A Textual Dataset for Situated Proactive Response Selection

Naoki Otani[α*], **Jun Araki**[β], **HyeongSik Kim**[β], **Eduard Hovy**[γ]
[α]Megagon Labs, Mountain View, CA, USA [β]Robert Bosch LLC, Sunnyvale, CA, USA
[γ]University of Melbourne, Melbourne, VIC, Australia
[α]notani@alumni.cmu.edu [β]{jun.araki,hyeongsik.kim}@us.bosch.com
[γ]eduard.hovy@unimelb.edu.au

Abstract

Recent data-driven conversational models are able to return fluent, consistent, and informative responses to many kinds of requests and utterances in task-oriented scenarios. However, these responses are typically limited to just the immediate local topic instead of being wider-ranging and proactively taking the conversation further, for example making suggestions to help customers achieve their goals. This inadequacy reflects a lack of understanding of the interlocutor's situation and implicit goal. To address the problem, we introduce a task of proactive response selection based on situational information. We present a manually-curated dataset of 1.7k English conversation examples that include situational background information plus for each conversation a set of responses, only some of which are acceptable in the situation. A responsive and informed conversation system should select the appropriate responses and avoid inappropriate ones; doing so demonstrates the ability to adequately understand the initiating request and situation. Our benchmark experiments show that this is not an easy task even for strong neural models, offering opportunities for future research.

1 Introduction

Conversational assistant systems have recently shown significant improvements for understanding users' inquiries along with background knowledge, conducting requested operations, and returning natural language responses. Yet, typical systems are likely to be *passive* and only process user-initiated requests or merely ask values for domain-specific slots (Williams et al., 2013; Ammari et al., 2019). In contrast, human assistants like hotel concierges are more *proactive*, acting to address unmentioned needs and expected future events (Cho et al., 1996; Bellini and Convert, 2016). They do not only make a direct response or a clarification question to their interlocutors but also provide personalized information/assistance based on context and knowledge.

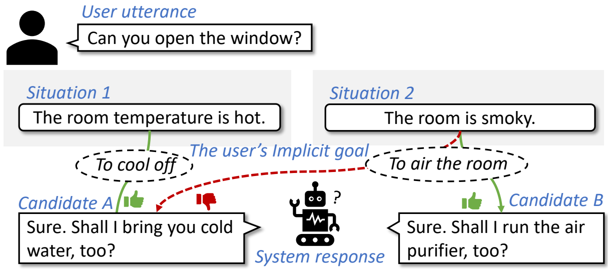


Figure 1: An example of situated goal-aware proactive response selection. The response candidate A is appropriate in Situation 1 but not in Situation 2.

To push the frontier of task-oriented conversation technologies, we propose a task of *proactive* response selection for single-turn help-seeking conversations in English. We mean by proactive that a system engages in an interaction in a cooperative manner (Grice, 1975) and suggests something helpful to a user. The proposed task touches upon two crucial aspects of help-seeking conversations: situation-awareness and goal-awareness.

Situation: Situational information plays an important role in conversations as we illustrate in Figure 1. The example shows a user utterance "Can you open the window for me?" (top) and two response candidates (bottom), "Sure. Shall I bring you cold water, too?" (left) and "Sure. Shall I run the air purifier, too?" (right). Although both candidates here sound helpful, their appropriateness varies depending on context: When the room is hot, suggesting a cold drink is appropriate assistance (left), but on the other hand, if the room is smoky, then running an air purifier is more helpful (right). Likewise, different situations make different responses more appropriate. A fair amount of situational information can be perceived as visual image, sound, and other kinds of sensory signals, and some of those are effectively incorporated into

*This work was done while the first author was at Carnegie Mellon University.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 3856–3874
July 9-14, 2023 ©2023 Association for Computational Linguistics

multi-modal conversational systems (Crook et al., 2019; Kottur et al., 2019). Yet, there are many other types of information that modern conversation assistance systems have access to, for example, via external APIs such as calendars and maps. In this study, we represent situational statements of six semantic categories (location, possession, etc.) in free English texts, which are more explicit as a semantic representation than just maintaining conversation histories (Lowe et al., 2015; Li et al., 2017; Henderson et al., 2019) and more flexible than structured representations of limited vocabulary (Williams et al., 2013; Budzianowski et al., 2018).

Goal: In the aforementioned example, the two actions address two different goals associated with opening a window, namely, *to cool off* and *to air the room*. While often being unspoken, underlying goals provide important semantic connections among context and utterances on many occasions (Allen and Perrault, 1980) particularly when language is indirect (Perrault, 1980; Walker et al., 2011; Stevens et al., 2015). We use goal information as a stimulus for soliciting naturalistic and proactive responses from human annotators in data collection.

We introduce a new dataset of **S**it**U**atated, **G**oal-**A**ware, and proactive **R**esponses (SUGAR; §3), which contains 1,760 examples of single-turn English conversations.[1] Each conversation includes a user request anchored by an implicit goal, a reference response, and 12 sentences of situational information. As a proof of concept, we perform the task of *situated* response selection on SUGAR by adding two extra response candidates to each example. All responses are annotated with three-point appropriateness ratings.

To create SUGAR, we extracted user utterances and goals from common-sense knowledge bases, ATOMIC (Sap et al., 2019) and ConceptNet (Speer et al., 2017), and collected proactive responses with supporting situational information by crowdsourcing. We then used a language generation model, COMET (Bosselut et al., 2019; Hwang et al., 2021), to generate additional situational statements. Finally, we selected two more response options for each reference response using an adversarial method to form examples of three-choice response selection. To ensure data quality, we performed multiple manual validation steps during data collection. In our experiments on SUGAR (§4), Transformer-based rankers achieved over 80% precision@1 when when only the relevant situational statements were presented. However, precision decreased when distractors were included in the input, and this trend further continued as more distractors were added in our controlled experiments. These results suggest potential opportunities for future research.

[1] https://github.com/notani/sugar-conversational-dataset

2 Related Work

2.1 Conversational Dataset

Acquisition of real or realistic conversational data has been an essential step for developing conversation engines that imitate human communication (Serban et al., 2018). Various datasets have been constructed with a focus on different aspects of communication.

With regard to target communicative aspects, the most relevant to our work is SIMMC (Moon et al., 2020). SIMMC encompasses surrounding situational information that gives a basis for verbal interactions in task-oriented scenarios in the shopping domain. Moon et al. collected visually-grounded conversation examples from pairs of human annotators interacting with each other in a virtual environment (Crook et al., 2019), where one annotator seeks help for shopping, and the other provides assistance. SUGAR is also concerned with how human interlocutors perform situated conversations in a help-seeking setting. Our work extends this direction to scenarios other than shopping and includes more diverse types of information that modern conversational assistants could access via sensors or external APIs (e.g., temperature and schedule) by representing situational information in a textual form as opposed to visual images.

The choice of modality is motivated by existing conversational datasets that express various kinds of background information in plain text: the persona of an interlocutor (Zhang et al., 2018; Dinan et al., 2020), emotional states (Rashkin et al., 2019; Ghosal et al., 2022), and related documents (Zhou et al., 2018; Dinan et al., 2019). These examples demonstrate the utility of textual forms for representing both explicit and implicit information of various kinds.

Some existing datasets are concerned with information-seeking conversations like restaurant recommendation where suggestions by assistants

Category	Definition	Example
Location	Information about [user]'s current location.	[user] is home. / [user] is at the entrance of a house.
Possession	Information about what [user] possesses.	[user] owns a car. / There are apples in the kitchen.
Time	Information about time.	It's midnight. / It's morning.
Date	Information about date and season.	It's [user]'s birthday. / It's summer.
Behavior	Information about [user]'s behavior.	[user] just woke up. / [user] came back from jogging.
Environment	Information about non-user entities (person, objects, etc.).	The room is hot. / [user]'s car has a flat tire.

Table 1: Definitions of the situation categories. [user] denotes the user of a conversation system.

naturally occur (e.g., "If you like French cuisine, how about RestaurantX?", "I can find transportation for you."). However, it is not trivial to solicit such naturalistic proactive utterances in more diverse help-seeking scenarios. In many cases, the minimum objective of a conversation can be achieved by responding to user-initiated inquiries, and such kinds of responses are relatively easy to collect from non-expert annotators (Budzianowski et al., 2018; Byrne et al., 2019; Eric et al., 2020). We address this problem by leveraging implicit goals behind user requests. The comprehension of goals in conversations has been recognized to be important not only in task-oriented dialog research but also in a broad range of research areas such as linguistics, psychology, and artificial intelligence. (Schank and Abelson, 1977; Clark and Schaefer, 1989; Gordon and Hobbs, 2004; Rahimtoroghi et al., 2017). Human interactions often involve indirect speech acts (Perrault, 1980; Gibbs and Bryant, 2008) and indirect responses like non-yes/no answers to polar questions (Hockey et al., 1997; de Marneffe et al., 2009; Stevens et al., 2015; Louis et al., 2020). These studies motivate our strategy for soliciting natural-sounding proactive responses from crowd workers.

In contrast to most datasets we introduced here, SUGAR only contains single-turn conversation examples due to the ease of data collection and quality control. Our primary focus is on conversational assistance, where short-turn conversations are common (Völkel et al., 2021). Thus, we believe that single-turn examples are still useful for system development. It is possible to extend our problem setting and data collection approach to a multi-tern setting, which we leave as future work.

2.2 Response Selection

Automatic response models can be divided into two approaches: response generation and response selection. Response generation directly generates natural language response text from scratch, and response selection selects a response from a candidate pool built by humans, templates, or language generation systems. The latter approach is widely used in many real-world applications cases because of the controllability of responses and the easiness of evaluation (Deriu et al., 2020). In this study, we focus on the task of response selection as a proof of concept. We assume that an external response generation system generates candidates based on the system's functionality and focus on picking the appropriate ones. SUGAR can also serve as a valuable resource for the development and evaluation of response generation systems, which is an interesting avenue for future research.

To train and evaluate a response selection system, each example must have distractors (negative responses), but typically, conversational datasets only contain ground truth responses. Thus, it has been commonly practiced to pick negative responses by random sampling (Lowe et al., 2015; Henderson et al., 2019). This approach comes in handy but may introduce negative responses that are clearly off-topic or false negatives (Akama et al., 2020; Hedayatnia et al., 2022). To alleviate this problem, we use an adversarial filtering algorithm (Zellers et al., 2018; Sakaguchi et al., 2019; Bhagavatula et al., 2020) to select competitive distractors and recruit crowd workers to rate candidates, allowing each example to have multiple acceptable responses.

3 Task and Data

The goal of this study is to provide a resource for developing a system that can observe situational information and return a proactive response to a user. We consider six categories of observable *situational statements* (Table 1): location (where the user is), possession (what the user has), time, date, behavior (what the user is/was doing), and envi-

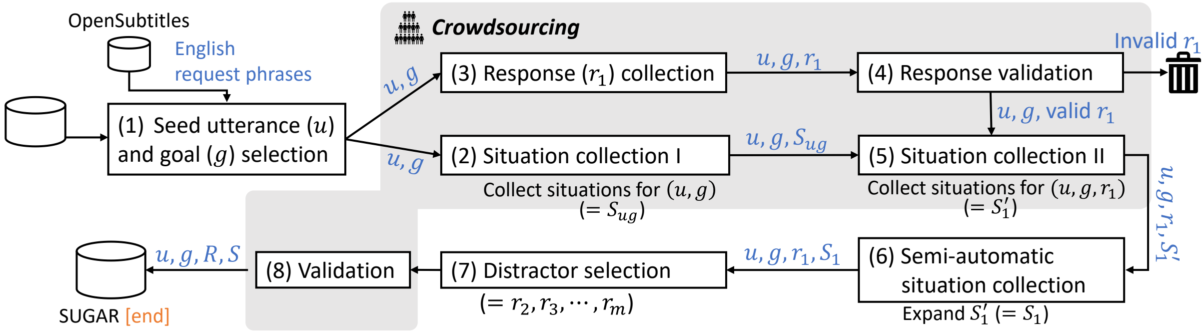


Figure 2: Pipeline for data collection. We start with existing common-sense knowledge bases (ATOMIC and ConceptNet) and extract utterance and goal events as a seed (1). We collect responses and situational statements for each seed by crowdsourcing (2-5), acquire more situational statements semi-automatically (6), and select distractor responses and situations to form response selection examples (7). We finally validate the examples manually (8). Steps (2) and (3-4) are executed in parallel.

	u	r	g	s
Unique sentences	380	1,738	431	4,450
Tokens	14,458	28,694	7,499	147,710
Avg. tokens/ex	8.2	16.3	4.3	83.9

Table 2: Dataset statistics. The dataset contains 1,760 examples (33,794 sentences).

ronment (temperature, etc.) We define a *proactive* response to be a response that provides *suggestions* to help users achieve their goals.

3.1 Problem Formulation

Our task has five components: (1) a user utterance u, (2) situational statements $S = \{s_i\}_{i=1,\cdots,l}$, where l is the number of statements, (3) responses $R = \{r_i\}_{i=1,\cdots,m}$, where m is the number of response candidates[2], (4) their appropriateness ratings $Y = \{y_i\}_{i=1,\cdots,m}$, where y_i is a three-point Likert scale, and (5) an implicit goal g. S can include distractors that are not directly relevant to the conversation. u, S, and R are given as input, and the task is to re-rank R. Response selection systems are trained and evaluated by Y. In this study, we set $l = 12$ and $m = 3$.

3.2 Data

SUGAR contains 1,760 high-quality examples, each of which has three response candidates and 12 sentences of situational information (situational statements). Table 2 shows the dataset statistics.

[2]We pick $m - 1$ responses automatically such that they are less appropriate than the reference response in a given context (See Step 7). Nevertheless, there usually exist one or more acceptable responses to a given user utterance. We thus annotate all acceptable responses manually (Step 8).

We constructed the dataset with the eight steps shown in Figure 2. We describe them below.[3]

(1) Seed Utterance & Goal Selection: We harvested action and goal events from two common-sense knowledge bases, ATOMIC (Sap et al., 2019) and ConceptNet (Speer et al., 2017), where knowledge is represented as nodes representing events or concepts and edges connecting them with semantic relations. The collected action-goal node pairs served as the seed utterance-goal for soliciting responses and situational statements in the following data collection steps. First, we extracted nodes consisting of verb phrases (VPs) that appear at least five times within English request phrases (e.g., Please `VP`, Could you `VP`?, etc.) in the OpenSubtitles corpus (Henderson et al., 2019). These request expressions were also used as the surface form of u. Two of the authors then selected 563 events that can be achieved within a reasonable time span, can be assisted by someone else, and can be triggered by a goal. We retrieved their implicit goals g by goal-related edges in ATOMIC and ConceptNet. Specifically, we used `xNeed` in the reverse direction and `xIntent` in ATOMIC and `HasPrerequisite` in the reverse direction and `MotivatedByGoal` in ConceptNet. Finally, two of the authors evaluated the node pairs and picked 501 (u, g) pairs for which we can naturally say "I do u to achieve g." (e.g., *open a window* to *cool off.*) We also merged synonymous expressions (e.g., *go to a market* and *go to a supermarket*) into a single entry and corrected grammatical errors and unnatural phrases.

[3]See also Appendix A for technical details.

(2) Situation Collection I: We collected situational statements in two phases to simplify annotation work. The first phase focuses on u and g, and the second phase considers r in addition to u and g. In this step, we presented a pair of u and g texts to crowd workers and instructed them to specify situational information that is required to guess the goal based on the utterance. For example, an implicit goal "to cool off" can be naturally inferred by situations like "The user is home. The room temperature is hot." We asked workers to write *observable* facts in the six semantic categories (Table 1). For example, "The room temperature is hot." is valid, but "The user feels hot." is invalid as assistance systems cannot *observe* the user's feeling. We recruited one worker for each (u, g) pair and paid \$0.12 per HIT[4] (one (u, g) pair/HIT).

(3) Response Collection: In parallel to Step (2), we recruited two crowd workers for each (u, g) pair to collect responses. The workers created at least two responses: one of the responses accepts and the other rejects the request. We asked the workers to write a *proactive* response, a response providing suggestions for goal fulfilment.[5] To solicit responses closely connected to implicit goals rather than to domain knowledge, we instructed the workers to avoid posing a clarification question like "Sure, I'll turn on the air conditioner for you. *Would you like it on a high or low setting?* (= clarification)" The workers were presented one u-g pair in each HIT and were paid \$0.30/HIT.

(4) Response Validation: We present the utterances, goals, and collected responses to crowd workers and evaluated the helpfulness of the response. A response is considered to be valid if it satisfies the following criteria: (1) the response suggests or requests something new, and (2) the suggestion or request is helpful for achieving the goal. Each response was evaluated by three workers. We then picked the responses that were approved by two or three workers. We call a verified response *a reference response* r_1 hereafter. Each HIT contains up to seven responses, and one of them is a dummy question for evaluating crowd workers. For quality control, we filtered out crowd workers who participated in the task twice or more and did not reach 0.75% accuracy for the dummy questions. The workers were paid \$0.18 for this task. Krippendorff's α was 0.547.

[4]Human Intelligence Task, a unit of task in MTurk.

[5]For a response that rejects a user's request, we instructed the workers to provide a reason for rejection (*e.g.,* we cannot brew coffee *because we are out of coffee filters*) in addition to a suggestion.

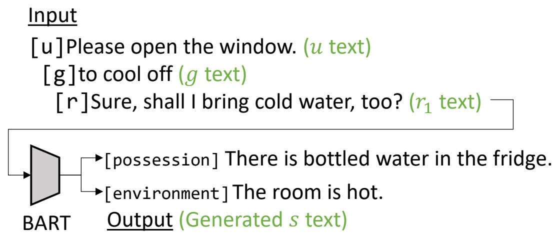


Figure 3: Example of automatic situation generation by BART (Step 6). `[u]`, `[g]`, and `[r]` are special symbols to denote the types of the following texts. The first output token is given as a prompt to control the semantic category of output.

Loc.	Poss.	Time	Date	Behav.	Env.
1990	3546	1083	152	1699	2793

Table 3: Number of situational statements ($\in S_1$).

(5) Situation Collection II: We collected situational statements from crowd workers with the following two goals: (1) to collect situational statements that cover the reference response r_1 and (2) to verify the situational statements collected in Step (2). We presented (u, g, r_1) with the statements obtained in Step (2) and again instructed crowd workers to write observable facts. The results of Step (2) were provided as editable initial values, and we encouraged workers to update the texts when it is necessary. We recruited one crowd worker for each (u, g, r_1) with the reward of \$0.42/HIT.

(6) Semi-automatic Situation Collection: We found that the collected situational statements were often under- or over-specified. We addressed this by automatic situation generation and manual verification.

The first author examined all the situational statements, discarded/modified inappropriate situations, and categorized them into six categories. We then used the cleaned and labeled texts to fine-tune a neural sequence-to-sequence to generate more situations. Specifically, we fine-tuned BART (Lewis et al., 2020) trained on ATOMIC$^{20}_{20}$ (Hwang et al., 2021)[6] to take a concatenation of u, g, and r_1 as input and generate a text for a given situation cat-

[6]Note that the framework of pre-training Transformer models on common-sense knowledge bases was originally proposed by Bosselut et al. (2019).

egory as illustrated in Figure 3. We performed a beam search of width 3 and took top-3 generation results for each input and relation. Finally, we manually verified the generated situations, resulting in 4,375 unique situations (6.4 ± 1.3 statements per example). We denote the situational statements attached to (u, g, r_1) by S_1. Table 3 shows the distribution of situation categories in SUGAR. Statements about possession and environment appear most frequently, which is reasonable because such situational information often decides actions that can be carried out (e.g., to drink coffee, coffee must be available). The other categories are less frequent, but 64% of examples have at least one time or date information, and 69% have a statement about behavior.

(7) Distractor Selection: The examples collected in the previous steps only contain reference responses r_1 and supporting situational statements S_1. We added $m - 1$ response candidates along with their relevant situational information as distractors so that all examples have m response candidates and l situational statements. We set $m = 3$ and $l = 12$. In this section, we describe the high-level idea of our algorithm. Appendix B presents technical details.

Distractors can be obtained by random sampling as practiced in many studies (Henderson et al., 2019) or by advanced methods such as adversarial filtering (Li et al., 2019; Gupta et al., 2021). However, such approaches may introduce off-topic responses that are easy to rule out and false negatives — acceptable responses treated as negative examples, degrading system performance as well as reliability of evaluation (Akama et al., 2020; Hedayatnia et al., 2022).

To alleviate this problem, we combine lexical matching and adversarial filtering (Zellers et al., 2018; Sakaguchi et al., 2019; Bhagavatula et al., 2020) to construct distractors and validate them manually (see Step 8). We first created an initial dataset by a lightweight method based on sentence embeddings and lexical matching. We then performed $J = 3$ rounds of adversarial filtering. In each round, we split the dataset into $K = 10$ folds, and for each split, we trained a binary logistic regression classifier that takes sentence embeddings of u, S_1, and a response candidate. We computed sentence embeddings by SentenceTransformers (Reimers and Gurevych, 2019) with MPNet (Song et al., 2020). We used the trained classifier to identify easy distractors and replace them with more confusing ones with respect to the score function. We sampled two responses for each example. All response candidates in the same example have the same polarity. Finally, we expanded S_1, which only contains relevant information to u and r_1, to obtain a set of $l = 12$ situations S such that some of them are related to distractors but do not disqualify r_1, and statements do not contradict with each other. We again used sentence embeddings to find topically related situational information and avoid contradiction with keyword-based heuristics.

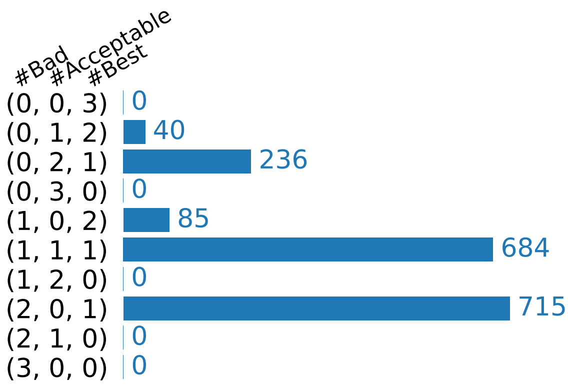


Figure 4: Result of rating annotations (Step 8). The labels denote (the number of *Bad* options, the number of *Acceptable* options, the number of *Best* options). We removed one example with three *Acceptable* responses from the dataset.

(8) Validation: There are usually multiple appropriate responses in one conversational context, and therefore, some of the challenging "distractors" picked in the previous step can be acceptable or even more appropriate than the reference r_1. To avoid introducing false negatives, we rated all response candidates on a three-point Likert scale (*Bad, Acceptable,* or *Best*) by crowdsourcing. We recruited three crowd workers per example with the reward of \$0.25/each and asked them to pick an appropriate response candidate (Krippendorff's α (Krippendorff, 2006) of 0.484). We then aggregated ratings by the statistical model proposed by Zhou et al. (2014) to obtain the final rating Y.[7] We discarded one example in this validation step and obtained 1760 examples with all responses rated. Figure 4 shows the annotation result. As we expected, a fair number of examples (56%) have

[7]In the first run, all candidates were rated as equally good or bad in 18 examples. We updated and re-annotated 17 examples.

more than one *Best* or *Acceptable* responses. The first author reviewed 61 examples (3.5%) where r_1 was rated as *Bad* and fixed contradicting situational statements. Examples without *Best* responses were also reviewed and revised if necessary.

4 Experiments

We evaluate several baseline models on SUGAR to explore two questions concerned with the nature of the proposed task and dataset: (1) Is understanding of situational information required to identify proactive responses in SUGAR? (2) Can standard matching-based systems capture relevant situational information and solve the task?

4.1 Baselines

We evaluate a lexical-matching approach and several Transformer-based response selection systems. A variety of neural networks have been proposed for the task of response selection Tao et al. (2021), but we opted to focus on the direct application of pre-trained Transformers rather than equipping them with extra modules/resources. Pre-trained models have proven effective in conversation tasks with minimal adaptation (Budzianowski and Vulić, 2019) and even achieves the best performance in a response selection task (Han et al., 2021).

TF-IDF ranker: We used a lexical-matching baseline system that ranks response candidates by cosine similarity of TF-IDF vectors of context and a response candidate (Lowe et al., 2015). While this ranker is quite simple, it can outperform or perform on par with more complex supervised models in certain tasks (Thakur et al., 2021). We calculated TF-IDF weights on a training split with `scikit-learn` library.

Transformer ranker: We fine-tuned and evaluated four variants of Transformer-based rankers:

1. **BERT-FP** (Han et al., 2021): This model is an uncased $\text{BERT}_{\text{base}}$that underwent additional training on the Ubuntu Dialogue Corpus (Lowe et al., 2015). The training process includes unsupervised post-training and supervised fine-tuning. As of 2023, this model is one of the leading systems on the Ubuntu dataset.
2. **BERT** (Devlin et al., 2019): We also tested an uncased $\text{BERT}_{\text{base}}$without the additional training of Han et al. to analyze its benefits in our task. In the experiments of Hedayatnia et al. (2022), the BERT ranker performed similarly to BERT-FP.
3. **RoBERTa** (Liu et al., 2019): RoBERTa has the same architecture as BERT as a backbone but was trained using improved training configurations, resulting in better performance across multiple tasks and datasets. We used the pre-trained base model (12 layers $\approx$ 125M parameters)
4. **DeBERTa** (He et al., 2021b,a): DeBERTa is a model that improves upon BERT and RoBERTa by using disentangled attention mechanisms. In our experiments, we used the base DeBERTa v3 model (12 layers $\approx$ 86M parameters).

Following Han et al., we encoded a concatenation of input tokens, which will be explained in the next section, and a response option using these Transformer encoders. We then roduced a score of the option by a logistic regression classifier that takes the last hidden state of a special token, `[CLS]`, at the first position in the input. Model parameters were optimized using Adam (Kingma and Ba, 2015) to minimize the max-margin loss.

4.2 Experimental Setup

Input format: We concatenated context and a response candidate for the Transformer rankers. To address our questions, we experimented with three variants of context:

1. u: Utterance (u)-only
2. $u + S_1$: Utterance (u) plus relevant situation (S_1)
3. $u + S$: Utterance (u) plus relevant and irrelevant situation (S)

Training and Test: We performed five-fold cross-validation (training:validation:test=6:2:2).[8] For each round, we trained a Transformer ranker for 10 epochs with a batch size of 32 and evaluated the model by nDCG@3 on the validation split every epoch. We then selected the best checkpoint for evaluation. To stabilize training, we applied weight decay of 0.05, set the maximum gradient norm to 5.0, and used a linear learning rate scheduler with 5% ($\approx$ 20) warm-up steps. We further performed light-weight grid-search for hyperparameter tuning based on an average nDCG@3 score on validation

[8]We removed examples without *Bad* response options from the validation and test splits

System	Input	Precision@1	nDCG@3
TF-IDF	u	$.5993_{\pm.0223}$	$.8377_{\pm.0042}$
	$u+S_1$	$.7995_{\pm.0119}$	$.9289_{\pm.0042}$
	$u+S$	$.5683_{\pm.0121}$	$.8499_{\pm.0035}$
BERT-FP	u	$.6455_{\pm.0254}$	$.8799_{\pm.0076}$
	$u+S_1$	$.8386_{\pm.0280}$	$.9461_{\pm.0084}$
	$u+S$	$.6631_{\pm.0273}$	$.8869_{\pm.0094}$
BERT	u	$.7292_{\pm.0256}$	$.9102_{\pm.0071}$
	$u+S_1$	$.8637_{\pm.0109}$	$.9563_{\pm.0030}$
	$u+S$	$.7266_{\pm.0158}$	$.9110_{\pm.0038}$
RoBERTa	u	$.7178_{\pm.0273}$	$.9055_{\pm.0097}$
	$u+S_1$	$.8723_{\pm.0173}$	$.9596_{\pm.0059}$
	$u+S$	$.6992_{\pm.0230}$	$.9039_{\pm.0040}$
DeBERTa	u	$.7787_{\pm.0265}$	$.9305_{\pm.0074}$
	$u+S_1$	$.8981_{\pm.0112}$	$.9686_{\pm.0041}$
	$u+S$	$.7850_{\pm.0286}$	$.9314_{\pm.0084}$

Table 4: Average test scores over five-fold cross-validation.

splits, with learning rate $\in \{5e-5, 1e-5\}$, and margin for the max-margin loss $\in \{1.0, 0.5, 0.1\}$. One epoch of training took 1-2m on GeForce GTX TITAN X. We report the average Precision@1 and nDCG@3 on the test splits.

4.3 Results

Table 4 shows the average test scores over a five-fold cross-validation. Two general patterns can be observed: (1) the Transformer-based models, except for BERT-FP, outperformed the TF-IDF baseline, and (2) the systems that were provided with the request utterance u and relevant statements S_1 outperformed their counterparts with different input settings. In regard to the key questions, the results reveal several interesting findings:

1. Comparison of two input settings u and $u+S_1$ demonstrates that relevant situational information leads to a clear performance boost as expected (e.g., +0.13 in Precision@1 and +0.05 in nDCG@3 with BERT).
2. The performance gain in $u + S_1$ can be attributed to the increased word overlaps between the context and the correct responses, as indicated by the performance of the TF-IDF baseline. However, with the addition of distractors in the $u + S$ setting, the performance of the TF-IDF baseline dropped substantially (-0.20 in Precision@1 and -0.09 in nDCG@3). This result suggests that our dataset effectively avoids superficial clues, highlighting the importance of a higher-level understanding of situational statements.
3. Interestingly, in the $u + S$ setting, the performance of Transformer rankers also decreased significantly to the same level as their corresponding systems without situational statements in the input (the u setting).
4. Additional pre-training of BERT-FP was not effective in our task, which is consistent with the observation of Hedayatnia et al. (2022). We speculate that this is due to a domain mismatch of training corpora. BERT-FP is pre-trained on technical topics related to Ubuntu, whereas SUGAR concerns a wider range of topics in daily life.

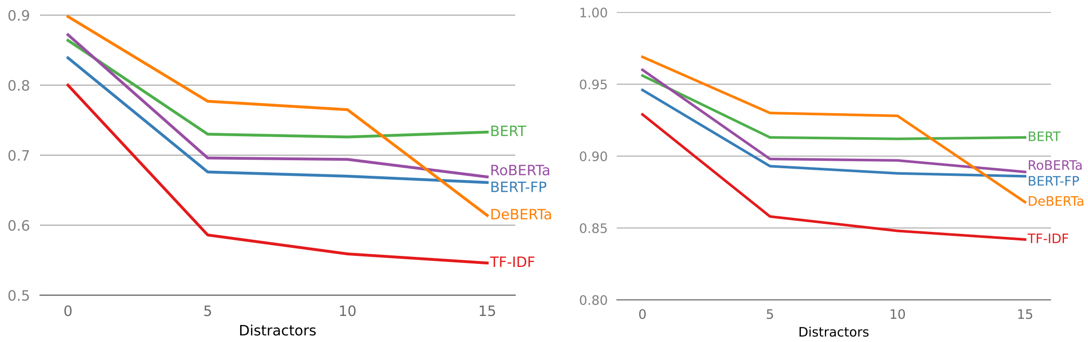
These findings provide valuable insights into our research questions. First, the understanding of relevant situational statements helps systems select proactive responses accurately, indicating that SUGAR is an effective resource for the development and evaluation of situated conversation systems. Secondly, it is challenging for Transformer rankers to identify useful clues from a mixture of relevant and irrelevant situational statements.

4.4 Robustness to Distractors

The results presented in the previous section indicate that Transformer rankers can be misled by irrelevant information. To explore this further, we evaluated these rankers with varying numbers of irrelevant situational statements (distractors).

In this experiment, we controlled the number of distractors by creating instances with 5, 10, and 15 distractors. Situational statements were randomly added as necessary. We trained and tested the same response rankers following the same setup, with the exception that we fixed the learning rate to 5e-5, which generally produced better results than 1e-5 in the main experiments. It is important to note that the first 1-7 distractors were adversarially selected (§3), while the remaining distractors were added at random.

Figure 5 displays the precision@1 and nDCG@3 scores of the response rankers. The performance of TF-IDF indicates that the addition of random distractors slightly increased the word overlap rates between input and distractor responses, but not substentially. However, as hypothesized, all systems demonstrated decreasing scores as more distractors were included. Interestingly, the performance of the advanced models, RoBERTa and DeBERTa, decreased drastically as more distractors were added ($0.87 \rightarrow 0.67$ for RoBERTa and $0.90 \rightarrow 0.61$ for



(a) Precision@1

(b) nDCG@3

Figure 5: Average test scores over five-fold cross-validation with different numbers of distractors

DeBERTa in Precision@1). We speculate that these models are powerful but also susceptible to overfitting spurious patterns between situational statements and response options, resulting in low test scores. In contrast, the BERT-based rankers were more robust to distractors, but their absolute performance remained low (Precision@1 of 0.73 and nDCG@3 of 0.91 for BERT). This finding highlights the need for future work to develop models that are more robust to the inclusion of irrelevant situational context.

5 Conclusion and Future Work

We proposed a task of situated proactive response selection for developing and evaluating conversational assistants that can help users proactively in various help-seeking scenarios. We constructed a dataset of 1.7k examples by crowdsourcing and semi-automatic generation.

There are several interesting directions for future research. First, as shown in our experiments, it is challenging to pick up relevant situational information and use it to reason about user requests and potential assistance. To achieve this, conversational systems will need to be equipped with world knowledge to effectively align situation information with an interaction. One promising approach is knowledge-based response models such as graph neural networks, which recently has shown to be effective in various NLP tasks (Zhang et al., 2020; Zhou et al., 2022; *inter alia*). Second, although we leveraged implicit goals only for soliciting proactive responses in data collection in this study, understanding of goals should be necessary for building better conversation engines as claimed in early studies (Allen and Perrault, 1980; *inter alia*). We believe SUGAR can facilitate future research in this direction.

Limitations

Data size: SUGAR is relatively small compared to recently published datasets. This is due to the complexity of our problem setting and annotation pipeline. We prioritized quality over quantity and performed multiple steps of manual intervention to reduce errors, false negatives, and annotation artifacts. These problems have been reported in various NLP tasks not limited to conversational tasks (Gururangan et al., 2018; Akama et al., 2020; Elazar et al., 2020). Nonetheless, our experiment has shown that pre-trained Transformer models can be trained to outperform a TF-IDF ranker by a clear margin, which is encouraging. In addition, we could automatically induce noisy but large-scale training instances from existing resources, for example, by harvesting event pairs that can be used as u and r from event knowledge bases such as ATOMIC$^{20}_{20}$and generating situation statements using our generator (§3).

Representation of situation information: In SUGAR, situation information is represented in textual expressions. In real-world applications, such information could be collected via external APIs (e.g., calendar and map) and sensors (e.g., camera) and stored in non-textual forms. Our study is a proof-of-concept that shows the understanding of situational information is very important for response selection. Future research should explore ways to process situation information that is expressed in other forms of data (e.g., structured texts, numbers, images). Even if the value is structured or images, we could transform them into textual forms as done in data-to-text research (Shen et al., 2020; Miura et al., 2021). Besides, we acknowledge that situational information is often under-specified in SUGAR because some information is considered to be common-sense (e.g., a room has a door) or presupposed (e.g., "Please open the door" presupposes that the door is closed.), and such information was not explicitly stated by human annotators during data collection. Therefore, response selection systems should be equipped with a mechanism to handle implicit knowledge to solve the task.

Ethical Considerations

Undesired bias and abusive content: A multitude of sources have reported that data-driven conversational systems can (re)produce undesired bias or abusive language existing in language resources used for development. To minimize such a risk, we carefully curated conversation examples in SUGAR. Our target task is response selection, where systems only produce language in a pre-compiled response list, and therefore, it is not likely that resulting systems yield harmful content. However, users of SUGAR should be cautious when it is used for developing generation systems in future work.

Human subjects: Crowd workers in Amazon Mechanical Turk (MTurk) participated in our data collection pipeline. Our annotation tasks were reviewed by the institutional review process before being published in MTurk to avoid ethical issues. We did not collect any personally identifiable information of workers other than (anonymized) Turker IDs. Task rewards were decided by several rounds of trials so that workers can receive at least $6.50 hourly.

Use of external data and tools: We used external datasets such as ATOMIC$^{20}_{20}$and ConceptNet and tools such as spaCy and Transformers library. We have confirmed that the use of these resources for our research does not violate usage restrictions.

Acknowledgments

We thank Yonatan Bisk, Benjamin Van Durme, Lori Levin, and the anonymous reviewers for their feedback.

References

Reina Akama, Sho Yokoi, Jun Suzuki, and Kentaro Inui. 2020. Filtering noisy dialogue corpora by connectivity and content relatedness. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 941–958, Online. Association for Computational Linguistics.

James F. Allen and C.Raymond Perrault. 1980. Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143–178.

Tawfiq Ammari, Jofish Kaye, Janice Y. Tsai, and Frank Bentley. 2019. Music, search , and IoT: How people (really) use voice assistants. *ACM Transactions on Computer-Human Interaction*, 26(3):17:1–17:28.

Nicola Bellini and Laetitia Convert. 2016. The concierge. tradition, obsolescence and innovation in tourism. *Symphonya. Emerging Issues in Management*, 0(2):17.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning. *The Eighth International Conference on Learning Representations*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense Transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2 - how can i help you? Towards the use of pretrained language models for task-oriented dialogue systems. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 15–22, Hong Kong. Association for Computational Linguistics.

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 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4516–4525, Hong Kong, China. Association for Computational Linguistics.

Wonae Cho, Robert T Sumichrast, and Michael D Olsen. 1996. Expert-system technology for hotels: Concierge application. *Cornell Hotel and Restaurant Administration Quarterly*, 37(1):54–60.

Herbert H. Clark and Edward F. Schaefer. 1989. Collaborating on contributions to conversations. In *Language Processing in Social Context*, volume 54 of *North-Holland Linguistic Series: Linguistic Variations*, pages 123–152. Elsevier.

Paul A. Crook, Shivani Poddar, Ankita De, Semir Shafi, David Whitney, Alborz Geramifard, and Rajen Subba. 2019. SIMMC: Situated interactive multi-modal conversational data collection and evaluation platform. *IEEE Workshop on Automatic Speech Recognition and Understanding*.

Marie-Catherine de Marneffe, Scott Grimm, and Christopher Potts. 2009. Not a simple yes or no: Uncertainty in indirect answers. In *Proceedings of the SIGDIAL 2009 Conference*, pages 136–143, London, UK. Association for Computational Linguistics.

Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. 2020. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review*, pages 1–56.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional Transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, Shrimai Prabhumoye, Alan W. Black, Alexander Rudnicky, Jason Williams, Joelle Pineau, Mikhail Burtsev, and Jason Weston. 2020. The second conversational intelligence challenge (ConvAI2). In *The NeurIPS '18 Competition*, The Springer Series on Challenges in Machine Learning, pages 187–208, Cham. Springer International Publishing.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-powered conversational agents. *The Seventh International Conference on Learning Representations*.

Yanai Elazar, Victoria Basmov, Shauli Ravfogel, Yoav Goldberg, and Reut Tsarfaty. 2020. The extraordinary failure of complement coercion crowdsourcing. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, pages 106–116, Online. Association for Computational Linguistics.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 422–428, Marseille, France. European Language Resources Association.

Deepanway Ghosal, Siqi Shen, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. 2022. CICERO: A dataset for contextualized commonsense inference in dialogues. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 5010–5028, Dublin, Ireland. Association for Computational Linguistics.

Raymond W Gibbs and Gregory A Bryant. 2008. Striving for optimal relevance when answering questions. *Cognition*, 106(1):345–369.

Andrew S. Gordon and Jerry R. Hobbs. 2004. Formalizations of commonsense psychology. *AI Magazine*, 25(4):49.

H. P. Grice. 1975. Logic and conversation. In *Speech Acts*, number 3 in Syntax and Semantics, pages 41 – 58. Academic Press, New York, NY.

Prakhar Gupta, Yulia Tsvetkov, and Jeffrey Bigham. 2021. Synthesizing adversarial negative responses for robust response ranking and evaluation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3867–3883, Online. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Janghoon Han, Taesuk Hong, Byoungjae Kim, Youngjoong Ko, and Jungyun Seo. 2021. Fine-grained post-training for improving retrieval-based dialogue systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1549–1558, Online. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. *arXiv*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. DeBERTa: Decoding-enhanced BERT with disentangled attention. *The Ninth International Conference on Learning Representations*.

Behnam Hedayatnia, Di Jin, Yang Liu, and Dilek Hakkani-Tur. 2022. A systematic evaluation of response selection for open domain dialogue. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 298–311, Edinburgh, UK. Association for Computational Linguistics.

Matthew Henderson, Iñigo Budzianowski Paweł and Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019. A repository of conversational datasets. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 1–10, Florence, Italy. Association for Computational Linguistics.

Beth Ann Hockey, Deborah Rossen-Knill, Beverly Spejewski, Matthew Stone, and Stephen Isard. 1997. Can you predict answers to yes/no questions? Yes, no and stuff. In *Proceedings of the Fifth European Conference on Speech Community and Technology*, pages 2267–2270, Rhodes, Greece.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. COMET-ATOMIC 2020: On symbolic and neural commonsense knowledge graphs. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 6384–6392, Online. AAAI Press.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *The Third International Conference for Learning Representations*.

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*, pages 582–595, Minneapolis, Minnesota. Association for Computational Linguistics.

Klaus Krippendorff. 2006. Reliability in Content Analysis: Some Common Misconceptions and Recommendations. *Human Communication Research*, 30(3):411–433.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jia Li, Chongyang Tao, Wei Wu, Yansong Feng, Dongyan Zhao, and Rui Yan. 2019. Sampling matters! An empirical study of negative sampling strategies for learning of matching models in retrieval-based dialogue systems. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 1291–1296, Hong Kong, China. 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. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv*.

Annie Louis, Dan Roth, and Filip Radlinski. 2020. "I'd rather just go to bed": Understanding indirect answers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 7411–7425, Online. Association for Computational Linguistics.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 285–294, Prague, Czech Republic. Association for Computational Linguistics.

Yasuhide Miura, Yuhao Zhang, Emily Tsai, Curtis Langlotz, and Dan Jurafsky. 2021. Improving factual completeness and consistency of image-to-text radiology report generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5288–5304, Online. Association for Computational Linguistics.

Seungwhan Moon, Satwik Kottur, Paul 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. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1103–1121, Barcelona, Spain (Online). International Committee on Computational Linguistics.

C. Raymond Perrault. 1980. A plan-based analysis of indirect speech act. *American Journal of Computational Linguistics*, 6(3-4):167–182.

Elahe Rahimtoroghi, Jiaqi Wu, Ruimin Wang, Pranav Anand, and Marilyn Walker. 2017. Modelling protagonist goals and desires in first-person narrative. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 360–369, Saarbrücken, Germany. Association for Computational Linguistics.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. WinoGrande: An adversarial Winograd Schema Challenge at scale. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, New York City, USA. AAAI Press.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. ATOMIC: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035. Association for the Advancement of Artificial Intelligence.

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press, New York, NY, USA.

Iulian Vlad Serban, Ryan Lowe, Peter Henderson, Laurent Charlin, and Joelle Pineau. 2018. A survey of available corpora for building data-driven dialogue systems. *Dialogue and Discourse*, 9(1):1–49.

Xiaoyu Shen, Ernie Chang, Hui Su, Cheng Niu, and Dietrich Klakow. 2020. Neural data-to-text generation via jointly learning the segmentation and correspondence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7155–7165, Online. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4444–4451, San Francisco, California, USA. AAAI Press.

Jon Stevens, Anton Benz, Sebastian Reuße, and Ralf Klabunde. 2015. A Strategic reasoning model for generating alternative answers. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 534–542, Beijing, China. Association for Computational Linguistics.

Chongyang Tao, Jiazhan Feng, Rui Yan, Wei Wu, and Daxin Jiang. 2021. A survey on response selection for retrieval-based dialogues. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 4619–4626. International Joint Conferences on Artificial Intelligence Organization.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Sarah Theres Völkel, Daniel Buschek, Malin Eiband, Benjamin R. Cowan, and Heinrich Hussmann. 2021. Eliciting and analysing users' envisioned dialogues with perfect voice assistants. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–15, New York, NY, USA. Association for Computing Machinery.

Traci Walker, Paul Drew, and John Local. 2011. Responding indirectly. *Journal of Pragmatics*, 43(9):2434–2451.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The Dialog State Tracking Challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 404–413, Metz, France. Association for Computational Linguistics.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2020. Grounded conversation generation as guided traverses in commonsense knowledge graphs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2031–2043, Online. Association for Computational Linguistics.

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? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.

Dengyong Zhou, Qiang Liu, John Platt, and Christopher Meek. 2014. Aggregating ordinal labels from crowds by minimax conditional entropy. In *Proceedings of the 31st International Conference on Machine Learning*, pages 262–270, Beijing, China. ACM Press.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 708–713, Brussels, Belgium. Association for Computational Linguistics.

Pei Zhou, Karthik Gopalakrishnan, Behnam Hedayatnia, Seokhwan Kim, Jay Pujara, Xiang Ren, Yang Liu, and Dilek Hakkani-Tur. 2022. Think before you speak: Explicitly generating implicit commonsense knowledge for response generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 1237–1252, Dublin, Ireland. Association for Computational Linguistics.

A Manual Annotation

We recruited non-expert crowd workers in Amazon Mechanical Turk in annotation steps (2-5). In all steps, crowd workers were required to meet the following qualification requirements: (i) Their number of tasks approved $\geq$ 5k, (ii) the task approval rate $\geq$ 99%, (iii) their location is the US, and (iv) they answer an exercise question correctly. Figure 6 shows the annotation interface.

Two of the authors were involved in the annotation steps (1), (4), (5), and (8). They are ESL with a degree in computer science from a school in the US (one holds a master's degree, and the other holds a Ph.D.). They all have backgrounds in NLP/CL research.

B Distractor Selection

This section presents the technical details of the distractor selection method (Step 7). Below, tunable parameters such as thresholds on scores and the number of iterations were empirically selected based on several pilot runs.

B.1 Response Selection

Our method selects distractor responses from all the responses in the dataset in two steps: We first create an initial dataset by a light-weight method (Algorithm 1) and then perform adversarial filtering (Algorithm 2).

First Step (Algorithm 1)

The objective of the first step is to avoid including false-negative responses (Lines 3-6). We discard responses that are too similar to r_1 in terms of the overlap coefficient of content words (noun, verb, adjective, and adverb).

$$\text{Overlap}(x, y) = \frac{|\text{CW}(x) \cap \text{CW}(y)|}{\min\left(|\text{CW}(x)|, |\text{CW}(y)|\right)}, \tag{1}$$

where $\text{CW}(x)$ is a set of content words in x. We set the threshold of overlap coefficient to 0.75. We use the same constraint on their goal texts. We also measure their closeness by the cosine similarity of their sentence embeddings (denoted as EmbSim) and discard candidates whose similarity is 0.5 or higher. We then sample $m-1$ responses from this filtered response pool one by one (Lines 11-15). To diversify response options, we remove similar responses to the picked one from the pool based on overlap coefficient (Line 16-19).

Second Step (Algorithm 2)

We then perform $J = 3$ rounds of adversarial filtering. Our method is a slightly modified version of the algorithm used by Bhagavatula et al. (2020). In each round, we split the dataset into $K = 10$ folds (Line 6), and for each split, we train a binary logistic regression classifier that takes sentence embeddings of u, S_1, and a response candidate $r \in R$ (Line 8). We pre-compute their sentence embeddings with the pre-trained Sentence-Transformers (Reimers and Gurevych, 2019) with MPNet (Song et al., 2020). Once the classifier is trained, we score response candidates in each example and identify distractors whose scores are lower than that of the reference response r_1 plus a margin $\gamma = 0.05$. We replace these *easy* distractors with more confusing ones (Line 14-16). In this way, we repeatedly update the dataset (Line 17) and output the final result (Line 18).

B.2 Situation Selection

Next, we update S_1, which only contains relevant information to u and r_1, to include l statements in total such that some of them are associated with distractors or not directly related to the conversation. Otherwise, reference responses can be easily identified by superficial clues. Having irrelevant situational statements is also for simulating real use cases, where a conversational system has access to a wide range of sensory information or external APIs, but most of them are unimportant for addressing a user's request.

It is required that (a) additional situational statements do not disqualify the reference response,

Algorithm 1 Create an initial dataset by light-weight filtering

Input: m, Dataset $\mathcal{D} = \{(u^{(i)}, g^{(i)}, r_1^{(i)}, S_1^{(i)})\}_{i=1,\cdots,N}$, ▷ N := num. of examples in the dataset.

Output: $\mathcal{D}' = \{(u^{(i)}, g^{(i)}, R^{(i)}, S_1^{(i)})\}_{i=1,\cdots,N}$ ▷ $R^{(i)} := \{r_1^{(i)}, \cdots, r_m^{(i)}\}$ ▷ Initial dataset

function InitDataset($m, \mathcal{D}$)
  $\mathcal{D}' \leftarrow \varnothing$
  for $i : 1..N$ **do**
    $\mathcal{P} \leftarrow \{r_1^{(j)}\}_{j=i,\cdots,i-1,i+1,\cdots,N}$ ▷ All the responses in $\mathcal{D}$ but $r_1^{(i)}$
    `# (1) Remove too similar responses`
    for $j : 1..N$ **do**
      if i=j **then**
        continue
      if Overlap$(u^{(i)}, u^{(j)}) \geq 0.75$ **or** Overlap$(g^{(i)}, g^{(j)}) \geq 0.75$
        or EmbSim$(u_1^{(i)}, r_1^{(j)}) \geq 0.5)$ **then**
        Remove $r_1^{(j)}$ from $\mathcal{P}$
    `# (2) Pick` $m-1$ `similar responses`
    $R^{(i)} \leftarrow \{r_1^{(i)}\}$
    for $j : 1..m-1$ **do**
      Sample $r \in \mathcal{P}$
      Add r to $R^{(i)}$
      `# (3) Remove similar responses from the pool`
      for all $r' \in \mathcal{P}$ **do**
        if Overlap$(r, r') \geq 0.75$ **then**
          Remove r' from $\mathcal{P}$
    Add $(u^{(i)}, g^{(i)}, R^{(i)}, S_1^{(i)})$ to $\mathcal{D}'$
  return $\mathcal{D}'$

and (b) they do not contradict others. To this end, we again use sentence embeddings with keyword-based heuristics. We first combine the statements associated with distractor responses and create a pool of candidates. Here, we drop statements that are similar to the response candidates in terms of the overlap coefficient of content words with a threshold of 0.75. We also used manually defined keywords to discard situational statements that tend to contradict others (e.g., the time is midnight, the user is injured, etc.). We then iterate over six categories and pick situational statements from the pool one by one. We score statement s of category c using the function below:

$$f(s; R, S') = \max_{r \in R} \text{EmbSim}(s, r) - \max_{s' \in S'_c} \text{EmbSim}(s, s') - \frac{1}{2} \max_{s' \in S'_{\mathcal{C} \setminus \{c\}}} \text{EmbSim}(s, s'), \quad (2)$$

where S' is the current situational statements, $S'_c \subset S'$ represents the statements in S of category c, and $\mathcal{C}$ denotes a set of situation categories. We pick distractor statements until we exhaust all the candidates in the pool or the maximum score does not reach 0. We then draw statements from the entire dataset in the same way until $|S|$ reaches $l = 12$. For time, date, behavior, and location categories, we pick zero or one statement as those categories are not likely to have more than one value.

C Response Selection Example

Table 5 shows a conversation example included in SUGAR.

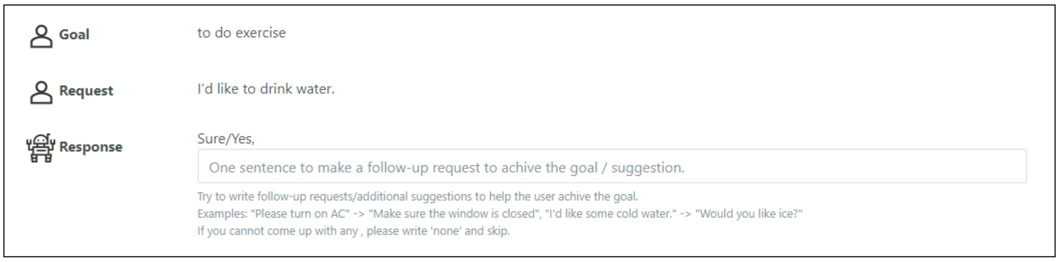
Algorithm 2 Adversarial filtering (AF) for R

Input: m, Dataset $\mathcal{D} = \{(u^{(i)}, g^{(i)}, r_1^{(i)}, S_1^{(i)})\}_{i=1,\cdots,N}$, ▷ N := number of examples in the dataset.
Output: $\mathcal{D}' = \{(u^{(i)}, g^{(i)}, R^{(i)}, S_1^{(i)})\}_{i=1,\cdots,N}$ ▷ $R^{(i)} := \{r_1^{(i)}, \cdots, r_m^{(i)}\}$

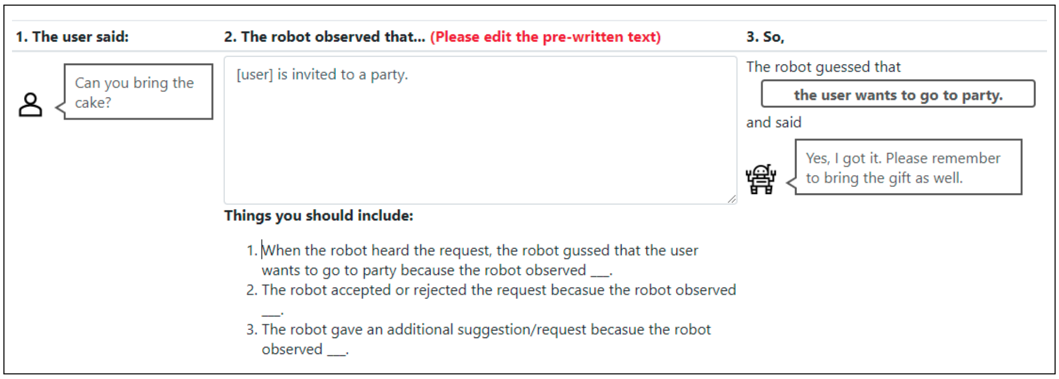
$\mathcal{P} \leftarrow \{(r_0)_i\}$ ▷ All responses in $\mathcal{D}$
`(1) Create an initial dataset` D_0
$\mathcal{D}_0 \leftarrow$ InitDataset$(m, \mathcal{D})$ ▷ See Algorithm 1
`(2) Run AF for` J `rounds`
for j : 1..J **do** ▷ We set $J = 3$
  Split $\mathcal{D}_{j-1}$ into K-folds $\{(\mathcal{T}^k, \mathcal{V}^k)\}_{k=1,\cdots,K}$ ▷ We set $K = 10$
  for k : 1..K **do**
    Train a binary logistic regression classifier $\mathcal{M}$ on $\mathcal{T}^k$
    for all $(u, g, R, S_1) \in \mathcal{V}^k$ **do**
      for all $r \in R \setminus \{r_1\}$ **do**
        (f: $\mathcal{M}$'s score function)
        if $f(r) + \gamma \leq f(r_1)$ **then** ▷ γ is a margin, which we set to 0.05.
          Remove r from R
          Pick r' s.t. $f(r') - \gamma > f(r_1)$
          Add r' to R
          Update $\mathcal{V}^k$ with the new R
  $\mathcal{D}_j \leftarrow \bigcup_{k=1}^{K} \mathcal{V}_k$
$\mathcal{D}' \leftarrow \mathcal{D}_K$ ▷ End

Utterance	Please turn on the TV.
Situations	It is evening now. [user] is home. [user] is in the living room. [user] is sitting on the couch. [user] has a TV in the house. [user] has an outfit on the bed. [user] has drinks and snacks in the kitchen. [user] has game cards on the shelf. The TV is off. [someone]'s birthday is today. There are several sports games available to watch. There is a basketball game scheduled.
Responses	Sure. Would you like me to check today's sports listings? *(Best)* Sure. Shall I pour a drink and bring some snacks for the game? *(Acceptable)* Sure, shall I select an outfit for you? *(Bad)*

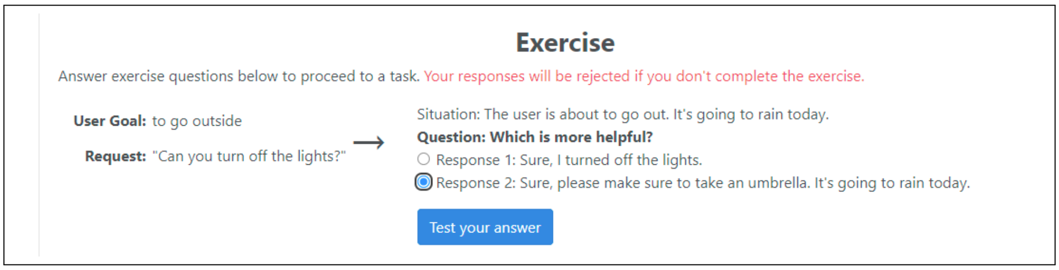
Table 5: Response selection example in SUGAR. Each example has 12 situational statements, some of which are distractors. [user] and [someone] are placeholders to denote person names.



(a) Step 3 (Response Collection)



(b) Step 5 (Situation Collection II). The output of Step 2 is provided as an initial value.



(c) Exercise question. (This figure is for Step 3.)

Figure 6: Annotation interface for data creation. In addition to annotation guidelines, we provide one exercise question per task to train crowd workers. We used exercise questions in all the crowdsourced annotation tasks in our pipeline (c).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations

☑ A2. Did you discuss any potential risks of your work?
Limitations

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract, Section 1

☑ A4. Have you used AI writing assistants when working on this paper?
Grammarly (https://www.grammarly.com/) and ChatGPT (https://chat.openai.com/) for proofreading and improving clarity (the whole paper).

B ☑ Did you use or create scientific artifacts?

Section 3

☑ B1. Did you cite the creators of artifacts you used?
Sections 3 and 4

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Limitations

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Limitations

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Ethical Considerations and Appendix A

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Sections 1 and 3

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 3

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Sections 3 and 4, Appendix.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
Section 3

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Appendix presents a few screenshots of the annotation interface. We will release more details in our GitHub repository upon internal approval.

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Appendix A

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Ethical Considerations

☒ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
As our annotation does not collect PI, our annotation study just underwent an internal review process (not IRB).

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Appendix A