Structural Constraints and Natural Language Inference for End-to-End Flowchart Grounded Dialog Response Generation

Dinesh Raghu [*] [1] [2], **Suraj Joshi** [1], **Sachindra Joshi** [2] **and Mausam** [1]
[1] Indian Institute of Technology, New Delhi, India
[2] IBM Research, New Delhi, India
diraghu1@in.ibm.com, mt1180045@iitd.ac.in,
jsachind@in.ibm.com, mausam@cse.iitd.ac.in

Abstract

Flowchart grounded dialog systems converse with users by following a given flowchart and a corpus of FAQs. The existing state-of-the-art approach (Raghu et al., 2021) for learning such a dialog system, named FLONET, has two main limitations. (1) It uses a Retrieval Augmented Generation (RAG) framework which represents a flowchart as a bag of nodes. By doing so, it loses the connectivity structure between nodes which can aid in better response generation. (2) Typically dialogs progress with the agent asking polar (Y/N) questions, but users often respond indirectly without the explicit use of polar words. In such cases, it fails to understand the correct polarity of the answer. To overcome these issues, we propose Structure-Aware FLONET (SA-FLONET) which infuses structural constraints derived from the connectivity structure of flowcharts into the RAG framework. It uses natural language inference to better predict the polarity of indirect Y/N answers. We find that SA-FLONET outperforms FLONET, with a success rate improvement of 68% and 123% in flowchart grounded response generation and zero-shot flowchart grounded response generation tasks respectively.

1 Introduction

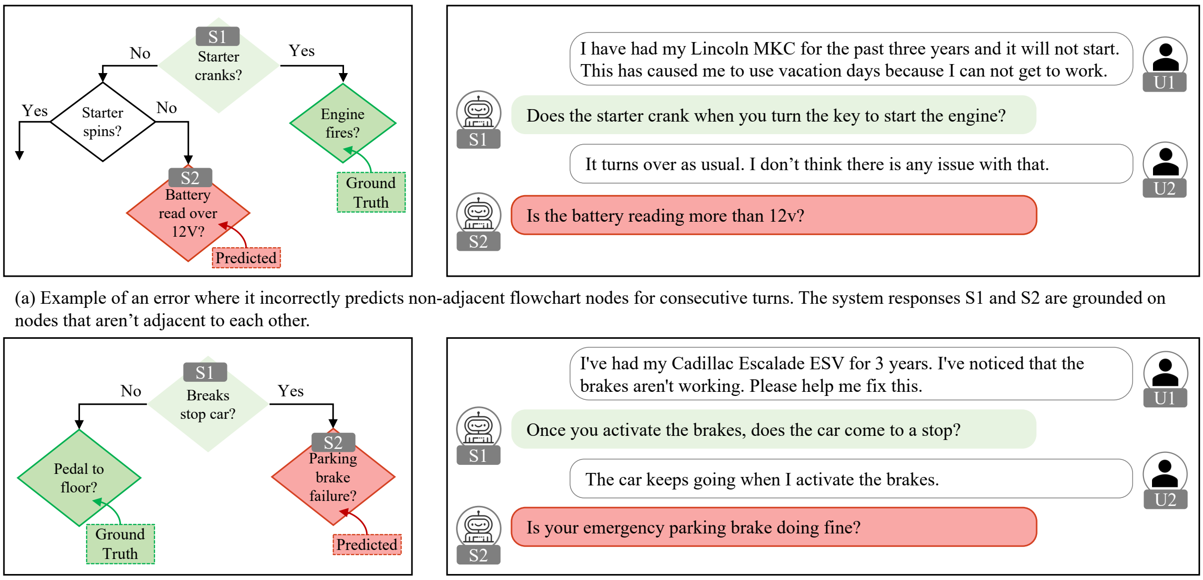
Task-oriented dialog systems converse with users to achieve a specific task (e.g., restaurant recommendation) using information from an associated knowledge source (e.g., a KB of restaurants). End-to-end approaches (Bordes and Weston, 2017; Madotto et al., 2018; Qin et al., 2020) have been proposed to learn these dialog systems, which require just the chat transcripts and no additional annotations. Recently, Raghu et al. released FLODIAL, a dataset for learning end-to-end flowchart grounded task oriented dialogs in which each dialog follows an associated flowchart.

The best performing end-to-end approach for learning flowchart grounded dialogs (Raghu et al., 2021) has two limitations. It follows a retrieval augmented generation (RAG) framework, which first retrieves a document (e.g., flowchart node) based on the dialog history and then generates the response using the retrieved document. The first issue arises due to the representation of the flowchart as a bag of nodes by the RAG retriever, which fails to capture the node connectivity structure in the flowchart. Due to this shortcoming, the model incorrectly grounds consecutive utterances in a dialog on non-adjacent flowchart nodes, whereas they are expected to be grounded on adjacent nodes. For example, in Figure 1(a) the system utterance S2 is grounded on a node that is not adjacent to the node corresponding to the previous system utterance.

Secondly, agents typically ask polar (Y/N) questions to traverse the flowchart. For example, the agent may ask ‘Does the car stop when you apply the brakes?”. Users often respond indirectly to these questions without any explicit use of polar words. For example, the user can indirectly respond to the agent question as “The car keeps going when I activate the brakes”. As shown in Figure 1(b), the existing approach fails to map the indirect user response to the correct polarity.

To tackle these two limitations, we propose *Structure Aware* - FLONET (SA-FLONET) that also follows the RAG framework. Our proposed approach augments the RAG retriever with *memory* to enable the retriever to store past decisions. The memory when used along with the flowchart structure ensures the dialog mostly follows a path in the flowchart. This helps consecutive utterances be grounded on adjacent nodes in the flowchart. SA-FLONET poses understanding user utterances as a natural language inference (NLI) task to leverage complex language reasoning ability from existing NLI models. We evaluate the performance SA-FLONET on the FLODIAL dataset. Com-

[*] D. Raghu is an employee at IBM Research. This work was carried out as part of PhD research at IIT Delhi.



U1: I have had my Lincoln MKC for the past three years and it will not start. This has caused me to use vacation days because I can not get to work.

S1: Does the starter crank when you turn the key to start the engine?

U2: It turns over as usual. I don't think there is any issue with that.

S2: Is the battery reading more than 12v?

(a) Example of an error where it incorrectly predicts non-adjacent flowchart nodes for consecutive turns. The system responses S1 and S2 are grounded on nodes that aren't adjacent to each other.

U1: I've had my Cadillac Escalade ESV for 3 years. I've noticed that the brakes aren't working. Please help me fix this.

S1: Once you activate the brakes, does the car come to a stop?

U2: The car keeps going when I activate the brakes.

S2: Is your emergency parking brake doing fine?

(b) Example of an error where the system is unable to map the correct polarity for an indirect user response. The system incorrectly understands the user utterance U2 as a YES to the question in the system utterance S1.

Figure 1: Two main types of errors made by FLONET (a) incorrectly predicting non-adjacent nodes for consecutive turns, and (b) failing to map indirect user responses to the correct polarity.

pared to the state-of-the-art approach, SA-FLONET improves the task specific metric (i.e., success rate) by 68% and 123% on flowchart grounded response generation (FGRG) and zero-shot flowchart grounded response generation (ZS-FGRG).

To summarize, we make the following contributions:

1. We propose SA-FLONET[2] for learning end-to-end flowchart grounded dialogs. It augments the RAG retriever with memory and infuses structural constraints into the retrieval process.
2. SA-FLONET uses NLI to map user's indirect response to agent's polar questions.
3. SA-FLONET outperforms existing approaches on FGRG and ZS-FGRG tasks.

2 Related Work

In this work, we propose a novel neural architecture for end-to-end flowchart grounded response generation and its zero-shot variant. Our main contributions are (1) augmenting *RAG* (Lewis et al., 2020) with memory for infusing structural constraints of flowcharts and (2) using *NLI* (Bowman et al., 2015; Williams et al., 2018) for predicting polarity of indirect Y/N answers. We now briefly discuss existing literature that uses RAG for dialog applications and then review works related to NLI.

Retrieval Augmented Generation: The RAG framework has been extensively used for knowledge intensive language generation tasks such as open-domain dialog response generation (Shuster et al., 2021; Xu et al., 2022), open-domain question answering (Lewis et al., 2020), document grounded task-oriented dialog response generation (Thulke et al., 2021) and flowchart grounded task-oriented dialog response generation (Raghu et al., 2021). Raghu et al. (2021) represent the flowchart as a bag of nodes and lose the inherent structure of the flowchart in the process. To the best of our knowledge, we are the first to incorporate structural constraints into the RAG framework by augmenting the retriever with a memory.

Natural Language Inference: NLI predicts whether a hypothesis entails, contradicts or is neutral to a given premise. NLI has been used for modelling persona-based dialogs (Song et al., 2020) to ensure the response generated by the dialog systems are consistent with a given persona description. NLI has also been used to ensure consistency within a dialog by ensuring generated responses do not contradict one another (Welleck et al., 2019). While NLI has been used for making the generated responses consistent, they have never been used for language understanding in dialogs. Although, NLI has been used for language understanding in question-answering (Louis et al., 2020). To the best of our knowledge, we are the first to use NLI for

[2] https://github.com/dair-iitd/sa-flonet

language understanding in a dialog setting.

3 Preliminaries

In this section, we describe the problem of end-to-end learning of flowchart grounded task oriented dialogs. We then briefly describe the previous work (FLONET) (Raghu et al., 2021) over which we build our proposed approach.

3.1 Problem Formulation

Let a dialog d between a user u and an agent a be represented as $\{c_i^u, c_i^a\}_{i=1}^m$ where m is the number of exchanges. Let $\mathcal{F} = (N, E)$ be the flowchart with a set of nodes N and edges E associated with d. Nodes and edges represent agent utterances and user responses respectively. Let $\mathcal{Q}$ be a set of frequently asked questions (FAQs). The task is to generate an agent response $\mathbf{y} = c_i^a =< y_1, y_2, ..., y_T >$ at turn i given (1) the dialog history $\mathbf{h}_i = \{c_1^u, c_1^a, ..., c_i^u\}$, (2) the flowchart $\mathcal{F}$ and (3) the set of FAQs ($\mathcal{Q}$).

3.2 FLONET

FLONET is the state-of-the-art approach for learning flowchart grounded task oriented dialogs in an end-to-end manner. It follows the RAG sequence model (Lewis et al., 2020). The RAG sequence model has two main components: (1) a retriever $p_\eta^{con}(z|\mathbf{h}_i)$ which computes a distribution over retrievable documents z (i.e., flowchart nodes and FAQs) based on the dialog history $\mathbf{h}_i$ and (2) a generator $p_\theta(y_t|\mathbf{h}_i, z, y_{1:t-1})$ which generates the agent response token-by-token. The overall RAG model is given by,

$$p(\mathbf{y}|\mathbf{h}_i) = \sum_{z \in N \cup Q} p_\eta^{con}(z|\mathbf{h}_i) \prod_{t=1}^{T} p_\theta(y_t|\mathbf{h}_i, z, y_{1:t-1})$$

The flowchart nodes and the FAQs together form the set of retrievable documents. The RAG model is trained end-to-end by using a retrievable document as a latent variable and uses a *top-k* approximation to marginalize over retrievable documents. The FLONET retriever p_η^{con} computes the probability for each retrievable document z as follows,

$$p_\eta^{con}(z|\mathbf{h}_i) = \frac{e^{-d(\phi_z(z), \phi_h(\mathbf{h}_i))}}{\sum_{z' \in N \cup Q} e^{-d(\phi_z(z'), \phi_h(\mathbf{h}_i))}}$$

where $d(.,.)$ is the Euclidean distance, $\phi_z(.)$ and $\phi_h(.)$ are hierarchical recurrent encoders (Sordoni et al., 2015). We refer to the FLONET retriever $p_\eta^{con}(z|\mathbf{h}_i)$ as content based scorer in the remainder of the paper. FLONET uses GPT2 (Radford et al., 2019) as its generator.

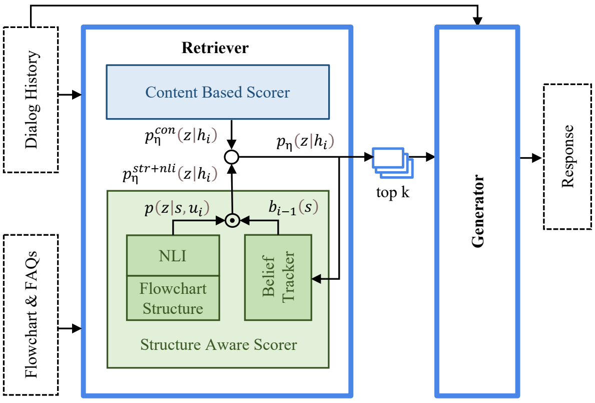


Figure 2: The architecture of SA-FLONET with structure aware scorer.

4 Structure-Aware FLONET

SA-FLONET follows the RAG sequence model. Figure 2 shows the overall architecture of SA-FLONET. The SA-FLONET retriever $p_\eta(z|\mathbf{h}_i)$ consists of two scorers: (1) a content based scorer (CBS) $p_\eta^{con}(z|\mathbf{h}_i)$, and (2) a structure-aware scorer (SAS) $p_\eta^{str+nli}(z|\mathbf{h}_i)$. The final retriever output is computed by normalizing the elementwise product of the two scorer outputs. SA-FLONET uses GPT2 as its generator p_θ. The overall response generator network is given by,

$$p(\mathbf{y}|\mathbf{h}_i) = \sum_{z \in N \cup Q} p_\eta(z|\mathbf{h}_i) \prod_{t=1}^{T} p_\theta(y_t|\mathbf{h}_i, z, y_{1:t-1})$$

where,

$$p_\eta(z|\mathbf{h}_i) = \frac{p_\eta^{con}(z|\mathbf{h}_i) * p_\eta^{str+nli}(z|\mathbf{h}_i)}{\sum\limits_{z' \in N \cup Q} p_\eta^{con}(z'|\mathbf{h}_i) * p_\eta^{str+nli}(z'|\mathbf{h}_i)} \tag{1}$$

The novel contribution of SA-FLONET is the structure-aware scorer $p_\eta^{str+nli}$ which maintains a belief of where in the flowchart the dialog is grounded. This belief along with the flowchart connectivity structure helps infuse flowchart *str*uctural constraints into the retriever. The use of flowchart structure serves as a backbone to plug in natural language inference (*NLI*) which improves the retriever's ability to predict the polarity of indirect user responses to agent's polar questions.

4.1 Structural Constraints

Dialogs typically follow a path in the flowchart. In order to imbibe this information into the model, it should remember the path traversed by the dialog so far. To remember the path, SAS uses a belief tracker that maintains a belief distribution b_i at turn i. The belief is distributed over a state space $\mathcal{S}$, where each state $s \in \mathcal{S}$ corresponds to a node in the given flowchart $\mathcal{F}$. Using the previous belief b_{i-1} and the connectivity structure in the flowchart, we can compute a score for each retrievable document z as follows:

$$\begin{aligned} p_\eta^{str+nli}(z|\mathbf{h}_i) &= \sum_{s\in\mathcal{S}} p(z|s,u_i).p(s,u_i|\mathbf{h}_i) \\ &= \sum_{s\in\mathcal{S}} p(z|s,u_i).p(s|\mathbf{h}_i) \\ &\approx \sum_{s\in\mathcal{S}} p(z|s,u_i).p(s|\mathbf{h}_{i-1}) \\ &= \sum_{s\in\mathcal{S}} p(z|s,u_i).b_{i-1}(s) \end{aligned} \tag{2}$$

The distribution $p(z|s,u_i)$ is computed using an NLI model and is described in Section 4.2. $p(s,u_i|\mathbf{h}_i)$ is equated to $p(s|\mathbf{h}_i)$ as u_i is independent of s and $p(u_i|\mathbf{h}_i) = 1$ as u_i is a part of $\mathbf{h}_i$. We then approximate $p(s|\mathbf{h}_i)$ to $p(s|\mathbf{h}_{i-1})$ as we do not use the user utterance u_i, but only the previous belief b_{i-1} computed using h_{i-1}.

Belief Update: Once the retriever generates the distribution over the documents $p_\eta(z|\mathbf{h}_i)$, it is passed to the belief tracker to update its belief b_i. The retriever's output distribution over the document space (flowchart nodes & FAQs) is converted to the belief over the state space $\mathcal{S}$ (over flowchart nodes) as follows:

$$\begin{aligned} b_i(s) = &\sum_{z\in N} \mathbb{1}_{z=s}.p_\eta(z|\mathbf{h}_i) \\ &+ b_{i-1}(s).\sum_{z\in Q} p_\eta(z|\mathbf{h}_i) \end{aligned} \tag{3}$$

The belief update for a state s is a sum of (1) the probability of the flowchart node corresponding to the state, and (2) the sum of all FAQ probabilities weighted by its previous belief. The second term prevents the belief tracker from forgetting its current state when the dialog moves away from the flowchart node to an FAQ. At $i = 0$ the belief is initialized with the root node having a higher probability compared to other nodes. Specifically, the state associated with the root node is made three times more likely than the other nodes.

4.2 Natural Language Inference (NLI)

Agents typically ask polar (Y/N) questions to traverse the flowchart. The existing approach often fails to correctly map the response to Yes/No when the user conveys it indirectly. To overcome this issue, we pose the task of understanding responses to the polar questions as an NLI task.

We use SemBERT (Zhang et al., 2020) as the NLI model. It takes a flowchart node text associated with the state s (premise) and current user utterance u_i (hypothesis) as input and predicts a distribution over entailment (p_e), contradiction (p_c) and neutral (p_n). $p(z|s,u_i)$ is computed using the output of the NLI model as follows,

$$p(z|s,u_i) = \begin{cases} p_e & \text{if } z \text{ is a YES child of } s \\ p_c & \text{if } z \text{ is a NO child of } s \\ p_n/N_Q & \text{if } z \in Q \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where N_Q is the number of FAQs. We perform additive smoothing on $p(z|s,u_i)$ (smoothing parameter set to 1E-4) to assign non-zero probabilities to all documents. Equation 4 incorporates both structure and NLI. It uses the user utterance u_i to assign probabilities to the YES and NO child based on p_e and p_c respectively. When the user response is neither a YES nor a NO, then it implies the user has digressed and hence the focus should be directed towards the FAQs. Hence the probability p_n is distributed across the FAQs.

4.3 Fine-tuning NLI

The traditional NLI task predicts if a hypothesis entails, contradicts or is neutral to a given premise. In our setting, we map the premise to a question associated with a flowchart node (e.g., Does the car stop when you apply the brakes?) and the hypothesis can be any of the three: positive response (e.g., Yes, it stops), negative response (e.g., The car keeps going when I activate the brakes), or neutral (e.g., I can't even start the car, forget about the brakes working). Since the interpretation of entails and contradicts has to now map to positive and negative responses respectively, we first finetune an

NLI model to learn our mapping before using it in the structure aware scorer.

We construct data to finetune the NLI model by using examples from two sources: (1) distantly supervised data constructed based on the intermediate document ranking of FLONET on the training data, and (2) Circa (Louis et al., 2020), a large-scale question answering dataset for learning indirect responses to polar questions. To construct the distantly supervised data, we first run each context-response pair in the train data through FLONET to identify the document (flowchart node or FAQ) used for response generation. Let (a_{i-1}, a_i) be two consecutive agent utterances in a dialog. Let z_{i-1} and z_i be the documents which were used by FLONET for generating a_{i-1} and a_i respectively. We now construct the distantly supervised data by using a_{i-1} as the premise, the next user response u_i as the hypothesis and the label is assigned based on z_{i-1} and z_i. The labels are assigned as follows:

1. If z_{i-1}, z_i are flowchart nodes and , z_{i-1} is the parent of z_i with an YES edge between them, then assign *entailment*.
2. If z_{i-1}, z_i are flowchart nodes and , z_{i-1} is the parent of z_i with an NO edge between them, then assign *contradicts*.
3. If z_{i-1} is a flowchart nodes and z_i is a FAQ, then assign *neutral*.
4. For all other cases, we skip the example.

The distantly supervised data constructed based on FLONET matches without interpretation of entailment, contradiction and neutral. Some examples of the constructed data are in Appendix B.

4.4 Only Structural Constraints

To study the contribution of NLI in SAS ($p_\eta^{str+nli}$), we propose an SAS variant (p_η^{str}) which uses only the structural constraints and no NLI. We refer to the overall network with p_η^{str} instead of $p_\eta^{str+nli}$ in Equation 1 as FLONET + Structural Constraints (SC). The score for each document in FLONET + SC is computed as follows,

$$\begin{aligned} p_\eta^{str}(z|\mathbf{h}_i) &\approx p(z|\mathbf{h}_{i-1}) \\ &= \sum_{s\in\mathcal{S}} p(z|s).p(s|\mathbf{h}_{i-1}) \\ &= \sum_{s\in\mathcal{S}} p(z|s).b_{i-1}(s) \end{aligned} \tag{5}$$

We approximate $p_\eta^{str}(z|\mathbf{h}_i)$ to $p(z|\mathbf{h}_{i-1})$ as we only use the previous belief computed using h_{i-1}. $p(z|s)$ captures the probability of document z for response generation given a state s and is computed as follows:

$$p(z|s) = \begin{cases} \alpha/|C_s| & \text{if } z \in C_s \\ \beta/N_\mathcal{Q} & \text{if } z \in \mathcal{Q} \\ \frac{1-\alpha-\beta}{N_\mathcal{F}-|C_s|} & \text{otherwise} \end{cases} \tag{6}$$

where α and β are hyper-parameters, C_s is the set of states that are associated with the children of the node underlying the state s, $N_\mathcal{Q}$ and $N_\mathcal{F}$ are the number of FAQs and number of nodes in the flowchart respectively. $p(z|s)$ encodes the following knowledge: (1) the dialog typically moves from a node to one of its children with a high probability, (2) the dialog can stay in the same node when the user asks for a clarification and the agent refers to an FAQ to answer it, and (3) the dialog can randomly jump to any other nodes in the flowchart with a very low probability.

5 Experimental Setup

Dataset: We perform our experiments on the FLODIAL dataset. It has 2,738 dialogs grounded on 12 different flowcharts from car and laptop troubleshooting domains. Each dialog in FLODIAL is grounded on a flowchart and a corpus of FAQs. The dataset has two different splits: Seen Flowcharts (*S-Flo*) split and Unseen Flowcharts (*U-Flo*) split. The test dialogs in *S-Flo* split are grounded on flowcharts seen during train time and this split is used for evaluating the task of flowchart grounded response generation (FGRG). The test dialogs in *U-Flo* are grounded on flowcharts unseen during train, and this split is used for evaluating zero-shot flowchart grounded response generation (ZS-FGRG).

Evaluation Metrics: We evaluate SA-FLONET and other baselines based on their ability to generate valid responses that are grounded on the flowchart or FAQs. We use *BLEU* (Papineni et al., 2002) and *perplexity* to evaluate generation performance. As we have the labels for the documents over which the responses are grounded, we measure the performance of the retriever using the standard recall@1 (R@1) and a task-specific metric called *success rate* (SR) (Raghu et al., 2021). Success rate measures the fraction of the test dialogs for which the system retrieved the correct document for all the agent utterances in the dialog.

We also perform a human evaluation study on two dimensions: (1) *relevance* - ability to generate relevant responses for the given dialog context and the flowchart, and (2) *grammar* - ability to generate grammatically correct response. The human judges were asked to score the responses on a Likert scale (0-4) (Likert, 1932).

Dialog Evaluation and Belief Propagation: During test, dialogs are first broken down into context-response (CR) pairs. Each CR pair is then used for dialog evaluation. This ensures that the error made by the system at any turn doesn't propagate to predictions at future turns. As SA-FLONET uses the system prediction at turn i to update its belief, it is prone to error propagation to future turns. So, to ensure even comparison, rather than using the system output, we use the gold response to update belief at turn i as follows:

$$b_i(s) = p(s|\mathbf{h}_i, \mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{h}_i, s)}{\Sigma_{s'} p(\mathbf{y}|\mathbf{h}_i, s')} \tag{7}$$

where $p(\mathbf{y}|\mathbf{h}_i, s)$ is computed using the pre-trained RAG generator. We use Equation 7 to update the $b_i(s)$ rather than Equation 3.

Implementation Details: SA-FLONET is implemented in PyTorch (Paszke et al., 2019). Hyper-parameters such as learning rates, dropout and embedding sizes we use the best values reported by Raghu et al. (2021). For SemBERT, we use the best values reported by Zhang et al. (2020). We sampled α, β from increments of 0.1 between [0, 1] and identify the best values based on the performance on the validation sets. For more details, please refer to Appendix A

6 Experimental Results

Our experiments answer three research questions:

1. How does the performance of SA-FLONET compare with existing approaches? (Section 6.1)
2. What is the performance gain from each novel contribution in SA-FLONET? (Section 6.2)
3. What is the importance of content based scorer in SA-FLONET? (Section 6.3)

6.1 Performance Analysis

We compare the performance of our proposed model SA-FLONET with FLONET. We also report numbers on a variant of FLONETcalled TF-IDF + GPT2 in which the retriever is replaced by a simple TF-IDF retriever. Table 1 reports the response generation performance on the two splits: S-Flo (FGRG) and U-Flo (ZS-FGRG). SA-FLONET achieves a 1.28 point improvement in BLEU on the S-Flo setting and an almost 4 point improvement in BLEU on the U-Flo setting. Given that SA-FLONET is built on top of FLONET, we attribute this improvement entirely to the novel structure based scorer which infuses structural constraints and NLI into the retrieval process. Moreover, we find that the overall improvement is better in U-Flo than in S-Flo as SA-FLONET can memorize the structure of the flowcharts seen during train. Hence, the S-Flo setting does not gain much compared to U-Flo by incorporating the structural constraints.

Model	S-Flo		U-Flo	
	BLEU	PPL	BLEU	PPL
TF-IDF + GPT2	7.90	13.28	6.90	18.53
FLONET	19.89	4.17	14.83	5.67
SA-FLONET	**21.17**	**4.10**	**18.81**	**5.08**

Table 1: Next response prediction performance.

Model	S-Flo		U-Flo	
	R@1	SR	R@1	SR
TF-IDF + GPT2	0.304	0.002	0.373	0.004
FLONET	0.793	0.318	0.677	0.133
SA-FLONET	**0.878**	**0.535**	**0.819**	**0.297**

Table 2: Retriever performance of various models.

Table 2 reports the retriever performance of the models in both settings. We find that SA-FLONET achieves an increase in retriever performance across settings compared to FLONET. In the S-Flo, we can observe an 8 point increase in R@1 for SA-FLONET compared to FLONET. This leads to a more than 20 point improvement in success rate (SR) with SA-FLONET being able to perfectly ground its responses in more than 50% of dialogs. The improvement is larger in the U-Flo setting with SA-FLONET gaining almost 15 points over FLONET in R@1 which consequently doubles the SR. Interestingly, SA-FLONET has narrowed the gap in R@1 between S-Flo and U-Flo compared to FLONET by infusing structural constraints.

Human Evaluation: We collected judgements on 75 randomly sampled context-response pairs each from the S-Flo and U-Flo splits. We collected two sets of judgements on the responses generated by FLONET and SA-FLONET. Table 3 re-

Model	S-Flo		U-Flo	
	Rel.	Gra.	Rel.	Gra.
FLONET	3.42	3.79	2.36	3.48
SA-FLONET	3.49	3.64	2.54	3.59

Table 3: Human evaluation of FLONET and SA-FLONET on the both S-Flo and U-Flo splits.

ports the human evaluation results. We see that the responses generated by SA-FLONET are more relevant than the ones by FLONET. We measure the inter-annotator agreement using Cohen's kappa (Cohen, 1960). The agreement was moderate on relevance (0.54) and fair on grammar (0.35).

6.2 Ablation Study

We assess the value of each model component, by adding them one at a time to FLONET. Table 4 reports both the response generation and retrieval metrics for various configurations on both the data splits. We represent the model with just structural constraints as FLONET + SC (as described in Section 4.4. SA-FLONET uses both structural constraints and natural language inference.

We define four error classes to analyse the performance of each model configuration. Table 5 shows the errors made by the retrievers of various model configurations on the validation sets. (1) *Sibling error* happens when the retriever assigns the highest probability to the sibling of the correct node. This indicates the system failed to map the last user utterance to the correct polarity (or choice) in the flowchart. (2) *Random jumps* happen when the system fails to capture the structural constraints and predicted a node that is neither the correct node nor its siblings. (3) *FAQ* errors occur when the gold response is grounded on a particular FAQ and the retriever fails to assign the highest probability to it. (4) *First utterance* errors: When the dialog starts, agents may skip a few nodes in the flowchart based on the information already present in the first utterance. This error happens when the system fails to skip a few nodes along the path in the flowchart to land on the correct node.

Adding structural constraints to FLONET improves the performance on U-Flo, but deteriorates on S-Flo. As FLONET memorizes the connectivity structure in the S-Flo setting, it has less scope for improvement. This is supported by the total number of random jump errors made due to the lack of structural awareness in Table 5. In S-Flo, the number of random jumps are quite low to begin with. In U-Flo, adding structural constraints reduces the random jumps errors from 376 to 152.

Adding NLI is expected to improve the understanding of indirect user response to agent's polar questions. In Table 5, we see that both S-Flo and U-Flo have issues with understanding user response to polar questions as their sibling errors are high. We can see that adding the NLI component reduces the sibling errors in both settings and improves the response generation performance.

6.3 Importance of Content Based Scorer

The SA-FLONET retriever has two modules: the content based scorer (CBS) and the structure aware scorer (SAS). To investigate the necessity of the CBS, we study the performance on a variant, SA-FLONET w/o CBS, in which the CBS is completely removed from the retriever. Table 4 and Table 5 shows the performance of this variant and errors made by it respectively. We find that the performance of SA-FLONET w/o CBS drops below our baseline FLONET. This severe drop is due to two main reasons. Firstly, SAS always assigns equal probability to each FAQ and it was the CBS that scored the FAQs based on the ability to generate responses. Thus when no CBS is used, the system fails to identify the correct FAQ. This is evident by the increase in FAQ errors made by SA-FLONET w/o CBS compared FLONET on both S-Flo and U-Flo settings.

Secondly, as the SAS discourages the retriever to jump to any non adjacent nodes in the flowchart, the systems fails in scenarios where the agent skips a few nodes based on the information already specified by the user. The increase in the number of first utterance errors, which arise due to incorrect grounding when the agent fails to skip a few flowchart nodes, shows that the system fails when the next utterance is grounded on non-adjacent node in the flowchart. Thus, when CBS and SAS are used together, CBS can help overcome the structural constraints imposed by SAS as and when needed. We conclude that both content based scorer and the structure aware scorer are necessary to improve document ranking and response generation.

7 Discussion

We illustrate the benefit of the structure aware scorer (SAS) using a qualitative example shown in Figure 3. The figure visualizes how SAS ensures

Model	S-Flo				U-Flo			
	R@1	SR	BLEU	PPL	R@1	SR	BLEU	PPL
FLONET	0.793	0.318	19.89	4.17	0.677	0.133	14.83	5.67
FLONET + SC	0.781	0.270	19.16	4.26	0.706	0.151	16.56	5.42
SA-FLONET	**0.878**	**0.535**	**21.17**	**4.10**	**0.819**	**0.297**	**18.81**	**5.08**
SA-FLONET w/o CBS	0.613	0.026	16.23	4.81	0.644	0.000	15.13	5.92

Table 4: Ablation study: impact of each model component in SA-FLONET.

Model	S-Flo				U-Flo			
	Sibling	Random Jump	FAQ	First Utterance	Sibling	Random Jump	FAQ	First Utterance
FLONET	178	21	105	68	193	376	391	140
FLONET + SC	195	12	120	74	244	152	341	138
SA-FLONET	43	22	91	66	145	124	375	148
SA-FLONET w/o CBS	35	224	320	354	148	143	492	350

Table 5: Counts of errors made by the retrievers of various models on the validation set.

the retriever follows a path in the flowchart rather than making random jumps across the flowchart. The figure shows the input dialog context along with the gold response, the part of the flowchart necessary for explaining the benefit of SAS, and components of SA-FLONET responsible for ranking the documents. We represent each component using the probabilities of states/documents computed based on the input dialog context.

In the given dialog context, the agent utterance a_1 is grounded on the flowchart node z_2. The belief tracker should ideally assign a probability close to 1 to the state s_2 corresponding to the node z_2. However, it has only a weak belief of around 0.16 on this state. Once the user responds with u_2, we expect it to be mapped to the NO child of z_2 and and hence the retriever should assign the highest probability to the node z_3. We use a simplified representation of the NLI model with just the document distribution conditioned on s_2 and u_2, as we need just the true belief (s_2) to explain the benefit of SAS. The NLI module assigns a high probability (0.97) to z_3 indicating that it confidently understands that the user is saying NO given s_2. The beliefs and NLI scores are combined according to equation 2 to compute the structure aware scores.

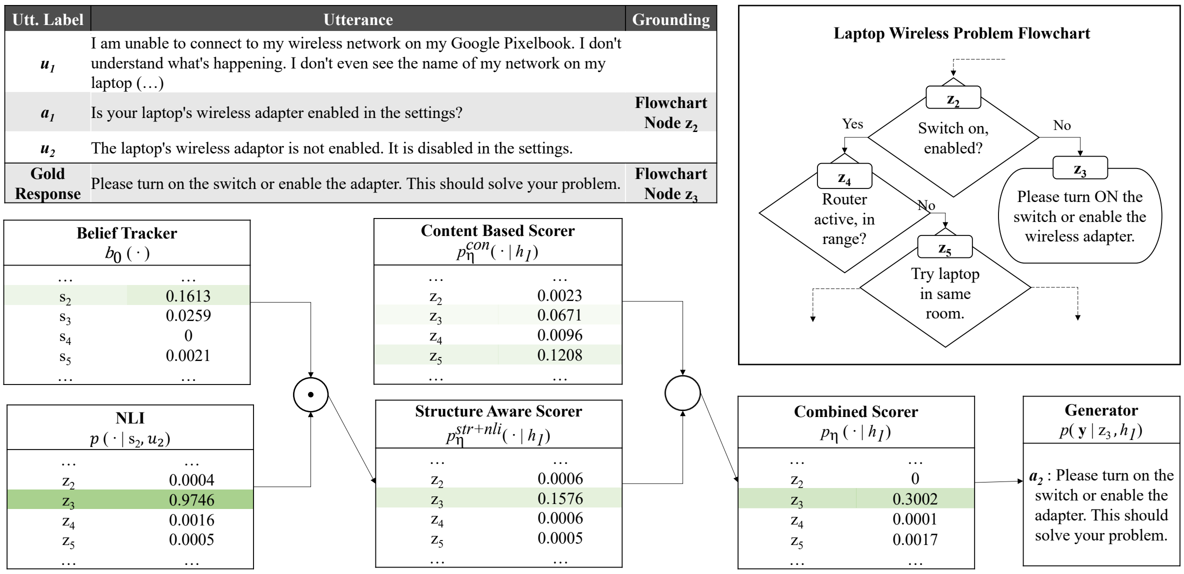
The content based scorer fails to assign a high probability on the correct document z_3 and incorrectly assigns the highest probability to node z_5. When the two scores are combined to compute the overall retriever score, we see that node z_3 receives the highest score and the response is grounded correctly. The example clearly shows how the two scorers have to work together to identify the correct document. Moreover, it shows the ability of the NLI module to correct the system even when both the content based scorer and the belief tracker do not provide strong signals.

8 Conclusion

We propose SA-FLONET for learning flowchart grounded dialogs in an end-to-end manner. SA-FLONET augments the RAG retriever with memory and incorporates flowchart structural constraints into the retrieval process. It uses NLI to better understand indirect user response to polar agent questions. SA-FLONET achieves the state-of-the-art results on both the seen flowchart and the unseen flowchart split of the FLODIAL dataset. It outperforms existing approaches by 15 or more points on success rate. Human evaluations show that SA-FLONET responses are more relevant than the previous state-of-the-art.

Limitations

SA-FLONET has the following limitations: (1) In the current form, NLI can only decide between YES and NO answers to polar questions. It does not apply to non-polar questions (e.g., do you use Windows or Ubuntu?) (2) The content based scorer of SA-FLONET is quite weak. This is evident by the number of FAQ errors and first utterance errors the system makes. Structural constraints are limited in their ability to correct these errors. (3) Human evaluations indicate that the grammar of



Utt. Label	Utterance	Grounding
u_1	I am unable to connect to my wireless network on my Google Pixelbook. I don't understand what's happening. I don't even see the name of my network on my laptop (…)	
a_1	Is your laptop's wireless adapter enabled in the settings?	**Flowchart Node z_2**
u_2	The laptop's wireless adaptor is not enabled. It is disabled in the settings.	
Gold Response	Please turn on the switch or enable the adapter. This should solve your problem.	**Flowchart Node z_3**

Belief Tracker $b_0(\cdot)$	
…	…
s_2	0.1613
s_3	0.0259
s_4	0
s_5	0.0021
…	…

NLI $p(\cdot \mid s_2, u_2)$	
…	…
z_2	0.0004
z_3	0.9746
z_4	0.0016
z_5	0.0005
…	…

Content Based Scorer $p_\eta^{con}(\cdot \mid h_1)$	
…	…
z_2	0.0023
z_3	0.0671
z_4	0.0096
z_5	0.1208
…	…

Structure Aware Scorer $p_\eta^{str+nli}(\cdot \mid h_1)$	
…	…
z_2	0.0006
z_3	0.1576
z_4	0.0006
z_5	0.0005
…	…

Combined Scorer $p_\eta(\cdot \mid h_1)$	
…	…
z_2	0
z_3	0.3002
z_4	0.0001
z_5	0.0017
…	…

Generator $p(\mathbf{y} \mid z_3, h_1)$

a_2 : Please turn on the switch or enable the adapter. This should solve your problem.

Laptop Wireless Problem Flowchart
z2
Switch on, enabled?
Yes
No
z4
Router active, in range?
No
z5
Try laptop in same room.
z3
Please turn ON the switch or enable the wireless adapter.

Figure 3: An example which shows how SA-FLONET uses NLI to ground on the correct node even when the belief tracker and content based scorer provide poor signals.

the response generated by SA-FLONET is slightly worse than the baseline in S-Flo setting.

Acknowledgments

This work is supported by IBM AI Horizons Network grant, grants by Google, Verisk, and 1MG, an IBM SUR award, and the Jai Gupta chair fellowship by IIT Delhi. We also thank the IIT Delhi HPC facility for its computational resources.

References

Antoine Bordes and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *International Conference on Learning Representations*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology*.

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

Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2018. Mem2Seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1468–1478, Melbourne, Australia. Association for Computational Linguistics.

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

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. *Advances in neural information processing systems*, 32.

Libo Qin, Xiao Xu, Wanxiang Che, Yue Zhang, and Ting Liu. 2020. Dynamic fusion network for multi-domain end-to-end task-oriented dialog. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6344–6354, Online. Association for Computational Linguistics.

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

Dinesh Raghu, Shantanu Agarwal, Sachindra Joshi, and Mausam. 2021. End-to-end learning of flowchart grounded task-oriented dialogs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4348–4366, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haoyu Song, Wei-Nan Zhang, Jingwen Hu, and Ting Liu. 2020. Generating persona consistent dialogues by exploiting natural language inference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8878–8885.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 553–562, New York, NY, USA. Association for Computing Machinery.

David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2021. Efficient Retrieval Augmented Generation from Unstructured Knowledge for Task-Oriented Dialog. In *Workshop on DSTC9, AAAI*.

Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. 2019. Dialogue natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3731–3741, Florence, Italy. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Jing Xu, Arthur Szlam, and Jason Weston. 2022. Beyond goldfish memory: Long-term open-domain conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5180–5197, Dublin, Ireland. Association for Computational Linguistics.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020. Semantics-aware BERT for language understanding. In *the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-2020)*.

A Training Details

All experiments were run on a single Nvidia V100 GPU with 32GB of memory. The S-Flo retriever, U-Flo retriever, SemBERT and generator have 3M, 23M, 340M and 117M trainable parameters respectively. Thus, SA-FLONET has a total of 460M trainable parameters for S-Flo and 480M trainable parameters for U-Flo. SA-FLONET has an average runtime of approximately 32 hours (220 mins per epoch) for both S-Flo and U-Flo settings. As the content based scorer and the generator in SA-FLONETis same the as in FLONET, we initialized their weights using the best performing model weights of FLONET. The best performing FLONET + SC model on S-Flo and U-Flo uses (0.1, 0,5) and (0.3, 0.1) for (α, β) respectively. We use BLEU as our early stop criterion. We ran each of these configurations twice. We report numbers on SA-FLONET and two of its variants in the ablation. Table 6 reports the best validation BLEU achieved by each model.

Model	S-Flo	U-Flo
SA-FLONET	22.38	14.04
FLONET + SC	20.91	13.16
SA-FLONET w/o CBS	16.32	10.75

Table 6: Validation performance of various models

B NLI Finetuning

We first discuss some details of the distantly supervised data used for fine-tuning SemBERT. We then briefly discuss the Circa dataset.

B.1 Distantly Supervised Data

For each setting, the data for finetuning SemBERT is collected in a distantly supervised manner using the intermediate document ranking generated by FLONET. Table 7 shows the number of data points constructed. Table 8 shows examples of (premise,hypothesis,label) tuple collected using distant supervision. As FLODIALcontains the annotation for the gold document to which each response is grounded on, we compute and report the accuracy of the distantly supervised data in Table 9.

B.2 Circa Data

We map the "Yes", "No" and "In the middle, neither yes nor no" from the RELAXED scheme from (Louis et al., 2020) to entailment, contradiction and neutral respectively. The entailment, contradiction and neutral classes had 12833, 16628 and 949 examples respectively. We randomly sampled 60% to create the train set and 20% for the validation.

C Qualitative Results

Table 10 compares the responses generated by SA-FLONET and FLONET. The example showcases the ability of SA-FLONET to handle indirect response to polar questions using NLI. It can be seen that even with no explicit polar words (such as yes, no, doesn't, and won't), SA-FLONET is able to map the user response to the correct polarity.

Dataset	S-Flo			U-Flo		
	Entailment	**Contradiction**	**Neutral**	**Entailment**	**Contradiction**	**Neutral**
Train	2868	3822	1370	2545	3579	1321
Validation	636	734	288	532	403	581

Table 7: Number of samples in each class (entailment, contradiction and neutral) constructed using distant supervision.

Premise & Hypothesis	Distantly Supervised Label	Gold Label
P: Did you recently change your car tires? H: I have not changed my car tires recently.	Contradiction	Contradiction
P: Is the network router active and in range? H: Yes, it is. It's both of those things.	Entailment	Entailment
P: Does the router have default settings? H: Where do I check that?	Neutral	Neutral
P: Is the brake fluid level OK? H: The brake fluid is too low.	Entailment	Contradiction
P: Does one or more wheels drag too much? H: The wheel on the car turn with ease.	Entailment	Contradiction
P: Does any of your tires have a loose hub cap? H: Some of my hub caps are loose.	Contradiction	Entailment

Table 8: Some examples of (premise, hypothesis, label) tuples constructed by distant supervision along with the ground truth label. Distant supervision gets the first three examples right and the latter three wrong.

Dataset	S-Flo			U-Flo		
	Entailment	**Contradiction**	**Neutral**	**Entailment**	**Contradiction**	**Neutral**
Train	0.921	0.965	0.902	0.875	0.926	0.890
Validation	0.735	0.873	0.843	0.667	0.866	0.884

Table 9: Class-wise accuracy of the distantly supervised data.

Speaker	Utterance	Grounding
User	I've had my Cadillac Escalade ESV for 3 years. I've noticed that the brakes aren't working and this is a issue that seems to come up a lot with this car nowadays. Please help me fix this.	
Agent	Once you activate the brakes, does the car come to a stop?	flowchart-node-1
User	What do I do now if they aren't working the same as before?	
Agent	If needed, make sure to apply more pressure than normal to the brake pedal for the brakes to work correctly. If this works then it means the brakes are still able to halt the car.	faq-0
User	The car keeps going when I activate the brakes.	
Gold Response	When you hit the brake pedal, does it go all the way to the floor?	flowchart-node-2
SA-FloNet Response	Does the brake pedal go all the way to the floor?	flowchart-node-2
FloNet Response	Is the emergency parking brake functioning as it should?	flowchart-node-9

Table 10: An example dialog from FloDialwith predictions from SA-FloNet and FloNet. flowchart-node-1 is the parent of flowchart-node-2 and flowchart-node-9. flowchart-node-1 is connected to flowchart-node-2 and flowchart-node-3 by a NO and YES link respectively.