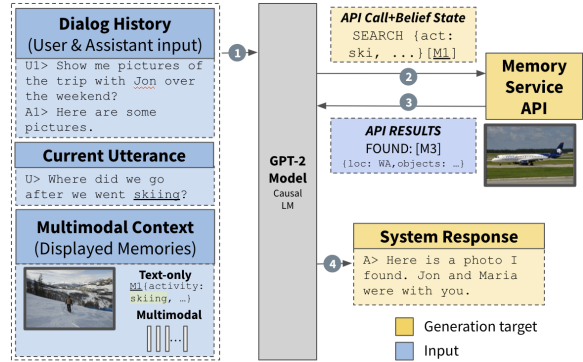


Figure 6: Baseline GPT-2 models for COMET. (1) Given the dialog history, multimodal context, and current user utterance, the model predicts the API call and belief state at the current turn, (2) The API call is executed and (3) the results are fed back into the model, (4) Finally, model produces a natural language response. As shown, GPT2-text uses attribute strings to represent memories, while GPT2-MM use image features.

**Dataset Split.** The dataset is randomly divided into: train (70%), val (15%), and test (15%). For our experiments, models are trained using train split and performance is reported on test, while val is used to pick the model hyper-parameters.

**Notations.** We follow the notation established in (Kottur et al., 2021), where each dialog of length  $N_r$  rounds is represented as  $\mathcal{D} = \{(U_i, A_i, M_i, B_i)\}_{i=1}^{N_r}$  with:

- $U_i$ : User utterance at turn  $i$
- $A_i$ : Assistant utterance at turn  $i$
- $M_i$ : Multimodal context, *i.e.*, memory graph and memories retrieved in the previous turns,
- $B_i$ : Multimodal belief state, a semantic parse of  $U_i$  (intent, slot, memory references).

Therefore, given the current user utterance ( $U_t$ ), dialog history  $H_t = (U_i, A_i)_{i=1}^{t-1}$ , and the multimodal context ( $M_t$ ), a COMET agent should predict the user belief state  $B_t$  and the natural language response  $A_t$  for every dialog turn  $t$ .

**Baselines.** Causal language models pretrained on large datasets have shown a lot of promise in multimodal and text-only task-oriented dialog modeling, when finetuned on the downstream task (Hosseini-Asl et al., 2020; Peng et al., 2020; Kottur et al., 2021; Moon et al., 2020). Following this popular approach, we adopt the transformer-based GPT-2 (Radford et al., 2019) model and jointly train it for API prediction, MM-Coref, DST, and response generation tasks, as shown in Fig. 6. In particular, we use the 12-layer GPT-2 (117M) model and fine-tune it on dialogs from COMET dataset, using early stopping based on token perplexity (<3 GPU hrs). We use two approaches to capture  $M_i$ :

(a) *text-only* (GPT2-text), where previously viewed memories and their attributes are represented as flattened strings. Note that this baseline uses ground-

truth activities from the memory graph.

(b) *multimodal* (GPT2-MM), where bottom-up and top-down (BUTD) (Anderson et al., 2018) and CLIP (Radford et al., 2021) image features are extracted for previous viewed memories, and fed as ‘visual tokens’ while finetuning the GPT-2 model.

**Analysis.** A key observation from Tab. 3 is that multimodal model GPT2-MM (BUTD) outperforms text-only in MM-Coref and response generation significantly. This is intuitive as multimodal coreference resolution requires understanding the memories beyond the obvious activity label in order to rightly resolve the reference. Consider the query: ‘When was the last time I played with my dog here?’. To resolve to the right memory, the system needs to understand which memory is about playing with the dog towards which a mere activity label *throwing frisbee* might be insufficient. For a similar reason, additional multimodal features improve response generation, especially to include chit-chat. On the other side, GPT-Text does better on capturing the dialog state suggesting complementary benefits.

**Conclusion.** We present a novel dataset for the dialogs for connected memories, COMET, with 11.5K user $\leftrightarrow$ assistant dialogs (103K utterances) grounded on the memory graphs. We present a novel multimodal dialog simulator, which generates simulated dialogs grounded on diverse memory graphs that are automatically configured. Our empirical analysis demonstrates many new challenges that our COMET dataset brings, highlighting new directions of research in this area.



590  
591  
592  
593  
594  
  
595  
596  
597  
598  
  
599  
600  
601  
602  
  
603  
604  
605  
606  
607  
608  
609  
  
610  
611  
612  
  
613  
614  
615  
616  
617  
618  
  
619  
620  
621  
622  
623  
624  
  
625  
626  
627  
628  
629  
  
630  
631  
632  
  
633  
634  
635  
636  
637  
  
638  
639  
640  
641  
642  
643

## 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*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual question answering. In *ICCV*.

Andrei Barbu, Siddharth Narayanaswamy, and Jeffrey Mark Siskind. 2013. [Saying what you're looking for: Linguistics meets video search](#). *CoRR*, abs/1309.5174.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *CVPR*.

Mihail Eric, Rahul Goel, Shachi Paul, Adarsh Kumar, Abhishek Sethi, Peter Ku, Anuj Kumar Goyal, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tur. 2019. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Mauajama Firdaus, Nidhi Thakur, and Asif Ekbal. 2020. Multidm-gen: Aspect-guided response generation in multi-domain multi-modal dialogue system using graph convolutional network. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2318–2328.

Shuyang Gao, Sanchit Agarwal, Abhishek Seth and, Tagyoung Chun, and Dilek Hakkani-Ture. 2019. Dialog state tracking: A neural reading comprehension approach. In *Special Interest Group on Discourse and Dialogue (SIGDIAL)*.

Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogerio Feris. 2018a. Dialog-based interactive image retrieval. In *NeurIPS*.

Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogerio Feris. 2018b. [Dialog-based interactive image retrieval](#). In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Neibbles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970.

Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, pages 263–272.

Chiori Hori, Anoop Cherian, Tim K. Marks, and Florian Metze. 2018. Audio visual scene-aware dialog track in dstc8. *DSTC Track Proposal*.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *arXiv preprint arXiv:2005.00796*.

Lu Jiang, Junwei Liang, Liangliang Cao, Yannis Kalantidis, Sachin Farfade, and Alexander Hauptmann. 2018. Memexqa: Visual memex question answering. *arxiv*.

Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021. [SIMMC 2.0: A task-oriented dialog dataset for immersive multimodal conversations](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4903–4912, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Satwik Kottur, José MF Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. 2019. Clevr-dialog: A diagnostic dataset for multi-round reasoning in visual dialog. *arXiv preprint arXiv:1903.03166*.

Shuang Li, Tong Xiao, Hongsheng Li, Bolei Zhou, Dayu Yue, and Xiaogang Wang. 2017. Person search with natural language description. *arXiv preprint arXiv:1702.05729*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Seungwhan Moon, Satwik Kottur, Paul A Crook, Ankita De, Shivani Poddar, Theodore Levin, David Whitney, Daniel Difranco, Ahmad Beirami, Eunjoon Cho, Rajen Subba, and Alborz Geramifard. 2020. Situated and interactive multimodal conversations. *arXiv preprint arXiv:2006.01460*.

Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. Memory graph networks for explainable memory-grounded question answering. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 728–736.

Nina Pantic. 2021. [\[link\]](#).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia,

699	Pennsylvania, USA. Association for Computational Linguistics.	
700		
701	Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayan-	
702	deh, Lars Liden, and Jianfeng Gao. 2020. Soloist:	
703	Few-shot task-oriented dialog with a single pre-	
704	trained auto-regressive model. <i>arXiv preprint</i>	
705	<i>arXiv:2005.05298</i> .	
706	Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya	
707	Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sas-	
708	try, Amanda Askell, Pamela Mishkin, Jack Clark,	
709	et al. 2021. Learning transferable visual models	
710	from natural language supervision. In <i>International</i>	
711	<i>Conference on Machine Learning</i> , pages 8748–8763.	
712	PMLR.	
713	Alec Radford, Jeffrey Wu, Rewon Child, David Luan,	
714	Dario Amodei, and Ilya Sutskever. 2019. Language	
715	models are unsupervised multitask learners. <i>OpenAI</i>	
716	<i>blog</i> , 1(8):9.	
717	Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara,	
718	Raghav Gupta, and Pranav Khaitan. 2019. Towards	
719	scalable multi-domain conversational agents: The	
720	schema-guided dialogue dataset. In <i>Association for</i>	
721	<i>the Advancement of Artificial Intelligence (AAAI)</i> .	
722	Nils Reimers and Iryna Gurevych. 2019. <a href="#">Sentence-bert:</a>	
723	<a href="#">Sentence embeddings using siamese bert-networks.</a>	
724	In <i>Proceedings of the 2019 Conference on Empirical</i>	
725	<i>Methods in Natural Language Processing</i> . Associa-	
726	tion for Computational Linguistics.	
727	Amrita Saha, Mitesh M Khapra, and Karthik Sankara-	
728	narayanan. 2018. Towards building large scale multi-	
729	modal domain-aware conversation systems. In <i>AAAI</i> .	
730	Jost Schatzmann, Blaise Thomson, Karl Weilhammer,	
731	Hui Ye, and Steve Young. 2007. Agenda-based user	
732	simulation for bootstrapping a pomdp dialogue sys-	
733	tem. In <i>Human Language Technologies 2007: The</i>	
734	<i>Conference of the North American Chapter of the As-</i>	
735	<i>sociation for Computational Linguistics; Companion</i>	
736	<i>Volume, Short Papers</i> , pages 149–152.	
737	Fuwen Tan, Paola Cascante-Bonilla, Xiaoxiao Guo, Hui	
738	Wu, Song Feng, and Vicente Ordonez. 2019. Drill-	
739	down: Interactive retrieval of complex scenes using	
740	natural language queries. In <i>Neural Information Pro-</i>	
741	<i>cessing Systems (NeurIPS)</i> .	
742	Stefanie Tellex and Deb Roy. 2009. <a href="#">Towards surveil-</a>	
743	<a href="#">lance video search by natural language query.</a> In	
744	<i>Proceedings of the ACM International Conference</i>	
745	<i>on Image and Video Retrieval, CIVR '09</i> , New York,	
746	NY, USA. Association for Computing Machinery.	
747	Jesse Thomason, Michael Murray, Maya Cakmak, and	
748	Luke Zettlemoyer. 2019. Vision-and-dialog naviga-	
749	tion. <i>arXiv preprint arXiv:1907.04957</i> .	
750	Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li,	
751	Li Fei-Fei, and James Hays. 2019. Composing text	
752	and image for image retrieval - an empirical odyssey.	
753	In <i>Proceedings of the IEEE/CVF Conference on Com-</i>	
754	<i>puter Vision and Pattern Recognition (CVPR)</i> .	
	Harm de Vries, Kurt Shuster, Dhruv Batra, Devi Parikh,	755
	Jason Weston, and Douwe Kiela. 2018. Talk the	756
	walk: Navigating new york city through grounded	757
	dialogue. <i>arXiv preprint arXiv:1807.03367</i> .	758
	Harm de Vries, Florian Strub, Sarath Chandar, Olivier	759
	Pietquin, Hugo Larochelle, and Aaron Courville.	760
	2017. Guesswhat?! visual object discovery through	761
	multi-modal dialogue. In <i>CVPR</i> .	762
	Hui Wu, Yupeng Gao, Xiaoxiao Guo, Ziad Al-Halah,	763
	Steven Rennie, Kristen Grauman, and Rogerio Feris.	764
	2021. Fashion iq: A new dataset towards retrieving	765
	images by natural language feedback. In <i>Proceed-</i>	766
	<i>ings of the IEEE/CVF Conference on Computer Vi-</i>	767
	<i>sion and Pattern Recognition (CVPR)</i> , pages 11307–	768
	11317.	769
	Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q.	770
	Weinberger, and Yoav Artzi. 2020. <a href="#">Bertscore: Eval-</a>	771
	<a href="#">uating text generation with bert.</a> In <i>International</i>	772
	<i>Conference on Learning Representations</i> .	773
		774

## 6 Supplementary Materials

### 6.1 Limitations of the proposed work

The generalizability of COMET is naturally bounded by the underlying graph simulator, especially around memory attribute labels of place, people, and time. However, we justify this as follows: (a) Recall that the focus of our work is to enable an assistant that can understand and execute user queries about connected memories through an interactive dialog. Even with the simulated dialog flows, COMET captures several interesting challenges related to multimodal dialog, for instance, coreference resolution and dialog state tracking (as seen in Sec. 3.3 and Sec. 5). This opens the door to new research directions in multimodal conversation, especially in the absence of a readily available large-scale personal photo collection dataset (along with attributes and metadata). (b) Due to the two-stage data collection pipeline, COMET is amenable to data augmentation techniques that can increase the robustness of the downstream dialog model. For instance, the dataset can be easily augmented by replacing named entities in the memory graph and utterances, without changing the flow.

### 6.2 Ethical Considerations

All identifiable faces from the COCO images were blurred using a CV algorithm, mitigating potential privacy risks. The dataset, when released publicly, will include those edited images.

Annotators for our task were employed as full-time and contracted via a leading NLP / linguistics annotation platform. Annotators were given clear instructions and disclaimers detailing the escalation path ("Report Dialog") for an (unlikely) case where the data may include sensitive topics or images (shown in Figure 9).

### 6.3 Dataset Example

Figure 7 illustrates an example dialog from COMET, along with the set of images associated with each turn (U: User, A: Assistant). API Annotations are formatted as follows: `INTENT [slot = value, ...]` (`request_slot`) `<memory: ID>`. When there is no new image introduced for a given turn, it is assumed that the images from previous turns (if any) are left visible to the user, therefore continuing to serve as the grounding multimodal context.

### 6.4 Annotation UI

Figure 8 illustrates the annotation UI used to collect multimodal paraphrases of the dialog. Annotators are shown the pre-generated dialog flows (templated utterances), along with the text boxes where the paraphrases can be entered. The top portion of the UI displays the images (assumed to be) shown to the user for each given turn, which gets dynamically updated as annotators click on new text boxes for entering paraphrases. A shortened list of meta data associated with each image is also shown for reference.





	Utterances	API Annotations / Dialog Acts	Display Images
U	I'm looking for photos from 2020 with my friends.	SEARCH [time = 2020]	
A	Here's a photo of your friends skiing.	INFORM:GET <memory: 17>	
U	Is there a similar photo except with Logan and Linda?	REFINE_SEARCH [participant = Logan, Linda] <memory: 17>	
A	Here's a similar memory of your friends at Mount Baldy the afternoon of October 25, 2020, and another of them surfing.	INFORM:GET <memory: 23, 47>	
U	When was the surfing one taken?	GET_INFO (time) <memory: 47>	
A	Which one are you talking about?	REQUEST:DISAMBIGUATE	
U	The second picture you showed me, the one of them surfing.	INFORM:DISAMBIGUATE <memory: 47>	
A	This happened at 2:12 PM on October 26, 2020.	CONFIRM:GET_INFO [time = 10/26/20 2:12PM] <memory: 47>	
U	Anything with a similar group to this picture except in 2019?	GET_RELATED [time = 2019] <memory: 47>	
A	Here is a similar picture of your friends. It looks like they are having a great time in both.	INFORM:GET <memory: 24>	
U	When and where did this happen?	GET_INFO (time, location) <memory: 24>	
A	This occurred at 12:50 PM on December 30, 2019 in Bay Area, California, USA.	CONFIRM:GET_INFO [time = 12/30/19 12:50PM] <memory: 24>	

Figure 7: **Dataset Example.** Dialog labels include intent, slots, and multimodal coreferences.



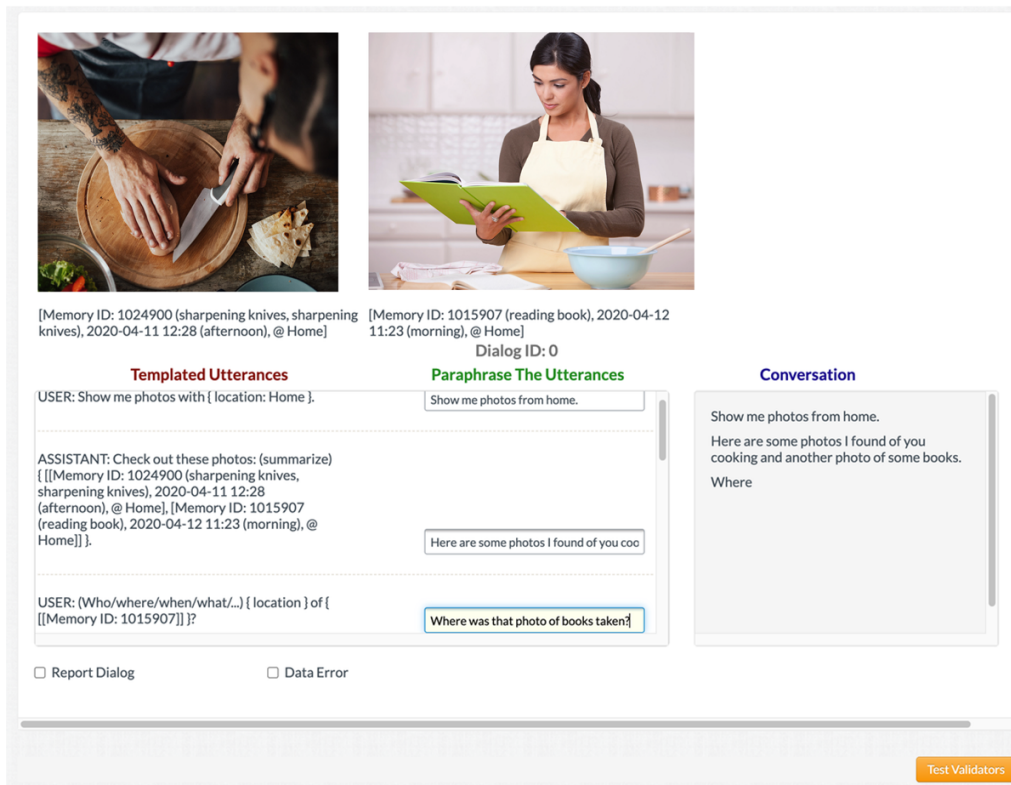


Figure 8: **The annotation tool UI.** Annotators are shown the templated utterances, and a set of photos that dynamically get updated for each turn, based on the pre-generated dialog flows.

## Sensitive topics

Any reference to the following topics is inappropriate and should be flagged by checking the "Report Dialog" button:

- PII
  - First Name & Last Name (just one name is not PII)
    - if the first and last name seem to be used in the a slot that would indicate a public figure, such as musical artist, fictional character, or political figure, please **do not mark as containing PII**. If you are unsure check if the name has a wikipedia page. If so, **do not mark as containing PII**.
  - phones numbers, credit card numbers, or social security numbers
  - email
  - Addresses are NOT considered PII unless they are accompanied but another piece of PII (i.e name), in which the combined information would allow you to identify the user.
- Offensive, racist, biased and non-tolerant behavior
  - Profanity, slurs, language that is offensive to any cultural, racial, and religious groups.
  - Bias towards or unequal treatment to any cultural, racial, and religious groups.
  - Anything inconsistent with the values of tolerance and respect for diversity.
- Violence and self-harm
  - Any content which facilitates or encourages violent behavior towards others or any form of self-harm.
  - Descriptions or depictions of violent behavior or self-harm.
  - Any reference to threats or weapons.
  - Any reference to human trafficking, child endangerment or exploitation, or animal abuse.
  - Violent or non-violent crime of any kind
- Sexual or flirtatious behavior
  - Any reference to sexual behavior or materials, legal or illegal.
  - Sexual or flirtatious expressions or innuendo.
  - Explicit or sexual language or physical descriptions.
- Controversial and Polarizing Topics
  - Political opinions or politically charged people or events. General political enquiries are okay, (e.g. *show me political news; Is there any coverage of the election?*)
  - Religion
  - Disputed regions or events
  - Sexuality
  - Cultural practices

Figure 9: Disclaimers shown to the annotators, detailing the escalation path.