Controllable Abstractive Dialogue Summarization with Sketch Supervision

Chien-Sheng Wu*[1], Linqing Liu*[2], Wenhao Liu[1], Pontus Stenetorp[2], Caiming Xiong[1]

[1] Salesforce Research [2] University College London

{linqing.liu, p.stenetorp}@cs.ucl.ac.uk, {wu.jason, wenhao.liu, cxiong}@salesforce.com

Abstract

In this paper, we aim to improve abstractive dialogue summarization quality and, at the same time, enable granularity control. Our model has two primary components and stages: 1) a two-stage generation strategy that generates a preliminary *summary sketch* serving as the basis for the final summary. This summary sketch provides a weakly supervised signal in the form of pseudo-labeled interrogative pronoun categories and key phrases extracted using a constituency parser. 2) A simple strategy to control the granularity of the final summary, in that our model can automatically determine or control the number of generated summary sentences for a given dialogue by predicting and highlighting different text spans from the source text. Our model achieves state-of-the-art performance on the largest dialogue summarization corpus SAMSum, with as high as 50.79 in ROUGE-L score. In addition, we conduct a case study and show competitive human evaluation results and controllability to human-annotated summaries.

1 Introduction

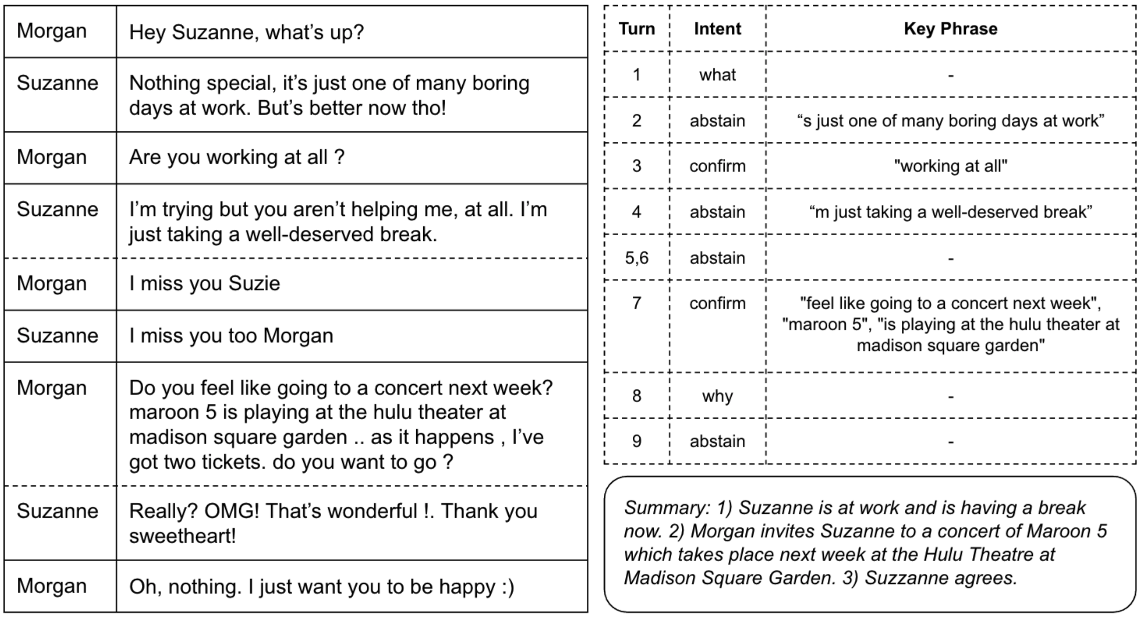
Text summarization aims to produce an abridged version of the input text by distilling its most critical information. In particular, abstractive – as opposed to extractive – summarization requires generative models with a high level of semantic understanding, as the output words do not necessarily appear in the source text. While it is more challenging, it gives more flexibility to a summary compared to extractive summarization models (Zhang et al., 2018). Significant research efforts have been focused on summarization of single-speaker documents such as text documents (Liao et al., 2018), News (Hermann et al., 2015; Nallapati et al., 2016; See et al., 2017) or scientific publications (Qazvinian and Radev, 2008; Nikolov et al., 2018). However, dialogue summarization has not received much attention despite the prevalence of dialogues (text messages, email, social media, etc.) and the vast application potential of dialogue summarization systems.

Since dialogue language is inherently different from written text, it poses a unique set of challenges (Zechner, 2001): 1) *Distributed information across multiple speakers.* The most important information is usually scattered across several conversation turns from different speakers, while in articles it mostly presents in titles or the first few sentences. 2) *Boundary detection.* In each turn pauses do not always match linguistic sensible segments; it is difficult to identify various critical information across turns due to surrounding non-content noise and disfluency. 3) *Modeling interactions between speakers.* The speaker interaction plays an important role as it would imply the current dialog state and the status of the next speaker. If we directly apply neural abstract summarization models which mostly encode the whole input only as a source sequence, the flow of the dialogue would be overlooked (Pan et al., 2018). Previous methods (Goo and Chen, 2018; Liu et al., 2019) rely on explicit annotations to capture the logic of the dialogue, however, such annotations are not always available in datasets and additional labeling is cumbersome.

To solve these challenges, we propose CODS, a COntrollable abstractive Dialogue Summarization model equipped with sketch generation. We first automatically create a summary sketch that contains user intent information and essential key phrases that may appear in summary. It identifies the interaction between speakers and salient information in each turn. This summary sketch is prefixed to the human-annotated summary while fine-tuning a generator, which provides weak supervision as the final summary is conditioned on the

*Equal contribution. Work mainly done when Linqing Liu was an intern at Salesforce Research.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 5108–5122
August 1–6, 2021. ©2021 Association for Computational Linguistics



Morgan	Hey Suzanne, what's up?
Suzanne	Nothing special, it's just one of many boring days at work. But's better now tho!
Morgan	Are you working at all ?
Suzanne	I'm trying but you aren't helping me, at all. I'm just taking a well-deserved break.
Morgan	I miss you Suzie
Suzanne	I miss you too Morgan
Morgan	Do you feel like going to a concert next week? maroon 5 is playing at the hulu theater at madison square garden .. as it happens , I've got two tickets. do you want to go ?
Suzanne	Really? OMG! That's wonderful !. Thank you sweetheart!
Morgan	Oh, nothing. I just want you to be happy :)

Turn	Intent	Key Phrase
1	what	-
2	abstain	"s just one of many boring days at work"
3	confirm	"working at all"
4	abstain	"m just taking a well-deserved break"
5,6	abstain	-
7	confirm	"feel like going to a concert next week", "maroon 5", "is playing at the hulu theater at madison square garden"
8	why	-
9	abstain	-

Summary: 1) Suzanne is at work and is having a break now. 2) Morgan invites Suzanne to a concert of Maroon 5 which takes place next week at the Hulu Theatre at Madison Square Garden. 3) Suzzanne agrees.

Figure 1: An input and output example. Given the dialogue, we first construct a summary sketch with intent and key phrase information for each turn, and then split the dialogue into several segments (marked with dashed lines on the left hand side) for model controllability and interpretability.

generated summary sketch. In addition, we propose a length-controllable generation method specifically for dialogue summarization. Desired lengths of summaries strongly depend on the amount of information contained in the source dialogue and granularity of information the user wants to understand (Kikuchi et al., 2016). We first segment the dialogue into different segments by matching each summary sentence linearly to its corresponding dialogue context. Then we train our model to generate only one sentence for each dialogue segment. This strategy makes use of the distributed information of the dialogue and make the generated summaries more trackable.

We base our model on BART-xsum (Lewis et al., 2019), which is first pre-trained with unsupervised denoising objectives, and further fine-tuned on the News summarization corpus XSUM (Narayan et al., 2018). We evaluate our approach on SAMSum (Gliwa et al., 2019), the largest dialogue summarization dataset. Experimental results show that CODS achieves state-of-the-art dialogue summarization performance on several automatic metrics. The main contributions of this work[1] are: 1) We propose a two-stage strategy that uses artificial summary sketch as weak supervision, 2) we introduce a text-span based conditional generation approach to control the granularity of generated dialogue summaries without human-written summaries at different detail levels, and 3) we conduct comprehensive case study and human evaluation to show that CODS can achieve consistent and informative summary, especially for controllable summary, where existing models either cannot do it or do it poorly.

2 Methodology

Our model is based on pre-trained generative language models (Section 2.1). Given an input dialogue history, our model first generates a summary sketch that serves as additional weakly supervised signal for the final summary (Section 2.2). Then it predicts the text span cutoffs over the entire dialogue and generates summaries accordingly (Section 2.3). We define the conversational history input as $D = \{X_1, X_2, \dots, X_N\}$, where each X_i has a sequence of words, N is the total numbers of dialogue turns, and the input may contain more than two speakers. We intend to generate M-sentence dialogue summary $Y = \{Y_1, \dots, Y_M\}$ that is suppose to be briefer than the overall dialogue history.

2.1 Generative Pre-trained Language Models

As a first, our model needs transform a conversational history input into a dialogue summary. Re-

[1] Our code is released at `https://github.com/salesforce/ConvSumm`

cently, self-supervised pretrained language models have been employed as encoders and decoders since they (Radford et al., 2019; Yang et al., 2019; Dong et al., 2019) have achieved remarkable success across many NLP tasks. For general text summarization, this has also been the case with models such as BART (Lewis et al., 2019) and PEGASUS (Zhang et al., 2019a). However, there are no results reported for self-supervised pretrained language models applied to dialogue summarisation, and people have argued that there is an intrinsic difference of linguistic patterns between human conversations and written text (Wolf et al., 2019b; Wu et al., 2020a; Wu and Xiong, 2020). We would like to answer the question which generative language model is the best base model for dialogue summarization tasks.

2.2 Sketch Construction

Conversational data, unlike news or scientific publications, includes lots of non-factual sentences such as chit-chats and greetings. Removing these least critical information in the dialogues could potentially help the model better focus on the main content. Based on this hypothesis, we combine a syntax-driven sentence compression method (Xu and Durrett, 2019) with neural content selection.

Another potentially useful attribute for the conversational data is each dialogue turn inherently encodes user intent. However, unlike task-oriented dialogue systems, which have explicit annotated intents (e.g., book flight and check account), dialogue summarization data rarely have such labels. Thus we use a few heuristics with Snorkel (Ratner et al., 2019) to programmatically label each turn with a predefined interrogative pronoun category. The generated intents and the compressed dialogues together constitutes the summary sketch as weakly-supervised signals.

To the best of our knowledge, in general, there is no non-task-oriented established label set. Thus we draw upon the FIVE Ws principle, which often mentioned in journalism and research investigation, in that a passage can only be considered as complete if it answers these questions starting with such interrogative words (Hart). We adapt this principle to the dialogue scenario and identify a set of interrogative pronouns to support diverse enough user intents of all utterances, serving as the dialogue's logic. For example, in Figure 1, Morgan asked Suzanne "Do you feel like going to a concert next week?" One can expect that Suzanne will confirm her willingness in the next utterance. We define such dialogue intent categories including why, what, where, confirm, and abstain. More information for each category is shown in the Appendix (A.1).

To compress and remove noisy sub-sentences in the dialog, we first use a trained constituency parser (Kitaev and Klein, 2018) to parse each utterance. Then we compare the parsed phrases with the ground-truth summary to find their longest common sub-sequence (lcs), we set a threshold to filter and remove non-meaningful words (e.g., stop words) in lcs. Note that there are circumstances where the whole utterance is noisy and removable. Overall, we construct a summary sketch by concatenating utterance index, user intent label, and compressed utterance within the entire dialogue history into a string, ending with a special token, "TL;DR". Take Figure 1 as an example, the summary sketch is "1 what 2 abstain 's just one of ... square garden 8 why 9 abstain TL;DR". We train our model first to generate this summary sketch and then generate the final summary in an autoregressive way. We use TL;DR token to distinguish sketch and final summary during inference time.

2.3 Controllability

Due to the success of controllable language modeling (Keskar et al., 2019), the ability to control text summarization in the News domain has gradually been attracting attention (Fan et al., 2018; Liu et al., 2018) The high-level intuition for our solution is that if we can control a generative model only to generate one sentence as output for a partially-highlighted input, we can control the number of output sentences by choosing how to highlight the input. We highlight each dialogue split using the special token $< hl >$. For example, in Figure 1, we generate the first summary sentence for the first segment from turn one to four, and the second and third from turn five to seven and turn eight to nine, respectively (separated by the dashed lines). This way, we can not only gain the summary controllability but also make the generation more interpretable.

The next challenge is, during training, we have to find a mapping between each sentence in a reference summary to its corresponding dialogue split. In other words, how do we know where to insert the highlighting tokens? We do so by training a dialogue-turn-level binary classifier (detailed below) that predicts whether each turn is a cutting point (i.e., dialogue segmentation). Our observation is that sentences within a reference summary usually have a strong temporal dependency, that is, people summarize the dialogue almost linearly. We use a simple approach to find the cutting points: the highest similarity score between conversations and each summary sentence. The cutting point

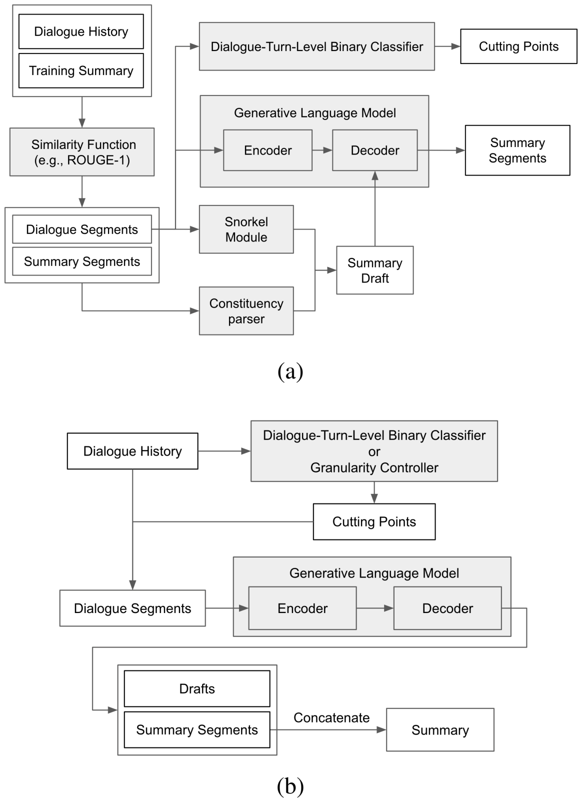
$$t_m = \arg\max_t \mathrm{SIM}(X_{c_m:t}, Y_m), \qquad (1)$$

where SIM could be any similarity functions (we use ROUGE-1), and c_m is the accumulated turn index ($c_1 = 1$ and $c_m = t_{m-1}$) that indicates which part of a dialogue has been covered. Note that for a summary with M sentences, we only need to decide $M - 1$ cutting points. With the pseudo labels (t_m) provided by this heuristic, we formulate the dialogue segmentation problem into a binary classification problem. Specifically, we train a classifier C, which takes dialogue history as input and predicts whether each dialogue turn is a cutting point. We prefix each dialogue turn with a separation token as input to the classifier.

$$\begin{aligned} H &= C([x_{sep}, X_1, x_{sep}, X_2, \dots]) \in \mathbb{R}^{N \times d_{emb}}, \\ \hat{P} &= \mathrm{Sigmoid}(W_1(H)) \in \mathbb{R}^N. \end{aligned} \qquad (2)$$

The classifier output H is the representations of those separation tokens, and each of them is a d_{emb} dimension vector. $W_1 \in \mathbb{R}^{d_{emb} \times 1}$ is a trainable linear mapping. The $\hat{P}$ is the predicted segment probability that is trained with binary cross-entropy loss. We use a BERT-base model (Devlin et al., 2018) as classifier and the i-th cutting point is triggered if $\hat{P}_i > 0.5$. This prediction means that our model can automatically determine how many sentences should be generated in the final summary. If no cutting point is triggered, we generate a one-sentence summary. If one cutting point is triggered, we will have a two-sentence summary, and so forth.

Finally, we can control the number of output summary sentences by controlling the dialogue split. Specifically, we first decide the expected number of output sentences (e.g., K), and then we choose the top $K-1$ indexes with highest probabilities in segmentation probability $\hat{P}$. We use these $K - 1$ indexes as cutting points. We can also generate one-sentence summary by clipping the whole dialogue with one pair of highlighting tokens at the beginning and the end of a dialogue (we call this setting as CODS-1).



(a)

(b)

Figure 2: (a) Training and (b) inference block diagrams of CODS. Grey boxes are trainable functions.

2.4 Overall Generation

The overall training and inference block diagrams are shown in Figure 2. CODS follows a standard encoder-decoder framework. During training, we use dialogue segmentation to add highlighting tokens for each summary sentence. We take the highlighted dialogue history as input and train our model to generate its corresponding summary sketch and summary sentence. For example in Figure 1, the first summary sentence, we input the whole dialogue with added highlighting tokens both at the beginning of the first turn and at the end of the fourth turn, and generate output that contains the corresponding summary sketch "1 what 2 abstain ... well-deserved break" and the first summary sentence "Suzanne is at work and is having a break now." The entire model is trained using cross-entropy loss for the generated tokens. During inference, we first use the trained binary classifier to predict cutting points. Then, we use the predicted segmentation to add highlighting tokens into a dialogue. Finally, after generating multiple summary sentences separately, we concatenate them to be the final summary.

	ROUGE-1	**ROUGE-2**	**ROUGE-L**
Longest-3*	32.46	10.27	29.92
Pointer Generator (See et al., 2017)*	37.27	14.42	34.36
Fast Abs RL (Chen and Bansal, 2018)*	41.03	16.93	39.05
Transformer (Vaswani et al., 2017)*	42.37	18.44	39.27
DynamicConv (Wu et al., 2019b)*	41.07	17.11	37.27
DynamicConv + GPT-2 emb*	45.41	20.65	41.45
D-HGN (Feng et al., 2020)	42.03	18.07	39.56
TGDGA (Zhao et al., 2020)	43.11	19.15	40.49
DialoGPT (Zhang et al., 2019d)	39.77	16.58	38.42
UniLM (Dong et al., 2019)	47.85	24.23	46.67
PEGASUS (Zhang et al., 2019a)	50.50	27.23	49.32
BART-xsum (Lewis et al., 2019)	51.74	26.46	48.72
BART-xsum + Sketch (Ours)	51.79	26.85	49.15
BART-xsum + Ctrl (Ours)	**52.84**	27.35	50.29
CODS (Ours)	52.65	**27.84**	**50.79**

Table 1: Dialogue summarization ROUGE evaluation on the SAMSum test set. Results with * are obtained from Gliwa et al., 2019. CODS achieves the highest ROUGE score. *BART-xsum + Sketch* and *BART-xsum + Ctrl* are ablated models individually removing controllability and sketch generation component from CODS.

	ROUGE_WE	BERTScore	MoverScore	BLEU	CIDEr	SMS
PEGASUS	0.3562	0.5335	0.3233	17.33	1.741	0.1608
BART-xsum	0.3606	0.5387	0.3391	17.55	1.701	0.1401
CODS	**0.3759**	**0.5458**	**0.3539**	**19.58**	**1.981**	**0.1689**

Table 2: Dialogue summarization evaluation on the SAMSum test set with additional recently introduced metrics that have been applied to both text generation and summarization.

3 Experiments

3.1 Dataset

We perform experiments on the recently released SAMSum dataset (Gliwa et al., 2019) [2], which is the most comprehensive resource for abstractive dialogue summarization tasks. It contains 16K natural messenger-like dialogues created by linguists fluent in English with manually annotated summaries. This dataset is more challenging than the previous corpus (McCowan et al., 2005) in the following aspects: 1) Unlike previous datasets consisting of only hundreds of dialogue-summary pairs, it has larger data size (16369 samples); 2) 75% of the conversations are between two interlocutors, the rest are between three or more people; 3) the conversations cover diverse real-life topics, and the summaries are annotated with information about the speakers. We preprocess the data by the following steps: 1) concatenate adjacent utterances of the same speaker into one utterance; 2) clean the dialogue text by removing hashtags, URLs and Emojis; 3) label each utterance with its corresponding interrogative pronoun category with a weak supervision approach (Ratner et al., 2019); 4) parse each utterance with a constituency parser and find the longest common sub-sequence between the phrases and summary to be the key phrases.

3.2 Evaluation Metrics and Baselines

We use the standard ROUGE metric (Lin, 2004) as automatic evaluation metrics, including ROUGE-1, ROUGE-2, and ROUGE-L. Following previous work (Gliwa et al., 2019), we use py-ROUGE[3] library with stemming. We compare our model with baselines reported in Gliwa et al., 2019: Longest-3 is a commonly-used extractive summarization baseline which takes the top three longest sentences as summary. The pointer generator and Fast abs are RNN-based methods with copy-attention mechanism or policy gradient. The Transformer is a random-initialized self-attention architecture with multi-head attention. The DynamicConv is a

[2] The conversations in SAMSum may contain offensive words, please use the dataset carefully.

[3] pypi.org/project/pyROUGE/

lightweight convolutional model that can perform competitively to self-attention. All of these models are not pre-trained.

Besides, we investigate four pre-trained generative language models to see which works the best for the dialogue summarization task. DialoGPT is a GPT model pre-trained on open-domain Reddit data. UniLM is pre-trained using three types of language modeling tasks: unidirectional, bidirectional, and sequence-to-sequence prediction on English Wikipedia and BookCorpus. PEGASUS masks important sentences from input and is trained to generate the missing parts, similar to an extractive summary approach. BART is trained by corrupting text with an arbitrary noising function and learning to reconstruct the original text. We use default parameters listed in the respective open-source repositories to fine-tune on the dialogue summarization task. We show the training details in the Appendix.

3.3 Results

In Table 1 of ROUGE results, we find that the methods that are pre-trained or with pre-trained embeddings perform better than those that are not. For instance, DynamicConv achieves a 3 – 4% improvement by adding GPT-2 embeddings. This further confirms the impact of language model pre-training on downstream tasks. Among the pre-trained generative language models examined, PEGASUS and BART are the two top performance models with ROUGE-1 higher than 50. DialoGPT, the model pre-trained on conversational data, does not achieve satisfactory results, implying that Reddit data has limited knowledge to be transferred to dialogue summarization tasks. CODS achieves the highest ROUGE score compared with other models, notably 50.79% ROUGE-L.

To understand the individual contribution of each component in our model, we also conduct an ablation study by removing summary sketch generation (BART+Ctrl) or controllability (BART+Sketch). In both cases we observe a performance drop, except a slight improvement on ROUGE-1 for BART+Ctrl. This suggests that the sketching step helps generate a more fluent summary even with lower unigram matching. Furthermore, recognizing the limitation of ROUGE scores in their ability to fully capture the resemblance between the generated summary and the reference, in Table 2, we follow (Fabbri et al., 2020) to compare model performances with additional metrics, including ROUGE-Word Embedding (Ng and Abrecht, 2015), BERTScore (Zhang et al., 2019b), MoverScore (Zhao et al., 2019), Sentence Mover's Similarity (SMS) (Clark et al., 2019), BLEU (Papineni et al., 2002), and CIDEr (Vedantam et al., 2015). As shown in Table 2, CODS consistently outperforms PEGASUS and BART. More information about these evaluation metrics are shown in the Appendix.

	Length Ratio	Consistent	Informative
Longest-1	0.27	**0.70**	0.23
BART-xsum-1	**0.16**	0.50	0.16
CODS-1	0.19	0.50	**0.49**
BART-xsum	0.26	0.65	0.51
CODS	**0.24**	**0.69**	**0.53**
Gold	0.27	0.74	0.55

Table 3: Human evaluation results on test set for both controllable summary and standard summary.

3.4 Analysis

3.4.1 Human Evaluation by Crowdsourcing

We leverage human judgement to evaluate the generated summaries via crowdsourcing, especially for granularity-controlled generation, since we do not have human-written reference summaries of various lengths (number of sentences). We ask workers to rate the summaries in two aspects on a scale from -1 (worst) to 1 (best): factual consistency and informativeness. *Factual consistency* acts as a precision measure, assessing whether the information provided in summary contains factual errors which are against the source dialogue; *Informativeness* is a recall-oriented measure, examining whether critical information in a dialogue is mentioned in summary. We also show the length ratio between a summary and a dialogue, where a lower ratio means a higher compression rate. For the crowdsourcing evaluation, we randomly select 6% dialogues from the test set, each of which is annotated by three workers. More details about human evaluation process are in the Appendix [4].

To show the proposed controllable generation's strengthens and quality, we provide two additional baselines, Longest-1 and BART-1. The longest-1 method is an extractive baseline that outputs the longest dialogue turn as the final summary. The BART-1 is a strong abstractive baseline where we train a BART-based summarization model with the

[4] The prediction file on the test set is provided in the supplementary file.

Kelly: I still haven't received the rent money. Did you check with your bank? John: Yes. I definitely sent it last week. Kelly: But I still don't have it. Can you please check that you sent it to the right account. John: Ok. Give me 5 min. Kelly: OK John: I checked and the money did go out of my account last week. Kelly: What account number did you send it to? John: 44-1278 Kelly: No wonder! My account number is 44-1279. You sent it to someone else's account. John: ...! I'm really sorry! Kelly: I still need the rent money though. John: I'm really sorry I'll have to go to the bank tomorrow and ask if they can re-send it to the right account. Kelly: Thanks !	
Longest-1	John said I'm really sorry I'll have to go to the bank tomorrow and ask if they can re-send it to the right account.
BART-1	Kelly still hasn't received the rent money from John.
CODS-1	John sent the rent money to the wrong account and will have to ask the bank to re-send it to the correct one tomorrow.
BART	Kelly still hasn't received the rent money. John sent it to the wrong account number 44-1278. John will go to the bank tomorrow and ask if they can re-send the money to the right account.
CODS	**Sketch**: 1 #confirm haven't received the rent money check with your bank 2 none 3 #confirm check that you sent it to the right account 4 none 5 none 6 #abstain the money did go out of my account last week 7 #abstain did you send it to 8 none 9 #what sent it to someone else's account 10 none 11 #abstain need the rent money though 12 #abstain 'm really sorry i'll have to go to the bank tomorrow and ask if they can re-send it to the right account 13 none **Summary**: John sent the rent money to the wrong account last week. John will go to the bank tomorrow and ask if he can re-send the money to the correct account.
Gold	Kelly hasn't received the rent money, because John sent it to the wrong bank account. He will go to the bank to tackle the issue.

Table 4: A test set example with generated summaries.

	Reference Summary	CODS Summary
Associate names with actions	Lilly will be late. Gabriel will order pasta with salmon and basil for her.	Lilly will be late for the meeting with Gabriel. Gabriel will order something for Lilly.
	Ann doesn't know what she should give to her dad as a birthday gift. He's turning 50. Fiona tries to help her and suggests a paintball match.	It's Ann's dad's 50th birthday. He's turning 50. Ann and Fiona are planning a surprise birthday party for her dad.
Extract information after the discussion	Paul will buy red roses following Cindy's advice.	Paul wants to buy red roses.
Decide important information	Rachel's aunt had an accident and she's in hospital now. She's only bruised. The perpetrator of the accident is going to pay for the rehabilitation.	Rachel is at the hospital with her aunt, who had an accident. She's bruised but fine. She will give her a hug.
	Hannah needs Betty's number but Amanda doesn't have it. She needs to contact Larry.	Amanda can't find Betty's number. Amanda suggests to text him.

Table 5: Case analyses by manually examining CODS generated summaries.

number of summary sentences in the training set as its start-of-sentence token during decoding. Similar to the approach from Liu et al., 2018, we can use different start-of-sentence tokens to control the BART output.

In general, it is preferable to have a factually consistent and informative summary that is succinct (low length ratio, high compression rate) at the same time. As shown in the first row of Table 3, CODS-1 achieves the highest informative score among all generated one-sentence summaries, indicating the strength of the proposed controllable method in producing succinct yet informative dialogue summaries. The Longest-1 method has a higher consistent score because its summary is directly copied from the original dialogue, preventing any factual mistakes. The second row of Table 3 shows that CODS, when automatically determining the granularity of the summary, produces summaries that are more succinct (lower length ratio), more factually consistent, and more informative, compared to the BART model.

3.4.2 Case Study

CODS outperforms the baseline models in both ROUGE scores and human evaluation metrics. We now further inspect its textual quality. In Table 4, we show an example from the SAMSum test set with summaries generated by different models. In this example, CODS and CODS-1 can both produce a near-perfect summary even compared to the human-written reference summary. On the other hand, the summary generated by BART includes overly detailed information (e.g., bank account). We show some more examples in the Appendix and all the predictions (including CODS-1 and CODS-2) in the supplementary file.

We also manually examine 100 summaries generated from CODS against the reference summaries in the test set. Specifically, we analyze each of the three following problematic cases, where summarization models frequently make mistakes, reported by Gliwa et al., 2019, and provide sample summaries in Table 5. 1) *Associating names with actions*: CODS performs well in dealing with speakers' names. It accurately associates "her dad" with "Ann's dad," also "Fiona tries to help her" with "Ann and Fiona." 2) *Extract information about the arrangement after discussion*: Even speakers hesitate about the flower's color to be yellow, pink or red in the middle of the discussion, CODS still correctly determines the right color after several turns. 3) *Decide important information in dialogues*: CODS fails to capture some of the important facts (marked as red) mentioned in reference summary. We conjecture the reason could be that 1) some of the important facts are located in the same part of the highlighted turns, and 2) those information is missed by the key phrase extraction. Simultaneously, we force the model to generate only the most important one under the constraint of controllability. The improvement of CODS on the first two summarization difficulties can be partially attributed to the clear logic in the sketch when input to the model.

4 Related Work

Neural Text Summarization There are two main paradigms for text summarization: extractive and abstractive. Inspired by the success of applying seq2seq models on neural machine translation, Rush et al., 2015 and Nallapati et al., 2016 introduce the neural seq2seq model on abstractive text summarization, with an attention-based encoder and a neural language model decoder. To solve the problem of out-of-vocabulary words and to capture salient information in source documents, See et al., 2017 propose a pointer-generator network that copy words from source to target. Many subsequent works (Gehrmann et al., 2018; Paulus et al., 2018) further demonstrate its effectiveness with reinforcement learning. Recently, Liu and Lapata, 2019 apply BERT on text summarization and propose a general framework for both extractive and abstractive models. Zhang et al., 2019c pre-train hierarchical document encoder for extractive summarization. Lewis et al., 2019 introduces BART, a denoising autoencoder for pretraining sequence-to-sequence models. BART significantly outperforms the best previous work in terms of ROUGE metrics.

Dialogue Summarization Regarding to the datasets in dialogue summarization, initial abstractive dialogue summarization work (Oya et al., 2014; Mehdad et al., 2014; Banerjee et al., 2015) are conducted on the AMI meeting corpus (McCowan et al., 2005), with only 141 summaries. Goo and Chen, 2018 propose to use the topic descriptions (high-level goals of meetings) in AMI as reference summaries and use dialogue acts as training signals. Pan et al., 2018 build the Dial2Desc dataset by reversing a visual dialogue task, aligning image dialogues with the image caption as a summary. Liu et al., 2019 collect their dataset from the logs in the DiDi customer service center. It is restricted to task-oriented scenario, where one speaker is the user and the other is the customer agent, with limited topics and it is also connected to the goal of dialogue state tracking task (Wu et al., 2019a, 2020b). Recently, Gliwa et al., 2019 introduce the SAMSum corpus, with 16k chat dialogues with manually annotated summaries. It is the first comprehensive abstractive dialogue summarization dataset spanning over various lengths and topics. Chen and Yang, 2020 propose a multi-view sequence-to-sequence model by extracting different views of structures from conversations. Both their method and ours leverage rich conversation structure information. Evaluating on SAMSum, our model CODS outperform theirs by 3 points in terms of ROUGE scores, indicating our utilized dialogue features are more effective.

Length-controllable Generation The most prevalent method for length control generation is using a special length embedding. Kikuchi et al., 2016 first propose length control for abstractive summarization by using length embedding as an additional input for the LSTM. Fan et al., 2018 train embeddings that correspond to each different output length and prepend that length marker at the beginning of the decoder. Liu et al., 2018 incorporates the length embedding into initial state of a CNN-based decoder. Takase and Okazaki, 2019 extends the positional encoding in

Transformer model by considering the remaining length explicitly at each decoding step. Saito et al., 2020 propose to control the summary length with prototype extractor. However, the retrieve-and-rewrite process is restricted by the extraction quality, leaving its performance limited by extractive solutions' capabilities. The aforementioned works all focus on structured text summarization (e.g. news document). We are the first to propose generate length-controllable summary on dialogues by highlighting arbitrary numbers of dialogue spans.

5 Conclusion

The dialogue summarization task is challenging but with vast application potential. We propose CODS, a state-of-the-art dialogue summarization model with granularity controllability. CODS uses a weakly-labeled summary sketch for its two-stage generation, and text-span conditional generation for a controllable summary. Our model surpasses existing models on the largest dialogue summarization dataset. We show with human evaluation that our model can generate factually consistent and informative summaries. We also point out several error cases to shed light on future research direction of controllable dialogue summarization.

References

Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Abstractive meeting summarization using dependency graph fusion. In *Proceedings of the 24th International Conference on World Wide Web*, pages 5–6.

Jiaao Chen and Diyi Yang. 2020. Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4106–4118.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. *arXiv preprint arXiv:1805.11080*.

Elizabeth Clark, Asli Celikyilmaz, and Noah A Smith. 2019. Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2748–2760.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13063–13075.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2020. Summeval: Re-evaluating summarization evaluation. *arXiv preprint arXiv:2007.12626*.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54.

Xiachong Feng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2020. Incorporating commonsense knowledge into abstractive dialogue summarization via heterogeneous graph networks. *arXiv preprint arXiv:2010.10044*.

Sebastian Gehrmann, Yuntian Deng, and Alexander M Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. *arXiv preprint arXiv:1911.12237*.

Chih-Wen Goo and Yun-Nung Chen. 2018. Abstractive dialogue summarization with sentence-gated modeling optimized by dialogue acts. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 735–742. IEEE.

Geoff Hart. The five w's of online help systems, year = 2002, url = http://www.geoff-hart.com/articles/2002/fivew.htm, urldate = 2002.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. *arXiv preprint arXiv:1805.01052*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract meaning representation for multi-document summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1178–1190.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Chunyi Liu, Peng Wang, Jiang Xu, Zang Li, and Jieping Ye. 2019. Automatic dialogue summary generation for customer service. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1957–1965.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3721–3731.

Yizhu Liu, Zhiyi Luo, and Kenny Zhu. 2018. Controlling length in abstractive summarization using a convolutional neural network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4110–4119.

I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, W. Post, Dennis Reidsma, and P. Wellner. 2005. The ami meeting corpus. In *Proceedings of Measuring Behavior 2005, 5th International Conference on Methods and Techniques in Behavioral Research*, pages 137–140. Noldus Information Technology.

Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. 2014. Abstractive summarization of spoken and written conversations based on phrasal queries. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1220–1230.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Jun-Ping Ng and Viktoria Abrecht. 2015. Better summarization evaluation with word embeddings for ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1925–1930, Lisbon, Portugal. Association for Computational Linguistics.

Nikola I Nikolov, Michael Pfeiffer, and Richard HR Hahnloser. 2018. Data-driven summarization of scientific articles. *arXiv preprint arXiv:1804.08875*.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. 2014. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 45–53.

Haojie Pan, Junpei Zhou, Zhou Zhao, Yan Liu, Deng Cai, and Min Yang. 2018. Dial2desc: end-to-end dialogue description generation. *arXiv preprint arXiv:1811.00185*.

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.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Vahed Qazvinian and Dragomir R Radev. 2008. Scientific paper summarization using citation summary networks. *arXiv preprint arXiv:0807.1560*.

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

Alexander Ratner, Braden Hancock, Jared Dunnmon, Frederic Sala, Shreyash Pandey, and Christopher Ré. 2019. Training complex models with multi-task weak supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4763–4771.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Itsumi Saito, Kyosuke Nishida, Kosuke Nishida, Atsushi Otsuka, Hisako Asano, Junji Tomita, Hiroyuki Shindo, and Yuji Matsumoto. 2020. Length-controllable abstractive summarization by guiding with summary prototype. *arXiv preprint arXiv:2001.07331*.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Sho Takase and Naoaki Okazaki. 2019. Positional encoding to control output sequence length. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3999–4004.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019a. Transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.

Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020a. Tod-bert: Pre-trained natural language understanding for task-oriented dialogues. *arXiv preprint arXiv:2004.06871*.

Chien-Sheng Wu, Steven C.H. Hoi, and Caiming Xiong. 2020b. Improving limited labeled dialogue state tracking with self-supervision. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4462–4472, Online. Association for Computational Linguistics.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019a. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819, Florence, Italy. Association for Computational Linguistics.

Chien-Sheng Wu and Caiming Xiong. 2020. Probing task-oriented dialogue representation from language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5036–5051, Online. Association for Computational Linguistics.

Felix Wu, Angela Fan, Alexei Baevski, Yann N Dauphin, and Michael Auli. 2019b. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*.

Jiacheng Xu and Greg Durrett. 2019. Neural extractive text summarization with syntactic compression. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3283–3294.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763.

Klaus Zechner. 2001. Automatic generation of concise summaries of spoken dialogues in unrestricted domains. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 199–207.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J Liu. 2019a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *arXiv preprint arXiv:1912.08777*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019b. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. *arXiv preprint arXiv:1808.07187*.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019c. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019d. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

Lulu Zhao, Weiran Xu, and Jun Guo. 2020. Improving abstractive dialogue summarization with graph structures and topic words. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 437–449, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*.

A Appendix

A.1

We define dialogue intent categories as follows: *WHY:* asks the reason of the state mentioned in the previous turn, e.g., "why" or "why not"; *WHAT:* requests more details about the aforementioned object, the sentence usually starts with "what's" or "what about"; *WHERE:* the location of the event; *WHEN:* the time of the event, e.g. ,"when" or "what time"; *CONFIRM:* expects the other speaker to establish the correctness of a certain case, the sentence usually starts with patterns like "are you", "will you" or "has he"; *ABSTAIN:* the utterance does not belong to any of the previous categories. It occurs when speakers continue to state or comment without seeking for more information from the others.

A.2

- DialoGPT: A GPT model pretrained on 147M conversation-like data extracted from Reddit comments. We use the model with 117M parameters. `github.com/microsoft/DialoGPT`
- UniLM: A multi-layer Transformer network optimized for three language modeling objectives: unidirectional, bidirectional and sequence-to-sequence prediction. It is initialized with BERT$_{\text{LARGE}}$, then pre-trained using English Wikipedia and BookCorpus. Same as BERT$_{\text{LARGE}}$, it contains 340M parameters. `github.com/microsoft/unilm`
- PEGASUS: They pretrain a Transformer-based encoder-decoder models with a new self-supervised objective - gap-sentence generation - on the C4 corpus. We use the PEGASUS of 568M parameters. `github.com/google-research/pegasus`
- BART: Transformer-based encoder-decoder model trained by corrupting text with an arbitrary noising function and learning a model to reconstruct the original text. We use BART$_{\text{LARGE}}$ model which contains 400M parameters. `huggingface.co/transformers/model_doc/bart.html`
- CODS: It's based on BART$_{\text{LARGE}}$ model which contains 400M parameters.

<table>
<tr><td colspan="2">Keith: Meg, pls buy some milk and cereals, I see now we've run out of them .
Megan: hm, sure, I can do that .
Megan: but did you check in the drawer next to the fridge ?
Keith: nope, let me have a look .
Keith: ok, false alarm, we have cereal and milk .
Deana: glad to hear it !</td></tr>
<tr><td>Summary</td><td>Megan needn't buy milk and cereals. They're in the drawer next to the fridge.</td></tr>
</table>

Table 6: Example key phrases in summary sketch.

<table>
<tr><td colspan="2">Norbert: we need to hurry to catch the tour .
Wendy: ok , am buying something . be right out !
Norbert: ok . am not waiting long though . missed the last one because of you .
Wendy: just be patient for once .
Norbert: im always patient .
Wendy: at the register now .
Norbert: alright .</td></tr>
<tr><td>Summary</td><td>Wendy is shopping, but she needs to hurry up to catch the tour.</td></tr>
</table>

Table 7: Example key phrases in summary sketch.

A.3 Sketch Construction

Previous methods (Goo and Chen, 2018; Pan et al., 2018) heavily rely on explicit intent annotations in datasets. We label user intent automatically for each utterance with the Snorkel library in a weak supervision approach. For each interrogative pronoun category, we first manually identify its most frequent key words and patterns (can be found in our source code). Then we use the labeling functions in Snorkel to label all the utterances.

For the utterance compression, we do LCS on the phrases generated from the constituency parser. In the example of 1, *s just one of many boring days at work* the parsed constituent overlapping with 'at work' in the summary, so we keep this phrase. However, in other examples, not all overlapped words are meaningful (e.g. stop words). We thus filter the LCS results and only keep important key phrases. Then we train our model to predict these key phrase spans in each turn. We show three examples of our generated key phrases in summary sketches on evaluation set (see Table 6, 7, 8)

A.4 Training Details

We use huggingface (Wolf et al., 2019a) implementation to fine-tune a BART model. We use the large

<table>
<tr><td colspan="2">Phil: is brandon in ?
Clara: not yet .
Phil: has he called to say he'd be late ?
Clara: no , he hasn't .
Phil: it's not the first time , ist it ?
Clara: no , it isn't .
Phil: when he arrives , tell him to come to me .
Clara: no , it isn't .
Phil: please prepare a report on the absenteeism and lateness . i expect it by friday on my desk .
Clara: it will be ready .</td></tr>
<tr><td>Summary</td><td>Brandon is late again. Clara will prepare a report on the absenteeism and lateness for Phil by Friday.</td></tr>
</table>

Table 8: Example key phrases in summary sketch.

version fine-tuned on the XSUM (Narayan et al., 2018) dataset with 12 self-attention encoder and decoder layers. We truncate input dialogue to a maximal length 512 with training batch size 4. We train the model with Adam optimizer (Kingma and Ba, 2014) with 0.1 proportion for linear learning rate warmup. We early stop on validation set ROUGE-1 score, and it is trained for around 40,000 steps on one NVIDIA V100 GPU. During inference, we do beam search decoding with beam size 4.

A.5 Evaluation Metrics

Information obtains from (Fabbri et al., 2020):

- ROUGE measures the number of overlapping textual units between the generated summary and a set of reference summaries.
- ROUGE-WE extends ROUGE by taking cosine similarity of Word2Vec embeddings into account.
- BERTScore computes similarity scores by aligning generated and reference summaries on a token-level based on the output of the BERT-based model. Token alignments are computed greedily with the objective of maximizing the cosine similarity between contextualized token embeddings. We report the F1 score.
- MoverScore measures semantic distance between a summary and reference text by making use of the Word Mover's Distance operating over n-gram embeddings pooled from BERT representations.
- Sentence Mover's Similarity (SMS) extends Word Mover's Distance to view documents as a bag of sentence embeddings as well as a variation which represents documents as both a bag of sentences and a bag of words.
- BLEU is a corpus-level precision-focused metric which calculates n-gram overlap between a candidate and reference utterance and includes a brevity penalty. It is the primary evaluation metric for machine translation.
- CIDEr computes 1-4-gram co-occurrences between the candidate and reference texts, downweighting common n-grams and calculating cosine similarity between the ngrams of the candidate and reference texts.

A.6 Human Evaluation

We use roughly 6% of the test set data in SAMSum for human evaluation and we do some filtering based on the annotation of the "gold summary". Specifically, we filter those annotations if a "gold summary" has been annotated as "-1" (the meaning of each score is shown below), implying that the annotators may not pay attention to the scoring. The final results reported in Table 3 is the mean from three different annotators.

The "gold summary" is actually not perfect and it might contain some noisy annotation, this is the reason why some workers may give 0 even if it is collected from humans. Below is the scoring instruction we sent to our workers:

- Factual Consistency (Precision): The rating measures whether the information provided in a summary is correct. Score -1 if a summary contains a serious factual error. Score 0 if a summary has some minor factual errors. Score 1 if everything in a summary is factually correct.
- Informative (Recall): The rating measures whether all the important information in a dialogue is included in a summary. Score -1 if a summary misses serious key points. Score 0 if a summary misses a few key points. Score 1 if a summary covers all key points.

Paul: what color flowers should i get Cindy: any just not yellow Paul: ok , pink ? Cindy: no maybe red Paul: just tell me what color and what type ok ? Cindy: ugh , red roses !	
Gold	Paul will buy red roses following Cindy's advice.
BART	Paul wants to get red roses. Cindy doesn't want pink or yellow roses.
CODS	Paul wants to buy red roses.

Table 9: Dialogue for the "Extract information after the discussion" sample in Table 5

Phil: good evening deana ! many thanks for this nice card from you . constantine was very happy !. are these sunglasses also from you ? Deana: i thought they belonged your cathreen ! Phil: nope . she says they aren't hers . Deana: mine either . look , maybe you feel like keeping them ?. i seem to have so many sunglasses .. 8 Phil: where did you find them , possible that they belong to adrian ? Deana: in this empty place above the radio . in the very back .. if adrian wants it , no pro !. exactly ! Phil: ok , they don't belong to any of us , and nobody else drove your car . but we can look after these sunglasses . Deana: glad to hear it !	
Longest-1	Phil said good evening deana ! many thanks for this nice card from you . constantine was very happy !. are these sunglasses also from you ?
BART-1	Phil and Deana will look after Adrian's sunglasses.
CODS-1	Deana found Adrian's sunglasses in the back of Phil's car.
BART	Phil and Deana are going to look after Adrian's sunglasses.
CODS	Phil got a card from Deana. Deana found them in the empty place above the radio. Deana has a lot of them.
Gold	Phil received a card from Deana. Constantine was happy. Phil has sunglasses, that Deana found in the back above the radio. Deana and Phil don't know who they belong too. Phil will keep the sunglasses.

Table 10: Test set example for qualitative study.

Celia: where do you want to go for holiday ? Mike: i was thinking about egypt Celia: too hot . what about croatia ? Mike: good idea , i've never been there	
Longest-1	Celia said where do you want to go for holiday ?
BART-1	Mike wants to go for holiday to Egypt.
CODS-1	Mike wants to go on holiday to Egypt or Croatia.
BART	Celia and Mike will go for holiday to Croatia.
CODS	Mike wants to go on holiday to Egypt. Celia thinks it's too hot. Mike has never been to Croatia, but he likes the idea.
Gold	Mike considers going to Egypt for holiday. It's too hot for Celia, she suggests Croatia instead. Mark likes the idea, he's never been there.

Table 11: Test set example for qualitative study.

Diane: how long do you have to work tonight ? Ross: about 2 hours , why ? Diane: i just wanted to do something maybe Ross: i think i'll be worn out after all hat work , baby Diane: we can just chill at home , don't worry. i just wanted to prepare Ross: ok. then just to be safe let's say it will take me 3 hours Diane: but you just said 2 ! Ross: ... , Diane , don't start again Diane: what am i starting !. you're impossible Ross: can't you understand that this is important to me !. my career depends on it ! Diane: well , if your career is the most important thing in the world then i wouldn't want to disturb !	
Longest-1	Diane said well , if your career is the most important thing in the world then i wouldn't want to disturb !
BART-1	Ross has to work for 2 hours tonight.
CODS-1	Ross has to work 3 hours tonight.
BART	Ross has to work tonight for 2 hours. Ross and Diane will chill at home.
CODS	Ross has to work 3 hours tonight.
Gold	Diane is not happy with Ross prioritising work over spending time with her.

Table 12: Test set example for qualitative study.