Getting Better Dialogue Context for Knowledge Identification by Leveraging Document-level Topic Shift

Nhat Tran and **Diane Litman**
University of Pittsburgh
nlt26@pitt.edu, dlitman@pitt.edu

Abstract

To build a goal-oriented dialogue system that can generate responses given a knowledge base, identifying the relevant pieces of information to be grounded in is vital. When the number of documents in the knowledge base is large, retrieval approaches are typically used to identify the top relevant documents. However, most prior work simply uses an entire dialogue history to guide retrieval, rather than exploiting a dialogue's topical structure. In this work, we examine the importance of building the proper contextualized dialogue history when document-level topic shifts are present. Our results suggest that excluding irrelevant turns from the dialogue history (e.g., excluding turns not grounded in the same document as the current turn) leads to better retrieval results. We also propose a cascading approach utilizing the topical nature of a knowledge-grounded conversation to further manipulate the dialogue history used as input to the retrieval models.

1 Introduction

Knowledge identification (KI) is the task of identifying relevant information from a database of documents that should be used when generating responses in a *knowledge-grounded dialogue system* (Feng et al., 2020; Wu et al., 2021). When the number of documents is large, information retrieval is typically used to find relevant documents (Karpukhin et al., 2020; Khattab and Zaharia, 2020; Yu et al., 2021). Most approaches encode both the knowledge sources and the dialogue context (i.e., all prior turns), which is later used as an input query, into the same vector space. Since the quality of the input query significantly impacts the retrieval results (Yu et al., 2020, 2021), using an optimal dialogue context is crucial.

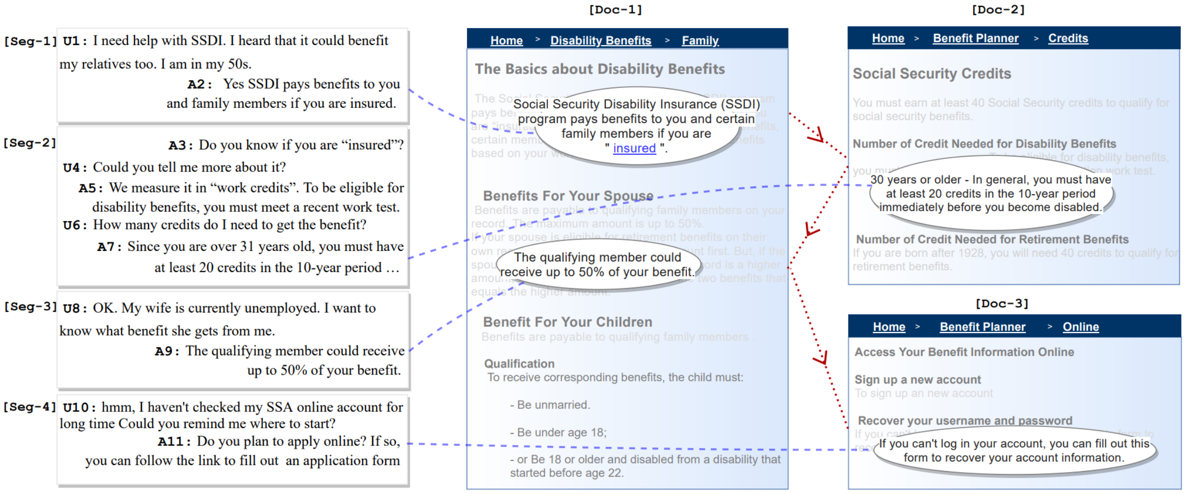
In knowledge-grounded dialogues, each turn can be grounded in a different document. Blindly including all previous turns into the dialogue context can introduce unnecessary noise because a turn grounded in a different document can provide redundancy or irrelevant information for the grounding process of the current turn. Our **hypothesis** is that including only turns in the dialogue context that are grounded in the same document as the current turn when creating a retrieval query will improve KI task performance. To test this hypothesis, we tried several approaches to select relevant turns to be included in the dialogue context. Specifically, we vary the input to a previously used predictive model (Lewis et al., 2020b) to see whether querying using only turns grounded in the same document as the current turn improves retrieval performance. After verifying our hypothesis using oracle results, we utilize automatically computed *document-level topic shifts* to improve the dialogue context used for KI. Even with imperfect automatic predictive models, our initial results show that improving dialogue context increases the retrieval results on dialogues grounded on at least 2 documents. Further analysis on errors from dialogues grounded only in 1 document leads us to a simple heuristic that raises the retrieval accuracy for the entire dataset.

Our contribution is twofold. First, we verify the importance of a proper contextualized query in the KI task, as excluding utterances from the dialogue context that are not grounded in the same document as the current turn leads to better knowledge retrieval results in an oracle condition. Second, based on that verification, we develop a simple automatic approach that improves KI in document-grounded dialogue by leveraging a proposed topic segmentation algorithm that uses both dialogue content and grounding documents.

2 Related Work

Our work is related to recent work in **knowledge identification (KI) in knowledge-grounded dialogues** (Choi et al., 2018; Dinan et al., 2019; Qu et al., 2020; Feng et al., 2020; Campos et al., 2020; Wu et al., 2021). However, prior work has largely

Proceedings of the SIGdial 2022 Conference, pages 368–375
Heriot-Watt University, Edinburgh, UK. 07-09, September, 2022 ©2022 Association for Computational Linguistics



[Seg-1] U1: I need help with SSDI. I heard that it could benefit my relatives too. I am in my 50s.

A2: Yes SSDI pays benefits to you and family members if you are insured.

[Seg-2] A3: Do you know if you are "insured"?

U4: Could you tell me more about it?

A5: We measure it in "work credits". To be eligible for disability benefits, you must meet a recent work test.

U6: How many credits do I need to get the benefit?

A7: Since you are over 31 years old, you must have at least 20 credits in the 10-year period …

[Seg-3] U8: OK. My wife is currently unemployed. I want to know what benefit she gets from me.

A9: The qualifying member could receive up to 50% of your benefit.

[Seg-4] U10: hmm, I haven't checked my SSA online account for long time Could you remind me where to start?

A11: Do you plan to apply online? If so, you can follow the link to fill out an application form

[Doc-1]

Home > Disability Benefits > Family

The Basics about Disability Benefits

Social Security Disability Insurance (SSDI) program pays benefits to you and certain family members if you are " insured ".

Benefits For Your Spouse

Benefits are payable to qualifying family members on your record. The maximum amount is up to 50%.

The qualifying member could receive up to 50% of your benefit.

Benefit For Your Children

Benefits are payable to qualifying family members .

Qualification

To receive corresponding benefits, the child must:

- Be unmarried.
- Be under age 18;
- or Be 18 or older and disabled from a disability that started before age 22.

[Doc-2]

Home > Benefit Planner > Credits

Social Security Credits

You must earn at least 40 Social Security credits to qualify for social security benefits.

Number of Credit Needed for Disability Benefits

30 years or older - In general, you must have at least 20 credits in the 10-year period immediately before you become disabled.

Number of Credit Needed for Retirement Benefits

If you are born after 1928, you will need 40 credits to qualify for retirement benefits.

[Doc-3]

Home > Benefit Planner > Online

Access Your Benefit Information Online

Sign up a new account

To sign up an new account

Recover your username and password

If you can't log in your account, you can fill out this form to recover your account information.

Figure 1: An example dialogue from MultiDoc2Dial (Feng et al., 2021) that is grounded in three different documents.

treated KI as reading comprehension since all turns in a conversation were typically grounded in one document. In a dataset such as MultiDoc2Dial (Feng et al., 2021), a reading comprehension approach is less computationally feasible due to the surge in the number of grounding documents. We thus approach KI as information retrieval following Dalton et al. (2020) and Yu et al. (2021). However, in those studies, turns were again closely related to the same topic, so a full dialogue context was typically used for the query. We instead use predicted document-level topic shifts as the basis of a simple discourse-informed query approach, yielding improved results for KI in MultiDoc2Dial.

Our focus on document-level topic shifts in dialogue is related to the task of **discourse segmentation**. Prior work in identifying topic changes has used topic tracking with predefined topics (Soleimani and Miller, 2016; Takanobu et al., 2018) and used coherence scores between consecutive utterances to split the conversation into smaller topics (Xu et al., 2021; Xing and Carenini, 2021). However, such segmentation approaches have typically been based solely on the content of conversations. In contrast, we propose a topic segmentation approach based not only on the dialogue content, but also on the grounding document.

3 Task and Dataset

3.1 Knowledge Identification Task

We follow the definition of *knowledge identification (KI)* from Feng et al. (2021): given the current user turn, dialogue context, and the entire set of documents from the same domain, find the grounding text span from one document that the next agent response needs to refer to.

3.2 Dataset

We use **MultiDoc2Dial** (Feng et al., 2021) as our dataset. It consists of 4796 information-seeking conversations grounded in 488 documents from 4 domains (only one domain per dialogue). 948 of them are grounded in only 1 document.

This dataset suits our study as the full dialogue context of a turn may span multiple topics. Figure 1 shows a dialogue in the corpus that contains four segments and is grounded in three different documents. A *segment* signals that all turns within it are grounded in the same document and the boundary between two segments indicates a *topic* shift. The presence of such document-level topic shifts can make a turn more contextually distant from the previous turn (Arguello and Rosé, 2006). In Figure 1, Seg-3 requires knowledge about "spouse" from Doc-1, but that information is unimportant for the query about "SSA online account" in Seg-4. Including U8 and A9 in the dialogue context when asking about "SSA online account" is not useful and can even add noise to the retrieval query.

4 Background

Passages as Retrieval Units. Since a grounding document can be very long, we split each one into passages and use them as the units for retrieval. We follow Feng et al. (2021) to split a document based on its original paragraphs indicated by markup tags and then attach the hierarchical titles from their html source to each paragraph as a *passage*.

Dense Passage Retrieval (DPR). DPR (Karpukhin

Algorithm 1 Cascading algorithm to get the top 10 passages for a N-turn dialogue Dial = $\{t_1, t_2, ..., t_N\}$

```
procedure FindTopK($Dial$)
    $DOCS$ = { }        ▷ List of documents have been used for grounding so far, empty in the beginning
    for $i = 1$ to $N$ do
        for $j = 1$ to len($DOCS$) do
            $h_j$ = concatenation of turns $t_k$ where $k < i$ and $ground[k] = j$
            for each passage $p_x$ in $DOCS[j]$ do        ▷ $DOCS[j] = \{p_1, p_2, ..., p_m\}$
                $Score[p_x] = PC(t_i, h_j, p_x)$
                if $Score[p_x] > Best\_score[j]$ then
                    $Best\_passage[j] = p_x$
                    $Best\_score[j] = Score[p_x]$
        $Best\_doc = \arg\max_d Best\_score[d]$
        if $Best\_score[Best\_doc] < 0.5$ then        ▷ No old documents can be used for grounding
            Use DPR with only the current turn $t_i$ as the query to retrieve the top 10 passages $TOP\_10[i]$
            $ground[i]$ = The document containing the highest-score passage from $TOP\_10[i]$
        else
            $ground[i] = Best\_doc$        ▷ Choose the highest-score passage for grounding
            $TOP\_10[i]$ includes:
              • The passage with the highest score: $Best\_passage[Best\_doc]$
              • Top 3 other passages from $Best\_doc$
              • Up to top 3 other passages with the highest $Score$ this turn
              • Remaining non-duplicate passages from the entire database retrieved by DPR, using
                only the previous turns of $t_i$ grounded on Best_doc for the dialogue history
            Add documents contains passages from $TOP\_10[i]$ to $DOCS$

    return $TOP\_10[N]$
```

et al., 2020) is an approach to quickly find the top k passages relevant to a given input query from a big database. DPR uses two BERT encoders (Devlin et al., 2019), one to index all passages to *d*-dimensional vectors, and one to map the input query to the same *d*-dimensional vector space. Because the similarity between a query and a passage is defined as the dot product of their vectors, retrieving the top k passages at inference time can be done efficiently when the encoded passages are indexed offline by FAISS (Johnson et al., 2021). The *input query* in our task is the concatenation of the current user turn and the dialogue context.

Retrieval-Augmented Generation (RAG). RAG (Lewis et al., 2020b) is our base response generation model. It consists of a *retriever* module (DPR) and a *generator* module (BART, Lewis et al., 2020a). The retriever gets the most relevant passages given the input query, and the generator takes the query and top-k passages as input to generate the response as output. In our task, the target response is the grounding span, that is, the specific piece of information used to ground the response for the current user turn (see ovals in Figure 1).

5 Method: Document-level Topic Shift

Since KI methods typically use all previous turns as the dialogue context, instead of focusing on improving model architectures for knowledge-grounded response generation, we examine whether varying the ***input*** (e.g., dialogue context) to such models improves the retrieval and generation results. Specifically, we hypothesise that for the current turn t_i, including only previous turns grounded in the same document as t_i in the dialogue context to DPR will improve the overall passage retrieval results. To verify this hypothesis, we first create an oracle model called **RAG-oracle**. It assumes that the correct grounding passages of previous turns are known, so it only uses the turns grounded in the same document as t_i in the input query to DPR.

However, since the gold-standard grounding information of the dialogue is not available in real use cases, we build a simple classification model to estimate it. This model, which we call the **Passage Checking Model (PC)**, is a BERT model fine-tuned on Multidoc2Dial. The input includes the current user turn t_i, the dialogue context h, and one passage p. The output is 1 if t_i should be grounded in p given h and 0 otherwise. During training, the dialogue context only contains turns grounded in the same document as t_i. For each training instance, we sample 128 negative passages[1], at most half of them are from the same document which p belongs to and the rest are from different documents. Our PC model achieved 69.4 F_1 score on validation set. We also use the probability scores from the last layer (softmax) as a confidence measure below.

Next, we use PC in a cascading algorithm to retrieve the top 10 passages for the current user turn (details in Algorithm 1). For each conversation, we process the turns increasingly while keeping track of a list of documents (*DOCS*) that have been used for grounding so far. At each turn t_i, we try to ground it to each document in *DOCS* and use only turns grounded in the same targeting document as the dialogue context. We add the documents containing one of the top-10 passages to the set *DOCS* before going to the next turn. The model based on this algorithm is called **RAG-cascade**.

Finally, since the BART generator relies on the top-5 passages to provide the grounding span, having a better top-5 can yield improved generation results. We explore this idea by reusing the probability scores from the PC model as a **ranking** metric instead of building another ranking model.

6 Experiments and Results

Following Feng et al. (2021), all numbers reported in this section are the mean of three runs with different random seeds. For retrieval, we use recall at k (R@k), which measures the frequency of the correct passage found in the top-k retrieved passages. Token-level F_1 score and Exact Match (EM) (Rajpurkar et al., 2016) are used to evaluate the grounding span generation results. Implementation details can be found in Appendix A.

6.1 Experiment Setup

RAG was the only model used to identify the grounding passage (retriever) and generate the grounding span (generator) in our experiments. We only vary the **input** to the RAG model to demonstrate different approaches to choose the dialogue context (details in Table 1).

6.2 Passage Retrieval Results

We report the passage retrievel results on the entire evaluation data of MultiDoc2Dial (*D*) as well as on a subset of data containing at least two segments (D_2) in Table 2. On *D*, `RAG-oracle` consistently outperforms the RAG baselines. The gap is most noticeable at R@10 (6.4 points). The discrepancy is even bigger on D_2 with more than 7.5 points increases in both R@1 and R@10. These numbers support our hypothesis that only using turns grounded in the same document as the current turn in the dialogue context creates a better contextualized input query for the retriever module (DPR).

While `RAG-cascade` has higher recall on D_2 compared to `RAG-baseline`, they perform similarly (less than 1.3-point differences) on D. This implies that the improvement on data with multiple segments was offset by the degradation in data with only one segment (about 19.7% of D). We believe these errors come from the loss of context from previous turns when our model incorrectly decides to split a one-segment dialogue into multiple segments at some point and this error starts propagating (see Appendix B for an example).

The distribution of incorrect segmentation in one-segment dialogues from validation set shows that about 70% of them occur when more than 6 turns appear in the dialogue context (Appendix C). A naive heuristic of limiting the number of turns in the context to 6, while it does not affect the retrieval performance on D_2, reduces errors on one-segment data, and as a result, increases the overall performance in *D*. This is demonstrated by the fact that `RAG-limit` is superior to `RAG-baseline` and `RAG-cascade` in the full evaluation data.

`RAG-topic` also uses topic segmentation as additional information to create the relevant dialogue context, but it has the worst performances in terms of passage retrieval. This implies that in contrast to our proposed `RAG-cascade` model where the "topic" is identified based on the grounding document, using a document-agnostic approach to do dialogue topic segmentation is ineffective.

Re-ranking does not always improve R@1. The rises in R@5 are clearer, where the largest boosts in *D* and D_2 come from `RAG-oracle` (3.3) and `RAG-cascade` (4.4), respectively. We observe several decreases in recall with re-ranking, but all of them are within 0.8 points. `RAG-oracle` with

[1] The same negative sample size used by Feng et al. (2021).

Model	Dialogue Context used in the Input to RAG
RAG-baseline	All previous turns
RAG-oracle	Turns grounded in the same document as the current turn
RAG-cascade	Turns grounded in the same document as the current turn, predicted by algorithm 1
RAG-limit	Same as RAG-cascade but the maximum number of turns is limited to 6
RAG-topic	Like RAG-oracle but uses a dialogue topic segmentation method (Xing and Carenini, 2021) to decide the thresholds from calculated coherence scores between 2 consecutive utterances while ignoring all grounding documents (in contrast to RAG-cascade)

Table 1: Dialogue context used in the input for the experimented RAG models.

Model	Passage Retrieval						Span Generation	
	All Data (D)			>2 Segments Data (D_2)				
	@1	@5	@10	@1	@5	@10	F_1	EM
RAG (Feng et al., 2021)	49.0	72.3	80.0	n/a	n/a	n/a	41.9	*24.9*
RAG-baseline	48.6	72.5	79.2	40.2	63.5	72.3	41.1	23.8
+ re-ranking	49.0	74.7	79.2	40.1	65.4	72.3	*43.7*	23.4
RAG-oracle	55.1	74.5	**85.6**	**47.9**	69.2	**79.8**	43.1	**25.9**
+ re-ranking	**55.3**	**77.8**	**85.6**	47.5	**73.2**	**79.8**	**43.8**	25.7
RAG-topic	42.1	65.7	71.3	40.2	60.9	70.3	36.2	20.9
+ re-ranking	42.0	67.6	71.3	39.4	62.6	70.3	36.5	20.6
RAG-cascade	48.9	72.8	80.4	44.4	67.2	76.1	41.0	23.7
+ re-ranking	49.7	75.3	80.4	*44.6*	*71.6*	76.1	41.2	23.8
RAG-limit	*52.8*	74.1	*82.3*	44.3	67.0	*76.3*	41.5	23.8
+ re-ranking	52.5	*75.4*	*82.3*	44.3	71.0	*76.3*	41.4	24.0

Table 2: Passage retrieval and span generation results. Best results from MultiDoc2Dial (Feng et al., 2021) are reported in the first row. **Bold** numbers are the best overall results, *underlined* numbers demonstrate the best results besides RAG-oracle. All numbers are statistically significant ($p < 0.05$) compared to RAG-baseline.

re-ranking achieves the best results in all categories, except for R@1 in D_2 where the version without re-ranking shows a 0.4-point lead.

6.3 Span Generation Results

We also report the grounding span generation results. With automation, we see no improvements in F_1 and EM. Even with increases in R@5 from re-ranking, we do not witness much gain in span metrics. Feng et al. (2021) reported a similar pattern where some models perform better in passage retrieval but are inferior in grounding span generation. Our assumption is that passages in top-5 that are not the correct grounding for the current user turn may contain irrelevant or contextually incorrect information for the BART generator.

7 Conclusion

In this work, we showed that exploiting document-level topic shifts in document-grounded dialogues relying on multiple documents as the knowledge base can raise passage retrieval results. We first proposed a simple cascading approach based on a simple BERT model for passage checking and re-ranking that yielded improved retrieval results for multiple-segment dialogues. An error analysis suggested that limiting the number of turns in the dialogue context to 6 reduced the false segmentation errors for the one-segment dialogues and thus improved the scores for the full corpus. Furthermore, no improvement from span generation with the increased retrieval results implied that a general-purpose generative model like RAG might not be a good fit for knowledge identification task in information-seeking dialogues. Future plans include using better generative models to generate better system responses from the identified knowledge and conducting further analysis on the segmentation yielded from the proposed algorithm.

Acknowledgments

We would like to thank Cuong Nguyen and Kien Do for their comments on the initial draft of the paper, and the reviewers for their helpful feedback.

References

Jaime Arguello and Carolyn Rosé. 2006. Topic-segmentation of dialogue. In *Proceedings of the Analyzing Conversations in Text and Speech*, pages 42–49, New York City, New York. Association for Computational Linguistics.

Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Deriu, Mark Cieliebak, and Eneko Agirre. 2020. DoQA - accessing domain-specific FAQs via conversational QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7302–7314, Online. Association for Computational Linguistics.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Jeffrey Dalton, Chenyan Xiong, Vaibhav Kumar, and Jamie Callan. 2020. *CAsT-19: A Dataset for Conversational Information Seeking*, page 1985–1988. Association for Computing Machinery, New York, NY, USA.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Song Feng, Siva Sankalp Patel, Hui Wan, and Sachindra Joshi. 2021. MultiDoc2Dial: Modeling dialogues grounded in multiple documents. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6162–6176, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Song Feng, Hui Wan, Chulaka Gunasekara, Siva Patel, Sachindra Joshi, and Luis Lastras. 2020. doc2dial: A goal-oriented document-grounded dialogue dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8118–8128, Online. Association for Computational Linguistics.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. *ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT*, page 39–48. Association for Computing Machinery, New York, NY, USA.

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

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. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W. Bruce Croft, and Mohit Iyyer. 2020. *Open-Retrieval Conversational Question Answering*, page 539–548. Association for Computing Machinery, New York, NY, USA.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Hossein Soleimani and David J. Miller. 2016. Semi-supervised multi-label topic models for document classification and sentence labeling. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 105–114, New York, NY, USA. Association for Computing Machinery.

Ryuichi Takanobu, Minlie Huang, Zhongzhou Zhao, Fenglin Li, Haiqing Chen, Xiaoyan Zhu, and Liqiang Nie. 2018. A weakly supervised method for topic segmentation and labeling in goal-oriented dialogues via reinforcement learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4403–4410. International Joint Conferences on Artificial Intelligence Organization.

Zeqiu Wu, Bo-Ru Lu, Hannaneh Hajishirzi, and Mari Ostendorf. 2021. DIALKI: Knowledge identification in conversational systems through dialogue-document contextualization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1852–1863, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Linzi Xing and Giuseppe Carenini. 2021. Improving unsupervised dialogue topic segmentation with utterance-pair coherence scoring. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 167–177, Singapore and Online. Association for Computational Linguistics.

Yi Xu, Hai Zhao, and Zhuosheng Zhang. 2021. Topic-aware multi-turn dialogue modeling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14176–14184.

Shi Yu, Jiahua Liu, Jingqin Yang, Chenyan Xiong, Paul Bennett, Jianfeng Gao, and Zhiyuan Liu. 2020. *Few-Shot Generative Conversational Query Rewriting*, page 1933–1936. Association for Computing Machinery, New York, NY, USA.

Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. 2021. *Few-Shot Conversational Dense Retrieval*, page 829–838. Association for Computing Machinery, New York, NY, USA.

A Implementation Details

Since we do not modify the architecture of the RAG models, we adopt the implementation from Feng et al. (2021) [2] and keep all of the hyperparameters unchanged. We also use the same 5:1:1 train/validation/test split. For the implementation of the Passage Checking (PC) model, we use the uncased BERT version with default parameters [3].

B An Example Error in One-segment Document

Table 3 illustrates a case when the prediction errors were propagated in a one-segment document grounded entirely in document **ssa#1**. Here, **ssa#1** refers to the document "How Financial Aid Works | Federal Student Aid#1_0" and **ssa#3** is "Teacher Loan Forgiveness | Federal Student Aid#1_0". At the turn 3, `RAG_cascade` incorrectly predicted the grounding document to **ssa#3**, which is still relevant to "loan", but for teachers instead. Starting from this, the algorithm favors **ssa#3** and omits the presence of "financial aid" from **ssa#1** in the dialogue context.

C Error Distribution in One-Segment Dialogues

Figure 2 illustrates the proportions of errors in relation to the number of turns included in the dialogue history when the entire conversation is grounded in one document.

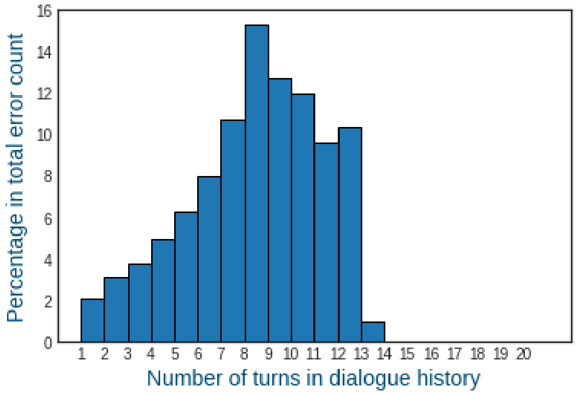


Figure 2: Error distribution in one-segment dialogues.

[2] https://github.com/IBM/multidoc2dial

[3] https://huggingface.co/bert-base-uncased

Turn	Utterance	Predicted Doc
1	Hello, I would like to know who can receive financial aid	ssa#1
1	of course we are here to give you More information	
2	How can I estimate the aid I can access	ssa#1
2	Use FAFSA4caster to get an early estimate of your eligibility for federal student aid.	
3	I also want to about the repayment. And would you recommend that I pay the student loans?	ssa#3
3	As you prepare to graduate, prepare to pay off your student loans. Good news! Federal student loan borrowers have a six-month grace period before payments begin.	
4	and how to determine if I am eligible for help?	ssa#3
4	Your college uses your FAFSA data to determine your eligibility for federal aid.	

Table 3: An example one-segment dialogue where the prediction errors are propagated. User's utterances are in grey.