Controllable Neural Dialogue Summarization with Personal Named Entity Planning

Zhengyuan Liu, Nancy F. Chen
Institute for Infocomm Research, A*STAR, Singapore
{liu_zhengyuan,nfychen}@i2r.a-star.edu.sg

Abstract

In this paper, we propose a controllable neural generation framework that can flexibly guide dialogue summarization with personal named entity planning. The conditional sequences are modulated to decide what types of information or what perspective to focus on when forming summaries to tackle the under-constrained problem in summarization tasks. This framework supports two types of use cases: (1) Comprehensive Perspective, which is a general-purpose case with no user-preference specified, considering summary points from all conversational interlocutors and all mentioned persons; (2) Focus Perspective, positioning the summary based on a user-specified personal named entity, which could be one of the interlocutors or one of the persons mentioned in the conversation. During training, we exploit occurrence planning of personal named entities and coreference information to improve temporal coherence and to minimize hallucination in neural generation. Experimental results show that our proposed framework generates fluent and factually consistent summaries under various planning controls using both objective metrics and human evaluations.

1 Introduction

Automatic summarization is the task of compressing a lengthy piece of text to a more concise version while preserving the information of the source content. Extractive approaches select and concatenate salient words, phrases, and sentences from the source to form the summary (Lin and Bilmes, 2011; Kedzie et al., 2018; Liu et al., 2020). On the other hand, abstractive approaches generate the summary either from scratch or by paraphrasing important parts of the original text (Jing and McKeown, 2000; Gehrmann et al., 2018). For abstractive summarization to be practically usable, it would require more in-depth comprehension, better generalization, reasoning, and incorporation of

Figure 1: Dialogue summary examples generated by personal named entity planning: some examples focus on perspectives from distinct personal named entities (e.g., John, Tony); comprehensive planning includes all personal named entities in the dialogue. Note that the content of the ground-truth summary depends on which personal named entity's perspective the focus is during summary formation.

real-world knowledge (Hovy et al., 1999; See et al., 2017). While extractive models could suffice for document summarization, abstractive approaches are essential for dialogue summarization to be more easily accessible to users.

Most benchmarked summarization datasets focus on the news domain, such as NYT (Sandhaus, 2008) and CNN/Daily Mail (Hermann et al., 2015) as material for large-scale corpus construction is readily available online. Neural approaches have achieved favorable improvements in both extractive and abstractive paradigms (Paulus et al., 2017; Liu and Lapata, 2019). Neural dialogue summarization is an emerging research area (e.g., Goo and Chen (2018), Liu et al. (2019)). While the

available data collections are much smaller than those for documents (Carletta et al., 2005; Gliwa et al., 2019), neural models have shown potential to generate fluent sentences via fine-tuning on large scale contextualized language models (Chen and Yang, 2020; Feng et al., 2021). Unfortunately, most summary generation tasks are constructed in an under-constrained fashion (Kryscinski et al., 2019): in their corpus construction process, only one reference summary is annotated. Models trained via supervised learning on such datasets provide general-purpose summaries, but are suboptimal for certain applications and use cases (Fan et al., 2018; Goodwin et al., 2020). For instance, as shown in Figure 1, a human can write summaries from John or Tony's perspective. However, a neural model with a general summarizing purpose may overlook information that is important to a specific person's perspective. On the other hand, if someone wants to collect as much information from the source content, the summary should be written in a comprehensive manner, taking into consideration all personal named entities. Such needs are not met with models providing only one possible output.

Furthermore, different from passages, human-to-human conversations are a dynamic and interactive flow of information exchange (Sacks et al., 1978), which are often informal, verbose, and repetitive. Since important information is scattered across speakers and dialogue turns, and is often embodied in incomplete sentences. Therefore, generating a fluent summary by utterance extraction is impractical, thus requiring models capable of generating abstractive summaries. However, neural abstractive models often suffer from hallucinations that affect their reliability (Zhao et al., 2020), involving improper gendered pronouns and misassigned speaker associations (Chen and Yang, 2020). For example, as shown in Figure 2, the model makes an incorrect description that *"she texted Larry last time at the park"* (in red). While this sentence achieves a high score in word-overlapping metrics, the semantic meaning it conveys is incorrect: in the context of the generated summary, *she* refers to *Amanda*, yet in reality it is *Larry* that called (not texted) *Betty*. Such factual inconsistency, the inability to adhere to facts from the source, is a prevalent and unsolved problem in neural text generation.

In this work, we introduce a controllable dialogue summarization framework. As the aim of dialogue summaries often focuses on *"who did what"*

Figure 2: One dialogue summarization example: each coreference chain is highlighted with the same color. The generated sentence in red is factually incorrect.

and the narrative flow usually starts with a subject (often persons), we propose to modulate the generation process with personal named entity plannings. More specifically, as shown in Figure 1, a set of personal named entities[1] (in color) are extracted from the source dialogue, and used in a generation model as a conditional signal. We postulate that such conditional anchoring enables the model to support flexible generation. It could be especially useful to address certain demands such as targeting specific client needs for customizing marketing strategies or drilling down customer dissatisfaction at call centers to educate customer agents. In addition, to improve the quality of conditional generation outputs, we integrate coreference resolution information into the contextual representation by a graph-based neural component to further reduce incorrect reasoning (Liu et al., 2021).

We conduct extensive experiments on the representative dialogue summarization corpus SAMSum (Gliwa et al., 2019), which consists of multi-turn dialogues and human-written summaries. Empirical results show that our model can achieve state-of-the-art performance, and is able to generate fluent and accurate summaries with different personal named entity plans. Moreover, factual correctness assessment also shows that the output from our model obtains quality improvement on both automatic measures and human evaluation.

---

[1] A complete named entity set includes personal names, locations, organizations, time expressions, etc.

Output Summary: Jason said Tom was staying at home.

Language Decoder: Transformer Decoding Layers

Fusing Coreference Layer

Fusing Coreference Information

Language Encoder: Transformer Encoding Layers

Position Embedding

Token Embedding: { Jason Tom } </s> Bob : where is Tom ... Jason : he is at home ... Bob

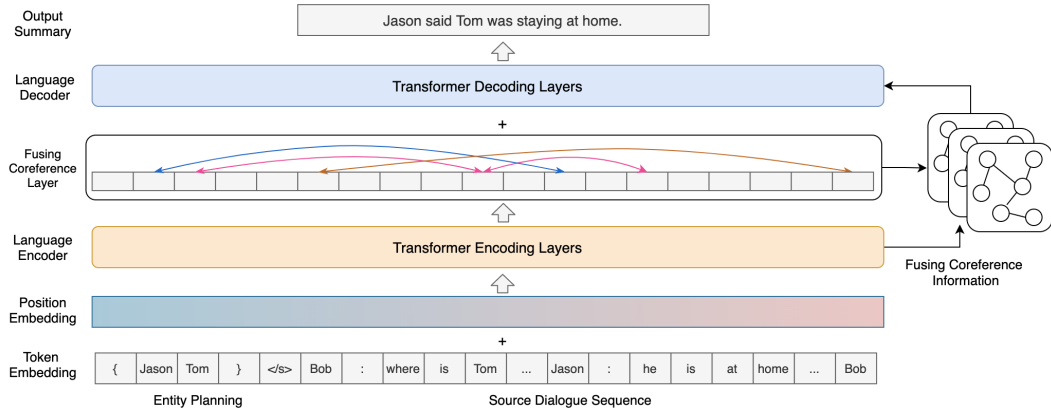Entity Planning   Source Dialogue Sequence

Figure 3: Overview of the proposed conditional generation framework with entity planning and coreference integration. Colored lines with arrows in the Fusing Coreference Layer denote the coreference links.

2 Related Work

Text summarization has received extensive research attention, and is mainly studied in abstractive and extractive paradigms (Gehrmann et al., 2018). For extractive summarization, non-neural approaches study various linguistic and statistical features via lexical (Kupiec et al., 1995) and graph-based modeling (Erkan and Radev, 2004). Much progress has been made by recent neural approaches (Nallapati et al., 2017; Kedzie et al., 2018). Compared with extractive methods, abstractive approaches are expected to generate more concise and fluent summaries. While it is a challenging task, with large-scale datasets (Hermann et al., 2015) and sophisticated neural architectures, the performance of abstractive models have achieved substantial improvements in the news domain: sequence-to-sequence models are first introduced by Rush et al. (2015) for abstractive summarization; pointer-generator network (See et al., 2017) elegantly handled out-of-vocabulary issues by copying words directly from the source content; Gehrmann et al. (2018) combines the two paradigms by integrating sentence rewriting into content selection; large-scale pre-trained language models also bring further improvement on summarization performance (Liu and Lapata, 2019; Lewis et al., 2020).

Recently, neural summarization for conversations has become an emerging research area. Corpora are constructed from meetings (Carletta et al., 2005) or daily chats (Gliwa et al., 2019). Based on the characteristics of the dialogues, many studies pay attention to utilizing conversational analysis for dialogue summarization, such as leveraging dialogue acts (Goo and Chen, 2018), multi-modal features (Li et al., 2019), topic information (Liu et al., 2019), and fine-grained view segmentation with hierarchical modeling (Chen and Yang, 2020).

Controllable text generation introduces auxiliary signals to obtain diverse or task-specific outputs. It has been studied in various domains such as style transferring (Shen et al., 2017) and paraphrasing (Iyyer et al., 2018). The conditional input can be in the form of pre-defined categorical labels (Hu et al., 2017), latent representations, semantic or syntactic exemplars (Gupta et al., 2020), and keyword planning (Hua and Wang, 2020). Recently, He et al. (2020) and Dou et al. (2021) proposed two generic frameworks for length-controllable and question/entity-guided document summarization, and we proposed personal named entity planning upon the characteristics of dialogue summarization.

3 Controllable Generation with Personal Named Entity Planning

In this section, we introduce the proposed conditional generation framework, elaborate on how we construct personal named entity planning, and delineate the steps for training and generation.

3.1 Task Definition

Controllable dialogue summarization with personal named entity planning is defined as follows:
Input: The input consists of two entries: (1) the source content D, which is a multi-turn dialogue; (2) a customized conditional sequence C, which is the proposed personal named entity planning.
Output: The output is a natural language sequence Y, which represents the summarized information from the source content D with the pre-defined personal named entity plan C. Given one instance of D, Y can be manifested as various summaries

conditioned on different choices of C. The output summaries are expected to be fluent and factually correct, covering the indicated entities in the specified conditional signal C.

3.2 Personal Named Entity Planning

Personal named entities are used to form a planning sequence. A customized plan represents what the summary includes, covering specific personal named entities that appear in the dialogue. These named entities **are not limited to** the speaker roles, but include all persons mentioned in the conversation (e.g., *"Betty"* and *"Larry"* in Figure 2).

3.2.1 Training with Occurrence Planning

Ground-truth samples for conditional training are built on gold summaries. First, given one dialogue sample and its reference summary, two entity sets are obtained by extracting all personal named entities from the source content and the gold summary respectively. Then, we take the intersection of the two sets, which represent the content coverage of the summary. For instance, given the example in Figure 2, the intersection is *{Larry, Amanda, Hannah, Betty}*. Next, in order to align the plan with gold summaries written in a certain perspective and narrative flow, we define **Occurrence Planning**, which reflects the order of personal named entities occurring in the gold summary. To this end, the entity set is re-ordered to *{Hannah, Betty, Amanda, Larry}*, and converted to a conditional sequence for training the controllable generation framework.

3.2.2 Inference: Comprehensive and Focus Planning Summarization Options

Once the model is trained on personal entity planning, one could customize the input conditional signal as a sequence of personal named entities based on downstream application needs. While our framework supports any combination and order of personal named entities that occurred in the given dialogue, here we focus on two conditional inputs during inference: (1) **Comprehensive Planning**, which includes all personal named entities in a source dialogue (they are ordered according to the occurrence order in the source) and aims to maximize information coverage. This type of summary supports general purpose use cases. (2) **Focus Planning** only targets one specific personal entity in the dialogue. Focus planning could be viewed as a subset of comprehensive planning and can be useful in more targeted applications.

3.3 Controllable Neural Generation

In our framework, a neural sequence-to-sequence network is used for conditional training and generation. As shown in Figure 3, the base architecture is a Transformer-based auto-regressive language model, since the Transformer (Vaswani et al., 2017) is widely adopted in various natural language processing tasks, and shows strong capabilities of contextual modeling (Devlin et al., 2019; Lewis et al., 2020). The input comprises a source dialogue with n tokens $D = \{w_0, w_1, ..., w_n\}$ and a pre-defined personal named entity planning with m tokens $C = \{c_0, c_1, ..., c_m\}$.

Encoder: The encoder consists of a stack of Transformer layers. Each layer has two sub-components: a multi-head layer with self-attention mechanism, and a position-wise feed-forward layer (Equation 1). A residual connection is employed between each pair of the two sub-components, followed by layer normalization (Equation 2).

$$\widetilde{h}^l = \text{LayerNorm}(h^{l-1} + \text{MHAtt}(h^{l-1})) \quad (1)$$

$$h^l = \text{LayerNorm}(\widetilde{h}^l + \text{FFN}(\widetilde{h}^l)) \quad (2)$$

where l represents the depth of the stacked layers, and h^0 is the embedded input sequence. MHAtt, FNN, LayerNorm are multi-head attention, feedforward and layer normalization components, respectively. Moreover, the additional linguistic feature (e.g., coreference information) is added in the encoded representations.

Decoder: The decoder also consists of a stack of Transformer layers. In addition to the two subcomponents in the encoding layers, the decoder inserts another component that performs multi-head attention over hidden representations from the last encoding layer. Then, the decoder generates tokens from left to right in an auto-regressive manner. The architecture and formula details are described in (Vaswani et al., 2017).

During training (see Figure 3), the planning sequence C under *Occurrence Planning* is concatenated with the source dialogue content D as the input with a special token. The segmentation tokens are pre-defined in different Transformer-based models, such as *'[SEP]'* in BERT and *'</s>'* in BART. The model learns to generate the ground truth $Y = \{y_0, y_1, ..., y_t\}$ (where t is the token number) by summarizing the information from the dialogue context conditioned on the planning sequence. The loss of maximizing the log-likelihood

on the training data is formulated as:

$$l(\theta) = -\sum_{t=1}^{T} \log p(y_t|D, C, y_{<t}, \theta) \quad (3)$$

During inference, we first specify one condition sequence based on the planning schemes described in Section 3.2. Specifically, one can assess the model's learning capability by generating summaries guided by *Occurrence Planning*. For simulating the real-world controllable generation scenario, *Comprehensive Planning* and *Focus Planning* can be applied. The model then creates a summary that is based on the specific condition which is coherent with the context of the input conversation.

4 Improving Factual Correctness

While current neural abstractive systems are able to generate fluent summaries, factual inconsistency remains an unsolved problem (Zhang et al., 2020). Neural models tend to produce statements that are not supported by the source content. These hallucinations are challenging to eradicate in neural modeling due to the implicit nature of learning representations. In document summarization, it has been demonstrated that a certain proportion of abstractive summaries contain hallucinated statements (Kryscinski et al., 2020), as is observed in dialogue summarization (Chen and Yang, 2020). Such hallucinations raise concerns about the usefulness and reliability of abstractive summarization, as summaries that perform well in traditional word-overlap metrics may fall short of human evaluation standards (Zhao et al., 2020).

4.1 Factual Inconsistency Detection

To evaluate and optimize the summarization quality regarding factual correctness, we first build a model to assess the accuracy of generated statements. Negative samples for classification are built via text manipulation, as is done in prior work (Zhao et al., 2020; Kryscinski et al., 2020). Since we focus on conditional personal named entities in this work, we aim to detect the inconsistency issues of person names between the source content and the generated summaries.

As shown in Figure 4, we construct a binary classifier by reading the dialogue and a summary. The classifier output evaluates if the two input entries are factually consistent. A reference summary in the original dataset is labeled as *'correct'*. To
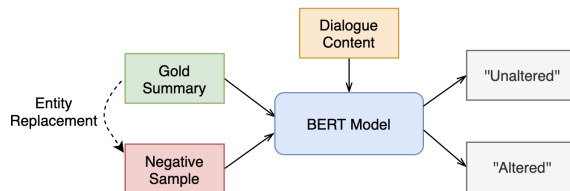

Figure 4: Factual inconsistency detection: a binary classification model that determines whether an input summary is altered with named entity replacement.

generate versions of this summary with label *'incorrect'*, we adopt three strategies to build negative samples: (1) Swapping the positions of where one pair of personal named entities are located in the gold summary with each other. The entities that are connected with word *"and"* and *"or"* in one sentence are excluded; (2) Replacing one name (e.g., *John*) in summaries with another randomly selected name of the same gender (e.g., *Peter*) in the source content; (3) Replacing one name with another from a person name collection built on the training data. With these samples, a *'BERT-base-uncased'* (Devlin et al., 2019) model was fine-tuned to classify whether the summary has been altered. The factual error detector achieved 91% F1 score on a hold-out validation set. To identify all personal named entities in both the conversation and the summary, Stanza Named Entity Recognition (NER) tagger (Qi et al., 2020) was used.

4.2 Exploiting Coreference Information

In conversations, speakers refer to themselves and each other and mention other objects/persons, resulting in various coreference links and chains across dialogue turns and speakers. We empirically observed that a sizable amount of errors stem from incorrect pronoun assignments in the generation process. Recent language models are also incapable of capturing coreference information without sufficient supervision (Dasigi et al., 2019). Thus, we exploit dialogue coreference resolution in a more explicit manner to enhance the model design as in (Liu et al., 2021).

To this end, we first use the AllenNLP toolkit (Gardner et al., 2017) for coreference resolution on the dialogue samples.[2] With the analyzed coreference mentions and clusters, we build a graph by connecting all nodes in each cluster. Here, we add bi-directional edges between each word/span and its neighboring referring mentions. Following (Liu

---

[2] allennlp-public-models/coref-spanbert-large-2021.03.10

et al., 2021), we incorporate the coreference information into the Transformer-based sequence-to-sequence model. Given a graph with n nodes, we represent the connected structure with an adjacency matrix A where $A_{ij} = 1$ if node i and node j are connected. For feature integration: (1) to model the linked information with a graph-based method, the multi-layer Graph Convolutional Network (GCN) is applied (Kipf and Welling, 2017). As shown in Figure 3, we feed hidden states from the last layer of the language encoder to the graph modeling component, then implicit features are computed and exchanged among tokens in the same coreference cluster, and we add them to the contextualized representation;[3] (2) we also conduct coreference information integration by adding self-attention layers and adopting head manipulation (Liu et al., 2021) which are parameter-efficient, and they can provide the same performance.

4.3 Data Augmentation via Entity Exchange

In addition to the data synthesis strategies in Section 4.1, we further propose an entity-based data augmentation to robustify the model, reducing incorrect correlations that might be made by the model due to data sparsity or imbalance classes. The augmented data is created by two steps: (1) a personal named entity pair with the same gender attribution is extracted; (2) we exchange them in both source content and reference summary to form new samples. For the data used in this paper, each conversation is independent from one another and each interlocutor from a particular dialogue is not an interlocutor in any other dialogue, nor is s/he mentioned in any other dialogue. Therefore, we postulate that this entity-based augmentation is helpful to reduce unnecessary inductive bias from the training data. In our experiment, the sample number of Data Augmentation (DA) is 4k.

5 Experimental Results and Analysis

5.1 Dataset Description

We conduct experiments with the proposed framework on SAMSum (Gliwa et al., 2019), a dialogue summarization dataset. It contains multi-turn daily conversations with human-written summaries. The data statistics are shown in Table 1. We retain the original text content of conversations such as cased

Type	Number
Training Set (14732 Samples)	
Mean/Std. of Dialogue Turns	11.7 (6.45)
Mean/Std. of Dialogue Length	124.5 (94.2)
Mean/Std. of Summary Length	23.44 (12.72)
Validation Set (818 Samples)	
Mean/Std. of Dialogue Turns	10.83 (6.37)
Mean/Std. of Dialogue Length	121.6 (94.6)
Mean/Std. of Summary Length	23.42 (12.71)
Testing Set (819 Samples)	
Mean/Std. of Dialogue Turns	11.25 (6.35)
Mean/Std. of Dialogue Length	126.7 (95.7)
Mean/Std. of Summary Length	23.12 (12.20)

Table 1: Details of the dialogue summarization dataset.

words, emoticons, and special tokens, and pre-process them using sub-word tokenization (Lewis et al., 2020). Since the positional embedding of the Transformer-based model can support 1,024 input length, none of the samples are truncated.

5.2 Model Configurations

To leverage the large-scale language models which provide semantically-rich contextualized representation to improve downstream tasks such as BERT (Devlin et al., 2019), we use the implementation of BART that is specially pre-trained for sequence-to-sequence language generation (Lewis et al., 2020),[4] to initialize parameters of the Transformer layers in Section 3.3, and fine-tune it to boost the performance on our dialogue summarization task.

The number of encoder layers, decoder layers, graph modeling layers, input and hidden dimension are $6/6/2/768$ for the *'BART-Base'* and $12/12/3/1024$ for the *'BART-Large'*, respectively. The learning rate of Transformer layers was set at $3e{-}5$, and that of the graph layers was set at $1e{-}3$. AdamW optimizer (Loshchilov and Hutter, 2019) was used with weight decay of $1e{-}3$ and a linear learning rate scheduler. Batch size was set to 8. Drop-out (Srivastava et al., 2014) of $rate = 0.1$ was applied as in the original BART configuration. The backbone parameter size is 139M for the *'BART-Base'* and 406M for for the *'BART-Large'*. For the data augmentation described in Section 4.3, we excluded samples that contain less than two personal named entities in their summaries. Best checkpoints were selected based on validation results of ROUGE-2 value. Tesla A100 with 40G memory was used for training and we used the Pytorch 1.7.1 as the computational framework (Paszke et al., 2019).

---

[3]Interested readers can refer to the Appendix for dialogue examples with coreference resolution information.

[4]https://huggingface.co/facebook/bart-{base,large}

	ROUGE-1			ROUGE-2			ROUGE-L		
	F	P	R	F	P	R	F	P	R
Pointer Generator*	40.1	-	-	15.3	-	-	36.6	-	-
DynamicConv + GPT-2*	41.8	-	-	16.4	-	-	37.6	-	-
Fast Abs RL Enhanced*	42.0	-	-	18.1	-	-	39.2	-	-
Multi-View BART-Large*	49.3	51.1	52.2	25.6	26.5	27.4	47.7	49.3	49.9
BART w/o Cond. (Base)	50.1	56.4	49.5	25.1	28.5	24.7	47.2	51.6	46.3
BART w/o Cond. (Large)	52.9	56.8	53.6	27.7	29.9	27.6	49.1	52.3	49.3
Generation with Occurrence Planning									
CTRLsum BART-Large (CNN/DM)	36.2	37.1	41.4	10.9	11.4	12.7	33.8	34.2	37.6
CTRLsum BART-Large (Fine-tuned)	54.0	58.7	54.9	30.1	31.7	30.5	51.9	55.7	53.1
Generation with Occurrence Planning (ours)									
Ctrl-DiaSumm (BART-Base)	52.3	57.0	52.6	27.6	30.2	27.6	50.2	53.1	50.1
Ctrl-DiaSumm+Coref	53.5	57.7	54.3	28.9	30.9	28.7	50.4	53.2	50.5
Ctrl-DiaSumm+Coref+DA	53.8	58.0	55.0	29.3	31.4	29.3	51.1	53.9	51.3
Generation with Occurrence Planning (ours)									
Ctrl-DiaSumm (BART-Large)	54.9	56.3	57.1	30.3	31.8	32.2	52.8	54.0	54.4
Ctrl-DiaSumm+Coref	55.3	57.5	57.9	31.3	32.9	32.8	53.2	55.0	55.2
Ctrl-DiaSumm+Coref+DA	56.0	59.8	57.6	31.7	34.4	32.2	54.1	57.8	55.3

Table 2: ROUGE scores on the SAMSum test set from baseline models and proposed methods. *Ctrl*, *Coref* and *DA* denote controllable, coreference modeling and data augmentation, respectively. F, P, R are F1 measure, precision, and recall. * denotes the reported results from (Chen and Yang, 2020). *BART w/o Cond.* is the baseline without entity planning conditional training. *CTRLsum* is the generic controllable summarizer proposed in (He et al., 2020), and we further fine-tuned it on the dialogue corpus with our entity planning scheme.

	Rouge-1 Recall	Rouge-2 Recall	Rouge-L Recall
BART w/o Cond.	53.6	27.6	49.3
Generation with Comprehensive Planning			
CTRLsum*	55.7	28.2	50.8
Ctrl-DiaSumm	56.3	28.2	51.4
Ctrl+Coref	58.1	28.4	52.5
Ctrl+Coref+DA	58.4	29.1	52.9

Table 3: ROUGE Recall scores under *Comprehensive Planning*. * *CTRLsum* model is fine-tuned on the dialogue dataset. See complete result table in Appendix.

	Rouge-1 Precision	Rouge-2 Precision	Rouge-L Precision
BART w/o Cond.	56.8	29.9	52.3
Generation with Focus Planning			
CTRLsum*	52.9	27.1	49.3
Ctrl-DiaSumm	52.4	27.0	49.7
Ctrl+Coref	53.1	27.2	49.9
Ctrl+Coref+DA	53.4	27.3	50.0

Table 4: ROUGE Precision scores under *Focus Planning*. * *CTRLsum* model is fine-tuned on the dialogue dataset. See complete result table in Appendix.

5.3 Quantitative Evaluation

We first conducted two evaluations with automatic metrics to assess the summarizers.

5.3.1 ROUGE Evaluation

We adopt ROUGE-1, ROUGE-2, and ROUGE-L, as ROUGE (Lin, 2004) is customary in summarization tasks to assess the output quality with gold summaries via counting n-gram overlap. We employ `Py-rouge` package to evaluate the models following (Gliwa et al., 2019; Feng et al., 2021).

Matched Training and Testing Conditions: We obtained summaries by conditioning the output generation with the personal named entities in the order they occur in the gold summary (i.e., *Oc-*

currence Planning). Since *Occurrence Planning* is extracted from the gold summaries, it serves as the upper-bound performance for the proposed conditional generation. As *Comprehensive Planning* and *Focus Planning* are mismatched testing conditions from the training process, we use *Occurrence Planning* to conduct a sanity check to ensure the proposed model performance meets expectations in idealistic scenarios where training and test conditions are matched: Table 2 shows that the conditional training in Section 3.2.1 is indeed effective. Moreover, the model with *'BART-Large'* backbone significantly performs better than that of *'BART-Base'*, thus we use it for the following generation evaluations. We also select a generic controllable

	Accuracy
BART w/o Cond.	77.4
Occurrence Planning	
CTRLsum (fine-tuned)	80.1
Ctrl-DiaSumm	79.9
Ctrl+Coref	81.5
Ctrl+Coref+DA	82.8
Comprehensive Planning	
CTRLsum (fine-tuned)	79.5
Ctrl-DiaSumm	79.0
Ctrl+Coref	80.8
Ctrl+Coref+DA	81.9
Focus Planning	
CTRLsum (fine-tuned)	74.9
Ctrl-DiaSumm	74.3
Ctrl+Coref	75.5
Ctrl+Coref+DA	76.2

Table 5: Automatic factual correctness evaluation on samples from baselines and our models.

	Consistency	Informative
BART w/o Cond.	0.71	0.70
Occurrence Planning		
Ctrl-DiaSumm	0.78	0.79
Ctrl+Coref+DA	0.79	0.81
Comprehensive Planning		
Ctrl-DiaSumm	0.74	0.83
Ctrl+Coref+DA	0.78	0.85
Focus Planning		
Ctrl-DiaSumm	0.68	0.70
Ctrl+Coref+DA	0.75	0.77

Table 6: Quality scoring on generated samples from models. Scores are normalized in the range of $[0, 1.0]$.

model *CTRLsum* (He et al., 2020) for comparison. We observed that the original *CTRLsum* trained on the news domain cannot generalize well on the dialogue corpus, and the performance can benefit from further fine-tuning.

Mismatched Training & Testing Conditions: As *Comprehensive Planning* covers the maximum number of personal entities in the dialogue, recall (a sensitivity measure) is more suitable in assessing its performance. Similarly, as *Focus Planning* only concerns a specific personal entity, precision (a specificity measure) is adopted. For evaluating *Focus Planning*, we randomly selected one speaker entity from each dialogue as condition input. While the aim of conditional summary generation (be it *Comprehensive Planning* or *Focus Planning*) is not to generate a summary that emulates the gold summary, we nonetheless provide comparison results with the unconditional baseline *'BART w/o Cond.'* for analysis purposes (see Appendix for complete ROUGE results and the generated summary examples). Results in Table 3 - 4 suggest: (1) Increasing the information coverage on personal named entities in dialogues improves general-purpose dialogue summarization; (2) Obtaining a reasonably accurate summary focused on a specified personal named entity is feasible; (3) Integrating coreference information and data augmentation improve performance consistently.

5.3.2 Factual Correctness Evaluation

We applied the factual consistency classifier built in Section 4.1 to assess the generated summaries using the accuracy metric (the proportion of samples that are predicted as true). As shown in Table 5, explicitly incorporating coreference information improves the accuracy of generated summaries guided with all conditional plannings, and data augmentation brings further improvements. Results of *Comprehensive Planning* is close to the upper-bound of *Occurrence Planning*. The difference is potentially due to the relatively longer generated summaries and the use of more novel words. Specifically, we observed that the novel word rate (See et al., 2017) of *Ctrl+Coref* under *Occurrence* and *Comprehensive* plannings are 0.28 and 0.33 respectively. The overall accuracy under *Focus Planning* is relatively lower, which is not unexpected, as more paraphrasing is needed for summarizing from a specified personal entity's perspective. Moreover, the fine-tuned *CTRLsum* performs similarly to the *Ctrl-DiaSumm* model, since both of them use *'BART-Large'* as the language backbone. However, here we did not pre-trained our models on out-of-domain summarization data.

5.4 Human Evaluation

5.4.1 Quality Scoring

We randomly sampled 50 dialogues with generated summaries for two linguistic evaluators to conduct quality scoring (Paulus et al., 2017). Since abstractive models fine-tuned on contextualized language backbones are able to generate fluent sentences (Lewis et al., 2020; Chen and Yang, 2020), we excluded *fluency* in the scoring criteria. Instead, *factual consistency* and *informativeness* were used to measure how accurate and comprehensive the extracted information is according to the source content. Summaries were scored of $[-1, 0, 1]$, where -1 means a summary was not factually consistent or failed to extract relevant information, 1 means it could be regarded as a human-written output, and

		Comprehensive Planning		Focus Planning	
	BART w/o Cond.	Ctrl-DiaSumm	Ctrl+Coref+DA	Ctrl-DiaSumm	Ctrl+Coref+DA
Average/Std. Length	21.3 (12.2)	26.52 (13.3)	27.05 (13.9)	15.44 (8.5)	15.87 (8.7)
Missing Information	17	6	4 [33% ↓]	16	12 [25% ↓]
Wrong References	11	11	8 [27% ↓]	14	11 [21% ↓]
Incorrect Reasoning	10	12	9 [25% ↓]	13	10 [23% ↓]
Improper Gender	2	2	1 [50% ↓]	5	3 [40% ↓]

Table 7: Error analysis on 50 samples. Values in round brackets denote standard deviations of length. Numbers are counted if one error is labeled in generated summaries. Values in square brackets denote the relative decrease.

0 means it extracted some relevant information or made minor mistakes. We averaged the normalized scores from evaluators. As shown in Table 6, *Comprehensive Planning* obtains slightly lower scores in *consistency* (related to ROUGE precision score) than the training scheme *Occurrence Planning*, but it achieves higher *informativeness* scores, which is consistent with the improvement on ROUGE recall scores in Table 3. Moreover, the proposed model (*Ctrl+Coref+DA*) outperforms base model significantly under *Focus Planning*.

5.4.2 Error Analysis

Similar to previous work (Chen and Yang, 2020), we conducted error analysis by checking the following 4 error types: (1) Missing information: content mentioned in references is missing in generated summaries; (2) Wrong references: generated summaries contain information that is not faithful to the original dialogue, or associate actions with wrong named entities; (3) Incorrect reasoning: the model learned incorrect associations leading to wrong conclusions in the generated summary; (4) Improper gendered pronouns. Linguistic analysts were given 50 dialogues randomly chosen from the test set and their corresponding summaries from baselines and our models. They were asked to read the dialogue content and summaries and judge if the 4 error types occurred. For each evaluator, the sequence of presentation was randomized differently. As shown in Table 7, summaries under both planning schemes make significantly fewer errors at all fronts. Under *Comprehensive Planning*, models with conference information and data augmentation (*Ctrl+Coref+DA*) outperform the base model especially in consistency-related classes. Under *Focus Planning*, both models produce more factual incorrectness due to more paraphrasing from various personal perspectives, this matches the result from automatic factual consistency evaluation in Section 5.3.2, and the *Ctrl+Coref+DA* model achieves significant quality improvement.

6 Conclusion

In this work, we proposed a controllable neural framework for abstractive dialogue summarization. In particular, a set of personal named entities were used to condition summary generation. This framework could efficiently tailor to different user preferences and application needs, via modulating entity planning. Moreover, the experimental results demonstrated that the abstractive model could benefit from explicitly integrating coreference resolution information, achieving better performance on factual consistency and standard metrics of word-overlap with gold summaries.

Acknowledgments

References

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Jiaao Chen and Diyi Yang. 2020. Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization. In *Proceedings of EMNLP 2020*, pages 4106–4118. Association for Computational Linguistics.

Pradeep Dasigi, Nelson F Liu, Ana Marasovic, Noah A Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of EMNLP 2019*, pages 5927–5934.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of

deep bidirectional transformers for language understanding. In *Proceedings of NAACL 2019*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. Gsum: A general framework for guided neural abstractive summarization. In *Proceedings of NAACL 2021*, pages 4830–4842. Association for Computational Linguistics.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

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. Association for Computational Linguistics.

Xiachong Feng, Xiaocheng Feng, Libo Qin, Bing Qin, and Ting Liu. 2021. Language model as an annotator: Exploring DialoGPT for dialogue summarization. In *Proceedings of ACL 2021*, pages 1479–1491. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of EMNLP 2018*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79. Association for Computational Linguistics.

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.

Travis Goodwin, Max Savery, and Dina Demner-Fushman. 2020. Towards zero shot conditional summarization with adaptive multi-task fine-tuning. In *Proceedings of EMNLP 2020*, pages 3215–3226.

Prakhar Gupta, Jeffrey P Bigham, Yulia Tsvetkov, and Amy Pavel. 2020. Controlling dialogue generation with semantic exemplars. *arXiv preprint arXiv:2008.09075*.

Junxian He, Wojciech Kryściński, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2020. Ctrlsum: Towards generic controllable text summarization. *arXiv preprint arXiv:2012.04281*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of NeurIPS 2015*, pages 1693–1701.

Eduard Hovy, Chin-Yew Lin, et al. 1999. Automated text summarization in summarist. *Advances in automatic text summarization*, 14:81–94.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596.

Xinyu Hua and Lu Wang. 2020. PAIR: Planning and iterative refinement in pre-trained transformers for long text generation. In *Proceedings of EMNLP 2020*, pages 781–793. Association for Computational Linguistics.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of NAACL 2018*, pages 1875–1885.

Hongyan Jing and Kathleen McKeown. 2000. Cut and paste based text summarization. In *1st Meeting of NAACL*.

Chris Kedzie, Kathleen McKeown, and Hal Daume III. 2018. Content selection in deep learning models of summarization. In *Proceedings of EMNLP 2018*, pages 1818–1828, Brussels, Belgium. Association for Computational Linguistics.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. *The 5th International Conference on Learning Representations (ICLR 2017)*.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. In *Proceedings of the EMNLP 2019*, pages 540–551, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of EMNLP 2020*, pages 9332–9346.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, page 68–73, New York, NY, USA. Association for Computing Machinery.

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 ACL 2020*,

pages 7871–7880. Association for Computational Linguistics.

Manling Li, Lingyu Zhang, Heng Ji, and Richard J. Radke. 2019. Keep meeting summaries on topic: Abstractive multi-modal meeting summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2190–2196, Florence, Italy. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 510–520, Portland, Oregon, USA. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of EMNLP 2019*, pages 3721–3731, Hong Kong, China. Association for Computational Linguistics.

Zhengyuan Liu, Angela Ng, Sheldon Lee, Ai Ti Aw, and Nancy F Chen. 2019. Topic-aware pointer-generator networks for summarizing spoken conversations. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 814–821. IEEE.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2020. Conditional neural generation using sub-aspect functions for extractive news summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1453–1463. Association for Computational Linguistics.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2021. Coreference-aware dialogue summarization. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 509–519, Singapore and Online. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *The International Conference on Learning Representations (ICLR 2019)*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of AAAI 2017*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of NeurIPS 2019*, pages 8026–8037.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of EMNLP 2015*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Harvey Sacks, Emanuel A Schegloff, and Gail Jefferson. 1978. A simplest systematics for the organization of turn taking for conversation. In *Studies in the organization of conversational interaction*, pages 7–55. Elsevier.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1073–1083.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Proceedings of NeurIPS 2017*, pages 6830–6841.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

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 *Proceedings of NeurIPS 2017*, pages 5998–6008.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *Proceedings of ACL 2020*, pages 5108–5120. Association for Computational Linguistics.

Zheng Zhao, Shay B. Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249. Association for Computational Linguistics.

A Appendix

	ROUGE-1			ROUGE-2			ROUGE-L		
	F	P	R	F	P	R	F	P	R
Generation with Comprehensive Planning on Subset-A									
CtrlSum (fine-tuned) (He et al., 2020)	54.1	56.1	55.3	27.1	29.3	27.8	48.7	50.8	48.5
Ctrl-DiaSumm	53.7	54.3	55.9	25.7	26.9	27.9	48.6	49.9	49.4
Ctrl+Coref	53.9	55.7	56.9	27.1	28.1	28.0	49.0	50.1	51.0
Ctrl+Coref+DA	54.6	56.5	57.4	27.6	29.1	28.6	49.7	51.0	51.5
Generation with Comprehensive Planning on Subset-B									
CtrlSum (fine-tuned) (He et al., 2020)	46.7	47.2	56.1	24.0	23.1	28.3	44.6	43.7	51.5
Ctrl-DiaSumm	47.3	43.8	57.6	23.5	21.7	29.1	43.3	40.8	51.7
Ctrl+Coref	47.9	44.3	59.1	23.7	22.1	29.4	44.0	41.2	53.2
Ctrl+Coref+DA	48.4	44.7	59.7	24.4	22.4	30.8	45.2	41.7	54.0

Table 8: ROUGE scores on summaries under the *Comprehensive Planning*. *Ctrl*, *Coref* and *DA* denote controllable, coreference modeling and data augmentation, respectively. F, P, R are F1 measure, precision, and recall. For fair comparison with ground-truth summaries, we split the test set into two subsets: Subset-A (461 of 819 test samples) contains the samples that personal entity set extracted from gold summaries and that of *Comprehensive Planning* is the same, and the rest 358 samples are included in Subset-B.

	ROUGE-1			ROUGE-2			ROUGE-L		
	F	P	R	F	P	R	F	P	R
Generation with Focus Planning									
CtrlSum (fine-tuned) (He et al., 2020)	47.0	52.9	45.7	23.3	27.1	23.1	44.5	49.3	43.7
Ctrl-DiaSumm	47.0	52.4	46.8	23.0	27.0	22.8	44.8	49.7	44.1
Ctrl+Coref	47.1	53.1	47.0	23.4	27.2	23.3	45.1	49.9	44.6
Ctrl+Coref+DA	47.4	53.4	47.9	23.8	27.3	23.9	45.3	50.0	45.0

Table 9: ROUGE scores on summaries under the *Focus Planning*. *Ctrl*, *Coref* and *DA* denote controllable, coreference modeling and data augmentation, respectively. F, P, R are F1 measure, precision, and recall. For the *Focus Planning*, we randomly selected one speaker entity from each dialogue as condition input. Worth-mentioned that the average length of generation under *Focus Planning* is smaller than that of *Comprehensive Planning*, resulting in some decrease of recall performance. Moreover, more paraphrasing is needed for generated difference summaries from different personal perspectives, as the examples shown in Table 11.

		Occurrence Planning		Comprehensive Planning	
	BART w/o Cond.	Ctrl-DiaSumm	Ctrl+Coref+DA	Ctrl-DiaSumm	Ctrl+Coref+DA
Average/Std. Length	21.3 (12.2)	20.96 (9.75)	19.82 (9.47)	26.52 (13.3)	27.05 (13.9)
Missing Information	17	6	4 [33% ↓]	6	4 [33% ↓]
Wrong References	11	7	6 [14% ↓]	11	8 [27% ↓]
Incorrect Reasoning	10	9	8 [11% ↓]	12	9 [25% ↓]
Improper Gender	2	1	1 [0% ↓]	2	1 [50% ↓]

Table 10: Error analysis on 50 samples from the baseline and our models. Here we compare the *Comprehensive Planning* with the training scheme *Occurrence Planning*. Values in round brackets are the standard deviation of summary length. Numbers are counted if there is an error labeled in the generated summary. Values in square brackets are the relative decrease.

Conversation	Reference Summary	Focus Planning
(i) <Natalie>: Well well weeeeeell, I see somethings going on here at last. <Martin>: (Y). Adam: any confirmed data? <Anna>: Hello everyone!!! Id love to invite everybody to my bday. I would be extremaly happy if you could come 6th of November at 19:30. <Martin>: (smile)] <Margot>: (smile) <Mia>: (Y)	Anna organises a birthday's party on the 6th of November at 19:30.	**Adam** will come to Anna's birthday party on 6th November at 19:30. **Anna** invites everyone to her birthday on 6th November at 19:30.
(ii) <Anne>: You were right, he was lying to me :/ <Irene>: Oh no, what happened? <Jane>: who? that Mark guy? <Anne>: yeah, he told me he's 30, today I saw his passport - he's 40 <Irene>: You sure it's so important? <Anne>: he lied to me Irene.	Mark lied to Anne about his age. Mark is 40.	**Anne** saw a man's passport today. He's 40. **Jane**'s friend lied to her about him being 30 years old.
(iii) <Josh>: Stephen, I think you've accidentaly taken my notebook home <Stephen>: wait lemme check. <Stephen>: nope, I don't see it anywhere <Jack>: oh xxx, I've got it xDDD I don't even know why. <Josh>: xDDD ok, no problem, cool I know where it is. <Jack>: I'll bring it tomorrow.	Josh thinks Stephen accidentally took his notebook. Jack has it and will bring it tomorrow.	**Jack** has taken Stephen's notebook and will bring it tomorrow. **Stephen** has left his notebook at home. He can't find it.
(iv) <George>: What have you gotten for Christmas? <Jacob>: I got a punchbag. <Jenny>: I got training shoes. <George>: Sporty team :P <Jenny>: What did you get? <George>: A cooking pot :-) <Jacob>: Your wife wants you to help her in the kitchen? <George>: It's me who is normally cooking. <George>: I really like it :P <George>: Jenny gave me this pot, it's amazing and has life long guarantee. <Jacob>: Cool <Jenny>: I wish my Michael was a better cook. <Jenny>: I think it's really sexy when a guy can cook well.	Jacob, Jenny and George are telling each other what they have gotten for Christmas.	**George** got a cooking pot for Christmas. His wife wants him to help her in the kitchen. **Jenny** got a sports bag for Christmas, a cooking pot and training shoes.

Table 11: Examples of generated summaries with *Focus Planning*. Speaker roles are bracketed, and the focused personal named entity is highlighted.

Dialogue Content:
Cristina: Hey
Cristina: Do you think we can meet up next week to discuss the shoot?
Gaya: Yes, I just need to double check my schedule
Gaya: If you like we can meet at my studio?
Gaya: It's easier
Cristina: Ok
Cristina: I'm more free on Wednesday and Friday
Gaya: Ok ;)

Reference: Gaya and Cristina are going to meet at Gaya's studio to talk about the photoshoot. Cristina prefers Wednesdays and Fridays.

BART-based: Gaya will meet Cristina at her studio next week to discuss the shoot.

Ctrl+BART+Coref+DA, Occurrence Plan:
Gaya and Cristina will meet next week at Gaya's studio to discuss the shoot.

Ctrl+BART+Coref+DA, Comprehensive Plan:
Cristina and Gaya will meet next week to discuss the shoot at Gaya's studio. Cristina is more free on Wednesday and Friday.

Dialogue Content:
Jane: hey david, you're coming home for christmas next week right?
David: of course
Jane: good
Jane: do you know what your dad would like for christmas?
Jane: i can't think of anything
David: you should get him an ipad
David: he can read books, email, watch movies, play games
Jane: ok, that sounds good, where can i get one?
David: i'll order it online and have it shipped home
Jane: thanks for your help
Jane: and please let me know when you'll get here once your travel arrangements are set.

Reference: David is coming home for Christmas next week. Jane has no idea what to buy their father so David is going to order an ipad online.

BART-based: David is coming home for Christmas next week. Jane wants to buy him an ipad. David will order it online and have it shipped home.

Ctrl+BART+Coref+DA, Occurance Plan:
David will order an ipad online and have it shipped home for Christmas. Jane will let David know when she arrives.

Ctrl+BART+Coref+DA, Comprehensive Plan:
David is coming home for Christmas next week. He will order an ipad for his dad. Jane will let him know when he arrives once his travel arrangements are set.

Dialogue Content:
Steve: hiya the dishwasher has turned up
Mum: good did they install it for you
Steve: yes they did a good job
Steve: <file_photo>
Mum: looks really nice
Steve: yes it does its nice and quiet
Mum: is it a 12 place setting one?
Steve: I think so, not that I ever have 12 for dinner lol
Mum: not but you dont have to put it on every day
Steve: no every other
Mum: yes I'm glad you are pleased with it xx
Steve: I am thanks mum xxx

Reference: Steve is happy that he got a new dishwasher installed.

BART-based: Steve's dishwasher has finally turned up. Mum is glad Steve is pleased with it.

Ctrl+BART+Coref+DA, Occurrence Plan:
Steve has a new dishwasher installed. Mum is glad Steve is pleased with it.

Ctrl+BART+Coref+DA, Focus Plan {Steve}:
Steve's new dishwasher has arrived. It is nice and quiet. Steve doesn't have to use a 12 place setting dishwasher every day.

Figure 5: Dialogue examples with summaries from a baseline model and our controllable generation.

Figure 6: Dialogue examples with coreference resolution information. Words/Spans in one coreference cluster are labeled with the same color. Noted that this is the original output from AllenNLP (Gardner et al., 2017) coreference resolution tool.