Graph-Guided Unsupervised Knowledge Identification for Dialogue Agents

Shrinivas Khiste1∗ Tathagata Raha2∗ Milan Aggarwal3 Sumit Bhatia3 Simra Shahid3
1 IIT Kharagpur 2 IIIT Hyderabad 3 Media and Data Science Research Lab, Adobe, India

Abstract
Dialogue systems that can effectively respond to user queries in a conversational style have become ubiquitous. Large Language Models (LLMs) have been extensively used as a key component in such systems owing to their linguistic capabilities and implicit knowledge. However, such models are prone to hallucinate while generating a response that can be detrimental, particularly in applications where accuracy is critical. While many works have attempted to address the hallucination concern by supplementing external knowledge in the input to the LLMs, most of them rely on supervised labels to train the knowledge identification module. Such labels might often not be available or difficult to obtain at scale. To address this, we propose our method RANKING, which leverages the structure of the external document to obtain a ranked subset of relevant sentences in an unsupervised manner that can be used for response generation. We model the dependencies in the form of a graph between the sentences present in the external document and the utterances till the given point in the dialogue. We demonstrate the efficacy of RANKING on a commonly used document-grounded conversation dataset (Doc2Dial) where it is observed that RANKING enables generating better responses than using the entire document.

1 Introduction
Dialogue systems that possess the ability to respond to user queries have become popular. The emergence of Chat-GPT (Ouyang et al., 2022) has further revolutionized the domain by facilitating human-like question-answering. Users now prefer to search for desired information through interactive dialogues, as opposed to retrieving and comprehending lengthy documents on their own.

Despite the remarkable capabilities of existing dialogue systems such as Chat-GPT, the cost, legal and privacy concerns surrounding such models make them unsuitable for general business applications. Additionally, queries posed by the users of a business application may require domain-specific knowledge, which a Large Language Model (LLM) may not be privy to. Further, such models are susceptible to hallucinating their responses (Roller et al., 2021; Marcus, 2020) which can be detrimental in scenarios where accurate responses are critical. To mitigate this, there is a growing interest in incorporating external knowledge while using such LLMs (Kaur et al., 2022).

Numerous works have attempted to leverage the knowledge present in external documents to generate automated dialogue responses. Most approaches rely on supervised labels to train knowledge identification module to retrieve text from an external document that is relevant to the dialogue (Lian et al., 2019; Kim et al., 2020; Zheng et al., 2020; Wu et al., 2021). Obtaining such labels requires manually annotating sentences that should be selected by the model at each step in the dialogue which can be expensive and time-consuming. This creates a need to develop approaches that can identify relevant external knowledge in an unsupervised manner (Huang et al., 2021). Typical unsupervised knowledge selection methods employ semantic similarity between the embedded representations of the dialogue context and the text present in external knowledge (Huang et al., 2021; Dinan et al., 2019). However, this ignores the structure in which information is organized in an external document which has been shown to be useful (Djeddal et al., 2021). Such methods also fail to properly capture the relationship between the information flow in the dialogue utterances and structured information in the external document.

To address this, we propose our method RANKING (Graph-Guided Unsupervised Knowledge Identification for Dialogue Agents) that captures the dependencies between the knowledge sen-

∗Work done as an intern at MDSR, Adobe, India

1Please refer to Appendix A for detailed related work
tences, the user and the agent utterances from the dialogue through a graph by representing them as nodes. The graph also captures information flow in the document based on its structure (such as paragraphs). RANKING learns contextualized representations for each node and provides a ranked list of relevant sentences from external knowledge for response generation. We study different design choices for the underlying graph and observe that RANKING performs better than simple TF-IDF or BERT-based embedding similarity for selecting relevant sentences. Further, it is observed that RANKING helps in precisely selecting relevant information by filtering out irrelevant noisy sentences and performs better than providing the entire document as input for response generation.

2 Our Methodology

Problem Definition: Given a dialogue \( D = \{u_1, a_1, u_2, a_2, ..., u_i\} \) (where \( u_j \) and \( a_j \) represent the user and agent utterances respectively); and a supporting grounding document \( \mathcal{K} = \{p_1, p_2, ..., p_P\} \) comprising of paragraphs, the task of the modeled agent is to generate the response \( a_i \). Here \( p_i = \{s^{i_1}_1, s^{i_2}_2, ..., s^{i_{|p_i|}}_{|p_i|}\}; s^{q}_k \) represents the \( k^{th} \) sentence in the \( q^{th} \) paragraph; \( |p_i| \) represents the total number of sentences in paragraph \( p_i \) and \( P \) represents the number of paragraphs in \( \mathcal{K} \). The model has to select sentences from \( \mathcal{K} \) relevant to \( D \) while generating the next agent response \( a_i \).

Overview of RANKING: Figure 1 depicts the architecture of our method. We construct a Document-Discourse graph \( \mathcal{G} \) by representing each utterance in \( D \) as well as paragraphs and sentences in \( \mathcal{K} \) as nodes. The nodes are connected with edges to capture the semantic dependencies between them. Higher-order relationships between different types of nodes are encapsulated using the Node2Vec algorithm that updates the embeddings of each node by utilizing the underlying graph structure. Subsequently, the similarity between the node representing the most recent user utterance and each knowledge sentence is estimated to rank the sentences in \( \mathcal{K} \). The top-ranked sentences are provided as input in addition to the dialogue history to a response generation module that consists of a pre-trained generative LLM.

2.1 Document-Discourse Graph

We refer to our proposed graph structure as Document-Discourse Graph (Figure 2 in appendix B). It consists of 4 types of nodes: sentence nodes \( (s^q_k) \), paragraph nodes \( (p_i) \), user utterance nodes \( (u_i) \), and agent utterance nodes \( (a_i) \). We first explain the graph structure followed by the intuition behind it. A sentence node in a paragraph is connected with the node representing the next sentence in the paragraph \( (s^q_k \rightarrow s^q_{k+1}) \). A paragraph node is connected to the next paragraph \( (p_i \rightarrow p_{i+1}) \) and also to the sentence nodes in that paragraph \( (s^q_k \leftrightarrow p_i) \). The user and agent utterance nodes are connected to the next utterance in the dialogue \( (u_i \rightarrow a_i, a_i \rightarrow u_{i+1}) \). Utterance nodes are also connected to the paragraph nodes as \( (u_j \rightarrow p_i \rightarrow a_j) \).

The graph nodes and the edges between them model various interactions needed to perform knowledge selection. Paragraph nodes and directed edges between them are important to encapsulate the hierarchical structure and information flow in the document. The flow of information between the dialogue and the document is facilitated through edges from user utterance nodes to paragraph nodes and back to agent utterance nodes. This allows it to ground the user query in knowledge sentences and match them with the agent response. The undirected edges between sentences and paragraphs facilitate the interactions between the local information present within a paragraph and the content in other paragraphs and dialogue utterances. We initialize the representation of each node by processing the corresponding text through a sparse featurizer - TF-IDF. The weight for an edge is computed as cosine similarity between the corresponding node representations. We ablate over different graph structures in the experiments section.

2.2 Learning Node Representations

To obtain the contextual representation of each node based on the graph structure, we use Node2Vec (Grover and Leskovec, 2016), an algorithm used to learn continuous feature representations of graph nodes. Specifically, it treats each node in the random walk sequence as a target node and the surrounding nodes as context nodes. It then uses stochastic gradient descent to maximize the likelihood of predicting the context nodes given the target node. We obtain representation \( V_i = \text{Node2Vec}(\mathcal{G})[i] \) for \( i^{th} \) node in \( \mathcal{G} \).

2.3 Sentence Selection

We select the most relevant knowledge sentences \( (s^q_k) \) based on the most recent user query \( (u_i) \) in the dialogue. This is achieved by computing the
pairwise cosine similarity between each $s_k^q$ with the representations of $u_i$. The top-k sentences ($K_{sel} = \{s_1, s_2, \ldots, s_k\}$) are selected based on this score and provided as additional context to a response generation module. The ordering of the sentences is maintained as seen in the document to avoid disturbing information flow.

### 2.4 Response Generation

A pre-trained generative LLM is fine-tuned to generate the agent response ($a_i$) conditioned on the dialogue context and the selected knowledge sentences. In our case, we use BART as the pre-trained LLM to perform response generation. The input to the generative LLM is provided as follows:

$$X = [CLS]["query"]u_i["usr"]u_1["agt"]a_1["usr"]u_i["grounding"]s_1[SEP]...s_k[SEP]$$

Special tags such as ["query"], ["usr"], ["agt"], ["grounding"] etc. are used to instruct the model about the corresponding segments of the input.

### 3 Experiments

We now discuss the experimental setup to study the efficacy of RANKING and various design choices.

#### 3.1 Dataset

We evaluate our proposed model using a goal-oriented document-grounded dialogue dataset Doc2Dial (Feng, 2021). It contains 3,474 dialogues with 44,149 turns for training and 661 dialogues with 8539 turns for evaluation. It contains multi-turn conversations grounded in relevant documents from four domains for social welfare.

#### 3.2 Implementation Details

For knowledge identification, we use the standard Node2Vec implementation\(^2\) with $p = 1, q = 0.1$, number of walks $= 200$, output vector size of 128, walk length of 10 and window size 5. These are decided after hyper-parameter tuning using Grid Search. We set the number of knowledge sentences to be selected as $k = 15$. Training is performed for 10 epochs with an initial learning rate of $3e^{-5}$ using Adam optimizer (epsilon $= 1e-6$) (Kingma and Ba, 2014). The BART model is fine-tuned on a single A100 GPU with 40GB memory, 24 vCPU threads, and 170GB RAM.

#### 3.3 Evaluation Metrics

We use Average Rank as the metric to evaluate the knowledge identification task. Given the actual set of grounding sentences - $s_1^q, s_2^q, \ldots, s_m^q$ for a given turn in a dialogue; and the corresponding rank of these sentences as predicted by a model - $r_1, r_2, \ldots, r_m$, the corresponding Rank (R) is estimated as

$$R = \frac{1}{m} \sum_{i=1}^{m} r_i$$

Based on the above definition of rank, Average Rank is computed accordingly by averaging over the evaluation samples. BLEU (Papineni et al., 2002; Post, 2018) metric is used to evaluate the response generation module.

\(^2\)https://github.com/eliorc/node2vec
Table 1: Evaluation of RANKING on the validation set of Doc2Dial dataset. It can be seen that our method improves the ranking (lower is better) of relevant sentences significantly over just using TF-IDF or BERT embedding-based similarity.

<table>
<thead>
<tr>
<th>Model</th>
<th>Average Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence BERT</td>
<td>19.62</td>
</tr>
<tr>
<td>TF-IDF</td>
<td>18.76</td>
</tr>
<tr>
<td>RANKING (our method)</td>
<td>11.09</td>
</tr>
</tbody>
</table>

Table 2: Response generation results on Doc2Dial dataset. RANKING improves the response quality compared to providing the entire document as input. RANKING also bridges the performance gap with supervised baseline - DIALKI which uses supervised labels for training knowledge identification module.

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entire document as input</td>
<td>17.67</td>
</tr>
<tr>
<td>RANKING (unsupervised)</td>
<td>18.09</td>
</tr>
<tr>
<td>DIALKI (supervised)</td>
<td>19.01</td>
</tr>
</tbody>
</table>

Table 3: Ablation experiments for the design of the graph structure to highlight the importance of different edges and their directionality.

<table>
<thead>
<tr>
<th>Description Information Flow between dialogue and document</th>
<th>Average Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent -&gt;Para -&gt;User</td>
<td>21.231</td>
</tr>
<tr>
<td>Para&lt;-&gt;User -&gt;Agent</td>
<td>12.684</td>
</tr>
<tr>
<td>User -&gt;Para -&gt;Agent</td>
<td>11.697</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Description Information Flow in the Document</th>
<th>Average Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undirected Sentence;</td>
<td>11.697</td>
</tr>
<tr>
<td>Para Connection</td>
<td>17.040</td>
</tr>
<tr>
<td>Sentence -&gt;Para</td>
<td>17.040</td>
</tr>
<tr>
<td>Undirected Adjacent</td>
<td>11.341</td>
</tr>
<tr>
<td>Para Connection</td>
<td>11.112</td>
</tr>
<tr>
<td>Directed Adjacent</td>
<td></td>
</tr>
<tr>
<td>Para Connection</td>
<td></td>
</tr>
</tbody>
</table>

3.4 Comparison with Baselines

We compare RANKING with the method where the entire document $K$ is provided as input with the dialogue context $D$ as input to the response generation model. We also compare with other simple unsupervised approaches like BERT embedding-based and TF-IDF-based ranking which uses cosine similarity between corresponding feature embeddings. This also evaluates the efficacy of utilizing the graph structure toward computing better embeddings for ranking. We also compare with DIALKI (Wu et al., 2021), a method that uses supervised labels for training the knowledge identification module.

It can be seen in Table 1 that our model improves knowledge identification significantly over using the TF-IDF or BERT embedding-based ranking outlining the importance of graph connections in obtaining enriched and contextualized embeddings. Further, Table 2 shows improvement in response generation BLEU score compared to providing the whole document as input. This highlights the fact that response generation improves if the selected knowledge is more precise, less noisy and the proportion of irrelevant sentences is lesser. Further, it can be noticed that RANKING helps in bridging the performance gap between unsupervised methods and the supervised baseline - DIALKI. Please refer to appendix C for qualitative analysis.

3.5 Ablation Study

We discuss the experiments performed to make design choices related to the graph structure. Results are shown in Table 3. We ablate over different ways of connecting various types of nodes as well as the directionality of edges. We observe that having a directed edge from the user utterance nodes to the paragraph nodes followed by an edge with the agent utterance node performs the best. For capturing the information flow within the document, we compare having directed vs. undirected edges between the sentence and paragraph nodes where the latter is found to perform better. Finally, directed edges between the adjacent paragraph nodes based on the order in which they appear in the document performs better than other options.

4 Conclusion and Future Work

We introduce a novel graph-based knowledge selection method RANKING that identifies relevant external knowledge in an unsupervised manner which can be used to generate better dialogue responses. RANKING captures the interdependencies between the dialogue utterances and text present in an external document. Most prior works rely on supervised labels to train the knowledge identification module which might not always be available. We show that our method improves knowledge selection over other unsupervised baselines and bridges the gap in performance with supervised methods. We also perform ablations to study the importance of graph structure. As future work, it can be explored to jointly train the knowledge identification and response generation modules to enable them to provide feedback to each other.
**References**


Zekang Li, Cheng Niu, Fandong Meng, Yang Feng, Qian Li, and Jie Zhou. 2019. Incremental transformer with deliberation decoder for document grounded conversations.


A Related Work

Existing approaches mainly apply a pipelined approach to perform Knowledge Identification followed by Response Generation by utilizing separate models for each task (Daheim et al., 2021; Xu et al., 2021; Li et al., 2021; Zhao et al., 2020; Lin et al., 2020). Some works have also attempted to jointly model both the steps together as a single generative objective (Guu et al., 2020; Lewis et al., 2021; Gao et al., 2022).

Very limited works have addressed the issue of identifying the relevant knowledge in an unsupervised manner. For instance, Li et al., 2019, Yavuz et al., 2019, and Lin et al., 2020 leverage implicit soft fusion techniques to combine externally available knowledge resources, without an explicit supervised training step. Although some attempts have been made to perform an unsupervised selection of external knowledge based on semantic similarity (Ghazvininejad et al., 2018; Huang et al., 2021 Dinan et al., 2019), the top-1 knowledge selection criteria employed by such methods make it difficult to identify the knowledge that should be present in the target response to be generated (Huang et al., 2021). Consequently, we employ top-k selection to improve the robustness of knowledge selection.

Some works have attempted to address the knowledge identification problem using graphs (Kim et al., 2022; Li et al., 2022; Wang et al., 2022). They model transitions between user and agent utterances in a dialogue to better identify knowledge over a knowledge graph. Most of these approaches are supervised and do not take full advantage of the interactions between the dialogue and the document, unlike our work. Moreover, these approaches rely heavily on standard knowledge graphs that are either provided or built on top of documents. In contrast, we focus on designing an unsupervised pipeline based on a custom graph that incorporates specific structures to ensure optimal performance for knowledge-grounded dialogue generation.

B Graph Structure

Our proposed graph structure is called a Document Dialogue Graph (Figure 2). It consists of 4 types of nodes: Sentence Nodes ($s^i_k$), Paragraph Nodes ($p_i$), User utterance nodes ($u_i$), and Agent utterance nodes ($a_i$). We explain the graph structure first followed by the intuition behind it. A sentence node in a paragraph is connected with the node representing the next sentence in the paragraph ($s^i_k \rightarrow s^i_{k+1}$). A paragraph node is connected to the next paragraph ($p_i \rightarrow p_{i+1}$) and also to the sentence nodes in that paragraph ($s^i_k \leftrightarrow p_i$). The user and agent utterance nodes are connected to the next utterance in the dialogue history ($u_i \rightarrow a_i$, $a_i \rightarrow u_{i+1}$). Utterance nodes are also connected to the paragraph nodes ($u_j \rightarrow p_i \rightarrow a_j$).

C Qualitative Analysis

Let us have a look at how the ground knowledge can be identified using examples from the dataset. This will also help us to reason about the importance of using a graph-based approach to identify relevant knowledge.

C.1 Example 1

Past dialogues reveal which topic is being discussed and can be used to locate the section which will have our grounding sentence. Thus noting where previous sentences have been grounded is important. This is done using edges between documents and dialogues. Refer to Figure 3 for an example.
Figure 3: The first dialogue mentions Medical Care Section and the rest of the user queries are all grounded in the Medical Care Section.

C.2 Example 2
Conversation flow sometimes follows the information flow in the section too. Thus edges between adjacent paragraphs and utterances are important. Refer to figure 4 for an example.

C.3 Example 3
There are two options for the agent to look for grounding knowledge: either follow the same topic on which the previous dialogue is based OR jump to a new topic which can be determined based on the current dialogue itself (not previous reference needed). Thus the two types of connections: between utterances and utterances and documents are important. Refer to Figure 5 for an example. An important thing to note is that the conversation does not jump back to a topic discussed in an earlier part of the conversation without the current dialogue being enough to decide that we have to jump to that topic. Thus it is unnecessary to model jumps to topics discussed earlier in the conversation and only recent topics have an effect on the current grounding which is done by our model with the help of edges between adjacent utterances in the conversation.
Figure 4: The information flow in the dialogue follows the information flow in the document.

Figure 5: Grounding Knowledge can be based on the same topic as the previous utterance or a new topic. The same coloured lines refer to the same topic.