From Answers to Guidance: A Proactive Dialogue System for Legal Documents

Ashish CHOUHAN 1 and Michael GERTZ

Data Science Group, Institute of Computer Science, Heidelberg University, Germany

Abstract. The accessibility of legal information remains a constant challenge, particularly for laypersons seeking to understand and apply complex institutional texts. While the European Union provides open access to legislation, parliamentary responses, and regulatory documents, these resources can be challenging for laypeople to explore. In this paper, we introduce EUDial, a proactive multi-turn dialogue dataset constructed from 204 blogs curated by the Citizens' Enquiries Unit (AskEP) of the European Parliamentary Research Service. EUDial contains 880 dialogue turns (averaging 4.3 turns per dialogue), where each dialogue includes initial questions, structured answers, and follow-up questions. Beyond dataset construction, we propose the LexGuide framework that leverages retrieval-augmented generation with hierarchical topic organization to structure dialogue progression, ensuring both comprehensive coverage of legal aspects and coherence across conversational turns. The results demonstrate that proactive, structured navigation closes the gap between the availability of legal information and citizen comprehension, establishing EUDial and LexGuide as practical resources for advancing proactive legal dialogue systems.

Keywords. legal natural language processing, dialogue dataset, European Union, follow-up questions, proactive dialogue system

1. Introduction

Around 865k legal documents are accessible via the EUR-Lex platform² for the years 1990 to 2025, including legislation, case law, and treaties. The Citizens' Enquiries Unit (AskEP)³ within the European Parliamentary Research Service (EPRS) allows citizens to pose questions and receive responses. While AskEP responses are not legal documents, they serve as informative texts of public relevance. Additionally, the European Union (EU) Publications Office has introduced Publio⁴, a virtual assistant for EU law access. Cherubini et al. [1] describe the project Chat-EUR-Lex, a dialogue system for EUR-Lex regulations in English and Italian.

¹Corresponding Author: Ashish Chouhan, chouhan@informatik.uni-heidelberg.de

²https://eur-lex.europa.eu/statistics/eu-law-statistics.html, accessed on 1st September 2025

³https://epthinktank.eu/author/epanswers, accessed on 2nd September 2025

 $^{^4\}mathrm{https://op.europa.eu/en/web/webtools/publio-the-publications-office-virtual-assistant, accessed on 2nd September 2025$

Despite these advances, a substantial gap remains between the availability of legal information and citizens' ability to comprehend and properly use that information. Legal texts are highly technical, characterized by complex terminology, intricate cross-references, and domain-specific reasoning patterns [2]. Consider a citizen seeking to understand healthcare rights in another EU member state. While Chat-EUR-Lex or Publio can respond, these are reactive dialogue systems [3], where users lead dialogue and systems follow the user's instructions. Reactive systems heavily rely on the user's ability to formulate precise queries without being guided toward broader or related issues, such as exceptions or related directives.

Traditional dialogue⁵ systems share this limitation, operating primarily using a reactive paradigm [3]. While generating accurate responses to independent questions, they rarely anticipate the user's informational needs where knowledge is layered and interconnected. This problem is particularly present in legal contexts, where non-expert users (laypersons) may not know which follow-up questions to ask for a comprehensive understanding, potentially leaving critical aspects unexplored. Although dialogue systems have advanced in healthcare and education, the legal domain remains comparatively underexplored [4]. Most legal natural language processing (NLP) research concentrates on document summarization, classification, or judgment prediction serving legal professionals [2,5,6,7], while resources for laypersons have received limited attention [5,8,9]. Large language models (LLMs) demonstrate remarkable multi-turn interaction capabilities [4], but face challenges with complicated document structures, complex legal language, and heterogeneous legal documents [2,10,11].

In this work, we introduce EUDial (European Union Dialogues), a dialogue dataset generated by transforming AskEP blogs into multi-turn dialogues with queries, answers, and relevant follow-up questions. We also propose the LexGuide (Guided Exploration of Legal Documents) framework that leverages Retrieval-Augmented Generation (RAG) [12] to retrieve relevant information and construct hierarchical topic trees for structuring dialogue systems. Using educational scaffolding principles [13], LexGuide helps transform dialogue systems from reactive dialogue systems to proactive dialogue systems that actively guide users through topic exploration. Our work makes two key contributions: First, we present a methodology that systematically transforms singleturn QA pairs from AskEP blogs into multi-turn dialogues, thereby creating the benchmark dataset EUDial for evaluating proactive dialogue systems. Second, we introduce the LexGuide framework, which formalizes dialogue progression as retrieval, hierarchical organization, and strategic navigation, thus enabling proactive behavior while maintaining topical coherence between dialogue turns⁶. The framework transitions from traditional reactive dialogue systems to intelligent conversation partners that can guide users through complex topics.

The paper is organized as follows: Section 2 reviews prior research, Section 3 describes the EUDial dataset, and Section 4 introduces the LexGuide framework. Sections 5 and 6 present the experimental setup and evaluation, and Section 7 concludes with key findings and future directions.

⁵In this paper, the terms "dialogue" and "conversation" are used interchangeably.

 $^{^6}$ Code for EUDial dataset construction and LexGuide framework are available at https://github.com/achouhan93/EUDial-LexGuide

2. Related Work

Our work sits at the intersection of dialogue systems, follow-up question generation, and legal NLP, specifically developing proactive dialogue systems that guide conversations and provide explanations to laypersons seeking legal information.

Proactive Dialogue Systems. Traditional dialogue systems operate reactively, following user-driven conversations. However, this reactive paradigm fails when serving laypersons who lack domain expertise to formulate proper follow-up questions. Recent research addresses this limitation through proactive dialogue systems that actively guide users toward their information goals [3]. Zhang et al. [14] propose the ask and retrieve knowledge framework, where LLMs iteratively generate sub-queries for decision making. Butala et al. [15] propose ProMISe, where systems generate suggested questions targeting atomic aspects of a user's intent. While promising, these approaches have not been systematically applied to information-seeking contexts, where users require a structured understanding of complex topics, particularly in domains such as the legal domain.

Follow-up question generation (FQG) serves as a critical mechanism for improving information-seeking dialogues [16]. Early work focused on clarification questions for resolving ambiguity [17,18,19]. However, for proactive systems targeting laypersons, clarification alone proves insufficient; instead, systems must provide follow-up questions alongside answers to guide a systematic exploration of complex topics [15]. The FOLLOWUPQG [20] dataset provides over 3K real-world (question, answer, follow-up question) triples from Reddit forums, offering layperson-friendly follow-ups but limiting the scope to single-turn dialogues. The TREC CAsT 2022 dataset [21] includes multi-turn dialogues with Wikipedia-grounded follow-up questions but remains general-domain without specialization for closed domains. These works demonstrate FQG's potential but reveal critical gaps, such as the lack of multi-turns, domain-specific proactive dialogue systems, and specialized FQG resources for complex information domains.

Legal Domain Datasets and Systems. The legal NLP landscape includes large-scale datasets and benchmarks like Pile of Law [22] and LegalBench [23]. Question-answering (QA) resources encompass CUAD [24], PrivacyQA [25], and LegalQA [26]. Recent conversational legal datasets include ConvLegal [27] for case retrieval dialogues and Legal-Conv [9] with 360k layperson questions and 780k lawyer responses. However, Legal-Conv does not capture actual turn-taking dialogues. LegalBot [28] provides short simulated layperson-lawyer exchanges extracted from law textbooks, averaging only two turns per dialogue. Existing research in conversational legal document access includes LexRAG [29] for Chinese legal statutes and Chat-EUR-Lex [1] for EU regulations. However, these systems cannot engage in a proactive dialogue or systematically guide laypersons through complex legal topics. Critically, existing legal conversational datasets predominantly address legal advice and lack follow-up questions designed for structured topic exploration.

We identify and address two critical gaps: (1) the absence of proactive multi-turn dialogue datasets in specialized domains, and (2) the lack of systematic approaches for realizing proactive dialogue systems that will guide laypersons toward specific information through structured follow-up question generation.

3. The EUDial Dataset

Document model and task definition. In the following, we detail the construction of dialogue datasets from AskEP⁷ blogs curated by the Citizens' Enquiries Unit (AskEP) within the European Parliamentary Research Service (EPRS). Each blog entry is represented as a document $d_i = \langle q_i, a_i \rangle$, where q_i is the citizen's question and a_i is the expert response decomposed into sections $a_i = [s_1^i, \ldots, s_j^i]$. Each section s_j^i is modeled as a pair $\langle h_j^i, c_j^i \rangle$, with header h_j^i and content c_j^i . Documents also include metadata such as publication date, hyperlinks, and thematic tags.

We view a document d_i as a single-turn dialogue, where a user query q_i is followed by an answer a_i . The task is to transform this into a multi-turn dialogue $T_i = \{t_1^i, \dots, t_j^i\}$, where each dialogue turn is defined as:

$$t_i^i = \langle u_j^i, r_j^i, f_j^i \rangle, \tag{1}$$

with u^i_j being the user utterance, which at each turn t^i_j may be (i) the initial user query, (ii) a system-proposed follow-up query when acknowledged by the user, or (iii) a new user query posed when the system-proposed follow-up query is not acknowledged. The system reply r^i_j is grounded in c^i_j , and f^i_j is a proactive follow-up query guiding further exploration. Algorithm 1 (Appendix F.1) describes the procedure for converting a single-turn QA pair into a multi-turn dialogue. Appendix C illustrates an example QA pair collected from AskEP and its corresponding structured representation. The resulting dataset $T = \{T_1, \ldots, T_n\}$ thus enables the evaluation of dialogue systems that respond and initiate systematic knowledge acquisition.

Dataset construction. The EUDial dataset consists of multi-turn dialogues with section-level references and supporting legal documents, systematically transformed from AskEP blogs. We collected 204 blogs published between January 2014 and July 2025, each addressing a specific citizen enquiry (question) through multi-sectioned expert responses (answer). Blogs are scraped into QA pairs while preserving their section structure and metadata. Since citizen questions often appear ambiguously (e.g., as titles or introductory paragraphs), GPT-40-mini⁸ is prompted to generate standardized question formulations (see also Appendix E.1). For each QA pair, multi-turn dialogues are then generated using GPT-40-mini by mapping sections to dialogue turns through a controlled prompt (see also Appendix E.2) and following Algorithm 1 (Appendix F.1). Generation of the full dataset required approximately 45 minutes on a standard personal computer (16 GB RAM, ARM-based processor), incurring a total cost of \$0.11. Importantly, hyperlinks embedded in response sections are preserved as references to legal documents, ensuring that each dialogue turn provides verifiable legal grounding.

As shown in Table 1, existing resources like Legalbot focus primarily on reactive dialogues (legal advice) without proactive follow-up mechanisms. At the same time, FOL-LOWUPQG provides open-domain single-turn follow-up triples. ProMISe represents a proactive, multi-turn dialogue dataset in an open domain but operates in a two-player simulation without grounding in authoritative sources. The EUDial dataset is constructed from human-written AskEP blogs, where each conversation is explicitly grounded to the

⁷https://epthinktank.eu/author/epanswers, accessed on 29th August 2025

 $^{^8 \}texttt{https://platform.openai.com/docs/models/gpt-4o-mini}, accessed on 2nd September 2025$

Table 1. Comparison of EUDial with existing conversational legal and follow-up question datasets. A dialogue is proactive when the system not only answers but also generates follow-up questions to guide exploration, while reactive systems only respond. One dialogue turn consists of a user utterance and a system reply (plus a follow-up in proactive). *Attribution* indicates response grounding in external sources.

Feature	EUDial	Legalbot	FOLLOWUPQG	ProMISe	
Domain	EU law/policy	Law texbooks	Open-domain	Open-domain	
Dialogue type	Multi-turn, proactive	Multi-turn, reactive	Single-turn, proactive	Multi-turn, proactive	
# dialogues	204	1200	3790	1025	
# dialogue turns	880	2400	3790	4453	
Avg # dialogue turns	4.3	2	1	4.3	
Attribution	Yes	No	No	No	

original blog content and referenced legal documents, thus acting as a benchmark dataset for proactive dialogue systems in the legal domain.

Dataset Assumption. The dialogue generation process operates under assumptions that define the scope of conversational behaviors captured in EUDial. First, follow-up questions are generated sequentially for each subsequent section without multiple questions targeting the same section content. Second, we assume users exhibit consistent engagement by always responding affirmatively to explore presented information, creating a linear progression through all available sections. As a result, the dataset does not capture alternative conversational paths such as users declining information, requesting different topics mid-conversation, or terminating dialogue before entire topic coverage. These assumptions enable the creation of a systematic dataset, but they represent areas for future extension to more complex user behaviors.

Dataset Characteristics. The AskEP blog entries that seed the EUDial dataset exhibit structural properties making conversational modeling both necessary and beneficial. Each answer has multi-sections (3.9 on average; maximum of 9) with substantial section-level content (mean of 155 words). Questions are comparatively short (mean length is around 18 words) relative to answers, yielding a very low question-to-answer ratio (\approx 0.04) at the QA-pair level, which indicates that user queries provide narrow entry points into broader content. Readability is challenging for lay users (Flesch Reading Ease =26.2 on average), indicating a "difficult" level. These characteristics highlight the need for a dialogue dataset that can (i) decompose long, multi-section answers into navigable units, and (ii) generate accessible follow-up questions to guide users through otherwise inaccessible legal material.

The resulting EUDial dialogue dataset comprises 204 dialogues, with each dialogue averaging 4.3 turns (ranging from 2 to 18 turns), and alternating roles between citizens (layperson) and EPRS (system). Importantly, each system turn is grounded in the underlying legal resources used to construct the answer. Moreover, every system response not only provides an answer to the user's question but also includes a follow-up question that introduces the next section of the legal content, thereby guiding the dialogue forward. Thus, EUDial acts as a benchmarck dataset for proactive dialogue systems. Further dataset insights for the collected AskEP blogs and constructed EUDial dialogue dataset are provided in Appendix D.

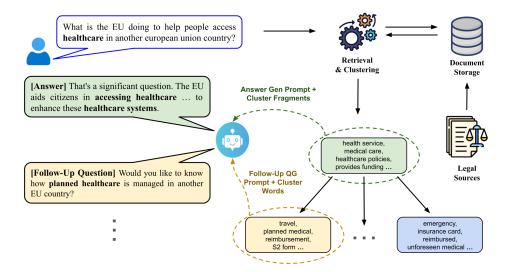


Figure 1. Overview of the LexGuide framework, illustrated with an example of the first turn in a multi-turn dialogue. The framework processes initial user requests, constructs topic tree representations, and leverages this structure to generate an answer and a follow-up question for guiding the user.

4. The LexGuide Framework

EUDial dialogue dataset (Section 3) serves as a benchmark dataset for evaluating proactive multi-turn dialogue systems. Building on EUDial, we propose the LexGuide framework (see Figure 1), a proactive dialogue framework integrating Retrieval-Augmented Generation (RAG) with hierarchical topic organization and system-driven follow-up question generation to guide users through legal information structurally. For a dialogue d_i with turns t_1^i, t_2^i, \ldots , at each dialogue turn t_j^i with user utterance u_j^i , the framework executes hierarchical topic organization for structuring retrieved information, and uses RAG for response and follow-up question generation.

Corpus preparation. Let $X = \{x_1, \dots, x_m\}$ denote a legal corpus of m documents, constructed from legal sources referenced in the EUDial dataset (refer to Figure 2f in Appendix D for the distribution of legal sources). Each document x_i is modeled as an ordered sequence of fragments $P_i = [p_1^i, p_2^i, \dots, p_l^i]$, where each fragment is a sequence of tokens, in most cases a sentence or a sequence of sentences. These fragments are embedded into a vector space and indexed for retrieval.

Hierarchical organization. On the initial dialogue turn t_1^i , the system retrieves the top-k fragments R that are most relevant to the user utterance u_1^i , i.e., the user's initial query. To reduce fragment redundancy, retrieval employs diversity-aware Maximum Marginal Relevance (MMR) [30]. The retrieved fragments R are clustered using BERTopic [31], which leverages fragment embeddings followed by density-based clustering to identify clusters. To capture structured knowledge representation, hierarchical clustering organizes clusters into parent-child relationships. Each cluster represents a topic, and term frequency-inverse document frequency (tf-idf) is used to extract topic representations. This creates a topic tree $G_i = (V, E)$ where $V = \{g_0, g_1, \dots, g_H\}$ represents topic nodes, and E specifies relationship between nodes. The root node g_0 encompasses the broadest

thematic coverage, while leaf nodes represent subtopics. Each intermediate level provides progressively refined semantic granularity, enabling users to explore information at their preferred level of detail.

Each topic node $g \in V$ is formally characterized by a triple $\langle W_g, C_g, F_g \rangle$, where $W_g = \{w_1^g, w_2^g, \dots, w_l^g\}$ is an ordered set of l representative words ranked by topic relevance score, $C_g \in \mathbb{R}^d$ is the topic centroid computed as the mean embedding of associated fragments, and $F_g \subseteq R$ is the subset of fragments assigned to topic g. This hierarchical topic organization transforms the initially flat retrieved fragment set into a structured exploration space, where each level of the tree G offers distinct semantic granularity for systematic dialogue navigation.

Dialogue navigation and follow-up question generation. The navigation state at turn t_i^i is represented as

$$S_{i}^{i} = (Y_{i}^{i}, L_{i}^{i}, Z_{i}^{i}, B_{i}^{i}, K_{i}^{i}, H_{i}^{i}), \tag{2}$$

where Y_j^i is the set of visited nodes, L_j^i is the current node, Z_j^i are unexplored children of L_j^i , B_j^i is the navigation strategy (Breadth-First Search (BFS), Depth-First Search (DFS), or user-driven), K_j^i is the path of visited nodes, and H_j^i is the dialogue history composed of user utterance u_j^i , response r_j^i , and follow-up query f_j^i .

At each turn t_j^i , dialogue navigation is determined by operations, i.e., *descend* into a child cluster for finer exploration, *lateral* movement across sibling clusters for related themes, *ascend* to a parent cluster for broader context, *jump* to any cluster in the hierarchy for direct access, or *backtrack* using path to previously visited states for re-exploration. Due to navigation state (as shown in Equation 2), users are provided flexiblity to return to the previous turn (t_{j-1}^i) , revisit an earlier turn (t_{j-n}^i) , returning to the point where a desired topic was active, or reopen a past discussion point to explore alternative trajectories.

The framework employs navigation strategies B_i^i based on dialogue context and topic characteristics: (1) BFS explores sibling clusters at the current level before descending, i.e., *lateral* operation, ensuring comprehensive coverage of related themes, (2) DFS follows topic hierarchies to a maximum depth, enabling detailed exploration of a specific topic, i.e., descend operation, and (3) a user-driven strategy adapts to explicit user navigation requests, i.e., ascend, jump, or backtrack operations. When a user acknowledges a system-proposed follow-up $(u_{j+1}^i = f_j^i)$, the framework continues traversal within G using the selected strategy, with the default being BFS for diverse topical coverage. If the user does not acknowledge a follow-up and instead poses a new query q', the framework first checks whether q' can be addressed by revisiting nodes that have already been explored (Y_i^l) or by descending into unexplored children of the current node (Z_i^i) . If q' cannot be resolved through these options, the framework computes coverage relative to the active tree G as $Cov(G,q') = \max_{g \in G} sim(q',C_g)$. Considering τ as the topic coverage threshold, if $Cov(G, q') \ge \tau$, the framework directly navigates to the bestmatching cluster. If $Cov(G, q') < \tau$, the user is prompted to clarify whether to shift to a new topic, in which case a new tree G' is constructed. Depending on the user's response, the framework flexibly applies these operations while RAG filters fragments associated with the active node L_i^i . These filtered fragments are concatenated with u_i^i and prompted to the LLM for a response generation r_i^i . Each turn is thus a triple $\langle u_i^i, r_i^i, f_i^i \rangle$, where f_i^i denotes the generated follow-up, consistent with the dialogue model defined in Eq. 1.

The dialogue navigation procedure identifies the next topic g_{next} to explore and follow-up question is generated using W_{next} and dialogue history H_i^i ,

$$f_i^i = \text{LLM}(\text{prompt}(W_{\text{next}}.H_i^i)),$$
 (3)

where $W_{\text{next}} = \{w_1, \dots, w_l\}$ denotes the top-l topic representative words for the next topic g_{next} . A dialogue terminates under one of three conditions: (i) complete topic coverage when all nodes in the topic tree have been visited, (ii) explicit user satisfaction signals detected, or (iii) dialogue abandonment indicated by user inactivity or explicit termination requests. Algorithm 2 (Appendix F.2) formalizes the procedure, showing how the framework gradually narrows from broad responses to fine-grained exploration while allowing flexible pivots.

5. Experimental Setup

For assessing the effectiveness of the proposed LexGuide framework, we design an experiment that evaluates both response generation quality and dialogue navigation performance. The evaluation is grounded in a carefully constructed legal corpus derived from the EUDial dataset. Our setup emphasizes three aspects: (i) corpus preparation and preprocessing to ensure a representative evaluation resource, (ii) methodological comparisons against established baselines, and (iii) automated evaluation metrics to evaluate factual accuracy, completeness, legal relevance, navigational effectiveness, and follow-up quality. This integrated approach enables us to examine not only whether LexGuide generates relevant answers, but also whether it systematically guides users through legal topics in a proactive and interpretable manner.

We construct the experimental corpus from the EUDial metadata by considering 2450 hyperlinks (legal source URLs). After URL normalization and de-duplication, 2036 unique URLs remain. Removing 181 social-media endpoints yields 1855 processable URLs. Of these, 1329 are web pages, 91 are PDF documents, and 435 return HTTP errors. The final corpus used in experiments comprises 1420 processable URLs. The corpus construction process begins with content extraction using unstructured.io⁹ for web pages and DocLing [32] for PDFs. Fragments obtained while content extraction are embedded using *nlpaueb/bert-base-uncased-eurlex* [33], and indexed in FAISS [34] for retrieval. Retrieval uses MMR with diversity hyperparameter $\lambda = 0.6$ for broader topic coverage. For text generation, we evaluate GPT-40-mini for efficient processing with strong reasoning capabilities, llama-3.1-8b-instant 10 and mixtral – 8x7B model 11 for open-source performance, and gemma2-2b 12 for resource-constrained deployment scenarios. Temperature settings are maintained at 0.3 across all models for balanced creativity and consistency in response generation and follow-up question formulation.

Topic modeling at the first dialogue turn t_1^t utilizes BERTopic with UMAP [35] dimensionality reduction, HDBSCAN [36] for density-based clustering, and tf-idf to ex-

⁹https://unstructured.io, accessed on 7th September 2025

¹⁰https://console.groq.com/docs/model/llama-3.1-8b-instant, accessed on 8th September 2025

¹¹https://ollama.com/library/mixtral:8x7b (accessed on 12th September 2025)

¹²https://ollama.com/library/gemma2:2b, accessed on 12th September 2025

tract representative keywords over cluster fragments. To obtain a hierarchy, hierarchical clustering employs agglomerative clustering with average linkage. This procedure yields a topic tree G=(V,E) whose depth is not fixed a priori; in the current EUDial dataset, dialogues yield shallow hierarchies of depth two, reflecting that citizen questions tend to anchor a dominant theme with a small number of focused subtopics. In other domains or dialogue styles, the same procedure may yield deeper trees without modification. Due to these characteristics of the EUDial dataset during the dialogue navigation in the LexGuide framework, the BFS navigation strategy is considered for lateral sibling navigation.

For the initial user utterance of a dialogue, the system retrieves fragments using diversity-aware MMR with $\lambda=0.6$ and considering top-k fragments with k=500. These retrieved fragments are used for hierarchical topic modeling, while immediate answering relies on a narrower subset of fragments from the node in focus (k=10). The follow-up generator consumes only the next node's representative words with turn context (refer to Equation 3).

6. Evaluation

Evaluation Methodology and Metrics. We evaluate the proposed LexGuide framework using the EUDial dataset (comprised of 204 multi-turn dialogues), focusing on response generation and dialogue navigation. For each dialogue, the LexGuide framework initializes the system with the first user question, constructs the hierarchical topic tree G, and executes the navigation-driven policy that generates answers and proposes follow-ups until the termination criteria are met, i.e., all sub-topics are explored. We compare against baselines: (i) RAG-Basic, a single-turn RAG without MMR or clustering; (ii) RAG-MMR, incorporating diversity-aware retrieval without clustering; and (iii) ConvRAG, a conversational RAG generating follow-ups without topic modeling or navigation state tracking. All baselines use the same FAISS index and embedding model (nlpaueb/bert-base-uncased-eurlex).

The answer quality is evaluated along: (i) groundedness, measured as the proportion of sentences overlapping with retrieved fragments; (ii) completeness, computed as ROUGE-L recall against gold responses; (iii) relevance, measured with BERTScore [37] to capture semantic similarity between generated and gold responses; and (iv) layperson readability, measured using the Flesch Reading Ease (FRE) [38] score estimating textual accessibility for non-experts. Including FRE enables us to complement legal correctness with accessibility, thereby capturing whether system responses are not only accurate but also comprehensible to lay audiences.

Follow-up quality is assessed in terms of (i) relevance, measured as similarity between generated follow-ups and gold expert follow-ups; (ii) diversity, average pairwise distance between follow-ups generated within dialogues; and (iii) temporal consistency, measuring conversational flow coherence. Topic coverage evaluates the proportion of gold-relevant sub-topics discussed during the dialogue.

Results on Answer Quality. As shown in Table 2, LexGuide demonstrates competitive performance in answer quality metrics, with notable architecture-dependent variations. The GPT-40-mini variant achieves the highest groundedness score (98.7%) and relevance score (0.761) across all methods. Readability remains consistently high across all variants, indicating that the inclusion of layperson-oriented phrasing is preserved.

LexGuide

Mixtral-8x7B

Method	Model	Grounded- ness (%)	Completeness (ROUGE-L)	Relevance (BERTScore)	Readability (FRE)
RAG-Basic	LlaMa-3.1-8B	96.4	0.40	0.72	96.1
RAG-MMR	LlaMa-3.1-8B	96.4	0.40	0.72	96.1
ConvRAG	LlaMa-3.1-8B	97.7	0.42	0.74	96.5
LexGuide	LlaMA-3.1-8B	96.5	0.38	0.73	96.6
LexGuide	GPT-4o-mini	98.7	0.37	0.76	96.2
LexGuide	Gemma2-2B	92.9	0.37	0.72	92.7

0.40

0.75

98.5

96.3

Table 2. Comparative evaluation of response quality across baselines and LexGuide.

Results on Navigation and Follow-up Quality. Topic coverage evaluation reveals LexGuide's novel navigation capabilities. Since baselines lack topic hierarchies, their topic coverage relies on content-based (86.6-87.6%). LexGuide's topic-word-based navigation achieves 73.7% topic coverage across all models, while content-based evaluation shows competitive performance (87.3-89.5%), confirming that hierarchical exploration maintains topic relevance. LexGuide's topic-word-driven design promotes substantial diversity (0.204-0.224) in follow-up questions compared to baseline methods (0.201-0.205), helping prevent redundancy in dialogues. The generated follow-ups remain semantically relevant and contextually linked to prior turns, thereby supporting coherent exploration.

Thus, LexGuide structured topic navigation maintains high-quality dialogue and framework consistency across model architectures (identical 73.7% topic coverage) demonstrates robust hierarchical navigation. The slight reduction in completeness reflects LexGuide's design choice of prioritizing depth and relevance over exhaustive coverage, aligning with guided legal exploration rather than comprehensive but potentially overwhelming responses.

7. Conclusion and Future Work

This paper addresses the gap between the availability of legal information and citizen comprehension through EUDial, a benchmark dataset comprising 204 proactive multiturn dialogues from European Parliamentary Research Service blogs, and LexGuide, a hierarchical topic-guided dialogue framework for proactive legal exploration. Empirical results demonstrate that LexGuide consistently maintains strong factual groundedness (98.7% with GPT-4o-mini), semantic relevance, and readability, thereby addressing both legal accuracy and accessibility for lay audiences. The topic coverage evaluation reveals that while topic-word-based coverage (73.7%) demonstrates structured hierarchical navigation, content-based coverage (87.3-89.5%) confirms comprehensive topic relevance, validating the framework's guided exploration approach. The evaluation results highlight a strategic design trade-off, where LexGuide's 2.7% reduction in completeness reflects a prioritization of focused, relevant responses over exhaustive coverage, aligning to deliver accessible legal information. The framework's model-agnostic design and consistent hierarchical navigation across architectures demonstrate robust applicability. In our ongoing work, we are incorporating human studies with legal experts and lay users to validate factual faithfulness and navigational clarity beyond automatic metrics.

References

- Cherubini M, Romano F, Bolioli A, De L, Sangermano M. Improving the accessibility of EU laws: The Chat-EUR-Lex project. In: Proceedings of the 4th National Conference on Artificial Intelligence; 2024. p. 1-6.
- [2] Ariai F, Mackenzie J, Demartini G. Natural language processing for the legal domain: A survey of tasks, datasets, models, and challenges, arXiv preprint arXiv:241021306, 2024.
- [3] Deng Y, Liao L, Lei W, Yang GH, Lam W, Chua TS. Proactive Conversational AI: A Comprehensive Survey of Advancements and Opportunities. ACM Transactions on Information Systems. 2025;43(3):1-45.
- [4] Mo F, Mao K, Zhao Z, Qian H, Chen H, Cheng Y, et al. A Survey of Conversational Search. ACM Transactions on Information Systems. 2025. Accepted on 26th July 2025.
- [5] Janatian S, Westermann H, Tan J, Savelka J, Benyekhlef K. From Text to Structure: Using Large Language Models to Support the Development of Legal Expert Systems. In: Proceedings of the 36th International Conference on Legal Knowledge and Information Systems (JURIX). vol. 379; 2023. p. 167-76.
- [6] Aumiller D, Chouhan A, Gertz M. EUR-Lex-Sum: A Multi- and Cross-lingual Dataset for Long-form Summarization in the Legal Domain. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP); 2022. p. 7626-39.
- [7] Chouhan A, Gertz M. LexDrafter: Terminology Drafting for Legislative Documents Using Retrieval Augmented Generation. In: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING); 2024. p. 10448-58.
- [8] Büttner M, Habernal I. Answering legal questions from laymen in German civil law system. In: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL); 2024. p. 2015-27.
- [9] Askari A, Aliannejadi M, Abolghasemi A, Kanoulas E, Verberne S. CLosER: Conversational Legal Longformer with Expertise-Aware Passage Response Ranker for Long Contexts. In: Proceedings of the ACM International Conference on information and knowledge management (CIKM); 2023. p. 25-35.
- [10] Ganguly D, Conrad JG, Ghosh K, Ghosh S, Goyal P, Bhattacharya P, et al. Legal IR and NLP: The History, Challenges, and State-of-the-Art. In: Proceedings of the 45th European Conference on Information Retrieval (ECIR); 2023. p. 331-40.
- [11] Jiang H, Zhang X, Mahari R, Kessler D, Ma E, August T, et al. Leveraging Large Language Models for Learning Complex Legal Concepts through Storytelling. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL); 2024. p. 7194-219.
- [12] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS); 2020. p. 9459-74.
- [13] Vygotsky LS. Mind in Society: Development of Higher Psychological Processes. vol. 86. Harvard University Press; 1978.
- [14] Zhang B, Wang S, Jiang Y, Sui D, Tu Z, Chu D. Ask and Retrieve Knowledge: Towards Proactive Asking with Imperfect Information in Medical Multi-turn Dialogues. In: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR); 2025. p. 1055–1065.
- [15] Butala Y, Garg S, Banerjee P, Misra A. ProMISe: A Proactive Multi-turn Dialogue Dataset for Information-seeking Intent Resolution. In: Findings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL); 2024. p. 1774-89.
- [16] Hu J, Guo J, Tang N, Ma X, Yao Y, Yang C, et al. Designing the Conversational Agent: Asking Follow-up Questions for Information Elicitation. Proceedings of the ACM on Human-Computer Interaction. 2024;8:1-30.
- [17] Rahmani HA, Wang X, Feng Y, Zhang Q, Yilmaz E, Lipani A. A Survey on Asking Clarification Questions Datasets in Conversational Systems. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL); 2023. p. 2698-716.
- [18] Rao S, Daumé III H. Learning to Ask Good Questions: Ranking Clarification Questions using Neural Expected Value of Perfect Information. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL); 2018. p. 2737-46.
- [19] Gatto J, Seegmiller P, Burdick TE, Khayal IS, DeLozier S, Preum SM. Follow-up Question Generation For Enhanced Patient-Provider Conversations. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL); 2025. p. 25222-40.

- [20] Meng Y, Pan L, Cao Y, Kan MY. FollowupQG: Towards information-seeking follow-up question generation. In: Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL); 2023. p. 252-71.
- [21] Owoicho P, Dalton J, Aliannejadi M, Azzopardi L, Trippas JR, Vakulenko S. TREC CAsT 2022: Going Beyond User Ask and System Retrieve with Initiative and Response Generation. In: Proceedings of the 31st Text REtrieval Conference (TREC); 2022. p. 1-12.
- [22] Henderson P, Krass MS, Zheng L, Guha N, Manning CD, Jurafsky D, et al. Pile of Law: Learning Responsible Data Filtering from the Law and a 256GB Open-Source Legal Dataset. In: Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS); 2022. p. 29217 29234.
- [23] Guha N, Nyarko J, Ho DE, Ré C, Chilton A, Narayana A, et al. LEGALBENCH: a collaboratively built benchmark for measuring legal reasoning in large language models. In: Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS); 2023. p. 44123-279.
- [24] Hendrycks D, Burns C, Chen A, Ball S. CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review. NeurIPS. 2021.
- [25] Ravichander A, Black AW, Wilson S, Norton T, Sadeh N. Question Answering for Privacy Policies: Combining Computational and Legal Perspectives. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP); 2019. p. 4947-58.
- [26] Askari A, Yang Z, Ren Z, Verberne S. Answer Retrieval in Legal Community Question Answering. In: Proceedings of the 46th European Conference on Information Retrieval (ECIR); 2024. p. 477-85.
- [27] Liu B, Hu Y, Ai Q, Wu Y, Liu Y, Li C, et al. Structure-Aware Conversational Legal Case Retrieval. ACM Transactions on Information Systems. 2025;43(3):1-28.
- [28] John AK, Di Caro L, Robaldo L, Boella G. Legalbot: A Deep Learning-Based Conversational Agent in the Legal Domain. In: Proceedings of International Conference on Applications of Natural Language to Information Systems (NLDB); 2017. p. 267-73.
- [29] Li H, Chen Y, YiRan H, Ai Q, Chen J, Yang X, et al. LexRAG: Benchmarking Retrieval-Augmented Generation in Multi-Turn Legal Consultation Conversation. In: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR); 2025. p. 3606-15
- [30] Carbonell J, Goldstein J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR); 1998. p. 335-6.
- [31] Grootendorst M. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv preprint arXiv:220305794. 2022.
- [32] Auer C, Lysak M, Nassar A, Dolfi M, Livathinos N, Vagenas P, et al. Docling technical report. arXiv preprint arXiv:240809869. 2024.
- [33] Chalkidis I, Fergadiotis M, Malakasiotis P, Aletras N, Androutsopoulos I. LEGAL-BERT: The Muppets straight out of Law School. In: Findings of the Conference on Empirical Methods in Natural Language Processing (EMNLP); 2020. p. 2898-904.
- [34] Johnson J, Douze M, Jégou H. Billion-scale similarity search with GPUs. IEEE Transactions on Big Data. 2019;7(3):535-47.
- [35] McInnes L, Healy J, Saul N, Großberger L. UMAP: Uniform Manifold Approximation and Projection. Journal of Open Source Software. 2018;3(29):861.
- [36] Campello RJGB, Moulavi D, Sander J. Density-Based Clustering Based on Hierarchical Density Estimates. In: Proceedings of the 17th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD); 2013. p. 160-72.
- [37] Zhang T, Kishore V, Wu F, Weinberger KQ, Artzi Y. BERTScore: Evaluating Text Generation with BERT. In: Proceedings of the 8th International Conference on Learning Representations (ICLR); 2020. p. 1-43.
- [38] Flesch R. A new readability yardstick. Journal of Applied Psychology. 1948;32(3):221.

A. Use of Large Language Models

In this paper, we use large language models (ChatGPT-5 from OpenAI and Claude Sonnet 4 from Anthropic) strictly for editorial and technical assistance. The tools were employed for proofreading, refining language for clarity and academic tone, and providing minor coding support (e.g., boilerplate generation and suggestions for efficiency). All intellectual contributions, research design, analyses, and conclusions remain entirely those of the authors. AI-generated content was reviewed, edited, and validated before inclusion.

B. Ethics and Data Statement

AskEP blogs consist of curated informational texts intended for public communication and education; they are not legally binding sources or official legal documents. Our dataset construction preserves the original sectioning, phrasing, and expert content while clearly distinguishing machine-generated elements, specifically the generated citizen questions and dialogue turns created through GPT-40-mini. All original metadata, including publication dates, topic tags, and embedded hyperlinks to legal sources, are retained to support reproducibility, enable temporal analyses of policy evolution, and facilitate potential future dataset updates. The transformation process maintains attribution to source sections and legal documents, ensuring transparency in the provenance of information.

C. Example QA Pair

As described in Section 3, each AskEP blog entry is modeled as a document comprising the citizen question, a multi-sectioned answer, and associated metadata. To illustrate, we present below a representative example of how such a blog entry appears after document modelling.

For a document on access to healthcare in another European Union (EU) country¹³, collected information from AskEP is as follows:

- **question** (q_i) : "What is the EU doing to help people access healthcare in another EU country?"
- answer (*a_i*):
 - * Planned healthcare: Under EU law, you have the right to travel to another EU country to receive medical care (consultation with a specialist, surgery ...
 - * Unplanned healthcare (emergency): Should you need unforeseen medical treatment while on holiday, a business trip or while studying in another EU country for a short period ...
- **Metadata** ($meta(d_i)$): {date = 2025-04-01, tags = [cross-border healthcare, public health], links = ["Planned healthcare" \rightarrow EU law, S2 form, ...; "Unplanned healthcare" \rightarrow European Health Insurance Card; ...]}

 $^{^{13}} https://epthinktank.eu/2025/04/01/what-is-the-eu-doing-to-help-people-access-healthcare-in-another-eu-country, accessed on 29th August 2025$

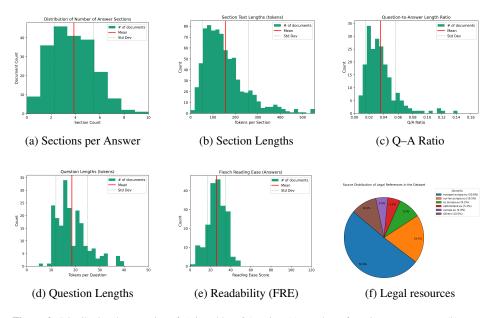


Figure 2. Distributional properties of *AskEP* blog QA pairs: (a) number of sections per answer, (b) section lengths (tokens), (c) question-to-answer length ratios, (d) question lengths (tokens), (e) readability scores (FRE), and (f) distribution of legal resources.

D. Dataset Insights

AskEP blogs. Each expert-written answer from the AskEP blog entry comprises multiple sections (mean = 3.89, maximum = 9) (Figure 2a). Answer sections vary widely, with an average of 155 tokens and a maximum of 772 words, indicating heterogeneity in information granularity (Figure 2b). The ratio between question length and total answer length averages 0.036, confirming that questions are short entry points while answers introduce substantially more information (Figure 2c). User questions average 18 words, consistent with concise layperson queries (Figure 2d). Answers have poor readability for non-experts, with an average Flesch Reading Ease score of 26.2 (Figure 2e), suggesting that proactive dialogue support is necessary for accessibility.

EUDial Dialogue Dataset. Dialogues average 4.3 turns (min = 2, max = 18). Figure 3a shows the distribution, confirming that most dialogues fall between 6 and 12 turns. By role, citizen utterances average 12.0 words, and system answers average 39.7 words. Figure 3b highlights the distribution of system answers. Pairing each citizen turn with the subsequent EPRS yields a dialogue-level $q \rightarrow a$ ratio distribution with mean = 0.311 (reflecting tighter, incremental exchanges compared to static QA pairs). Figure 3c illustrates the distribution across dialogues. Measured with Flesch Reading Ease (FRE), FRE averages 23.95 per turn; when averaged per dialogue, the mean per-dialogue FRE is 23.89, placing them in a difficult-to-read range for laypersons. Figure 3d shows the FRE score distribution across answers.

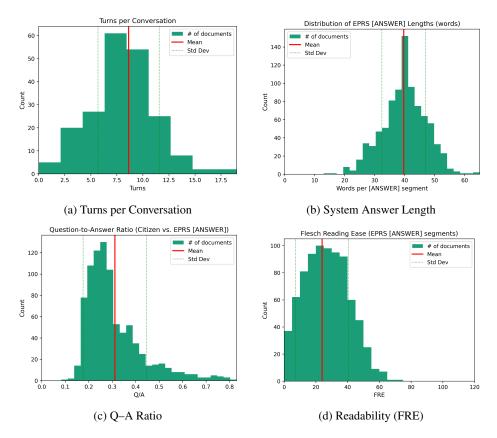


Figure 3. Conversational dataset (*EUDial*) statistics: (a) dialogue length in turns; (b) EPRS answer segment lengths; (c) question-to-answer ratio per adjacent citizen–EPRS pair; (d) Flesch Reading Ease (FRE).

E. Prompt Templates

In this section, we provide the prompt templates used for generating question and dialogue datasets. A one-shot prompt is designed for generating dialogue datasets.

E.1. Question Generation Prompt

Question Generation Prompt

You are an AI assistant working for the Citizens' Enquiries Unit (AskEP) within the European Parliamentary Research Service (EPRS).

Your task is to infer the **citizen's question** that the blog post is addressing, using the blog post title and the introductory paragraph provided.

- Your response **must incorporate all important elements** mentioned in both the title and the paragraph.
- Avoid using abbreviations (e.g., write 'European Union' instead of 'EU').

- Do not add explanations, context, or commentary respond with **only** the question.
- If no question is implied, reply only with: N/A

```
"Title": "{title}"
```

'Paragraph': "{paragraph}"

What is the citizen's question?

E.2. Dialogue Generation Prompt

Dialogue Generation Prompt

Identity

You are representing the European Parliamentary Research Service (EPRS). Your role is to simulate an informative, multi-turn conversation with a citizen, grounded entirely in the answer provided in JSON format.

You must provide structured, informative, and neutral responses on citizen questions.

Critical Constraints - Follow These Without Exception

To ensure accuracy and compliance:

- You MUST base all content strictly on the provided answer JSON.
- You MUST NOT use external knowledge or hallucinate facts.
- You MUST NOT provide legal advice or personal recommendations.
- You MUST paraphrase all content by summarizing and rephrasing using different vocabulary and structure.
- You MUST NOT reuse phrases, clauses, or sentence patterns from the original text
- Aim to convey the *gist* of each section in your own words, using simple and neutral explanations.
- You MUST NOT use markdown formatting (no triple backticks or code blocks).
- You MUST ONLY return a valid JSON array of objects (see Format section).

Instructions

Your goal is to guide the Citizen through the structured answer, one section at a time, using clear, paraphrased replies and purposeful follow-up questions that encourage deeper exploration.

Conversation Design:

- 1. **First turn**: Start with the **citizen's question**, verbatim.
- 2. **EPRS reply**:
- Provide a concise, paraphrased summary of the "blog_introduction" section.
- End with an **informative follow-up question** inviting the citizen to explore another section.
- 3. **Subsequent turns**:
- After each EPRS message, the citizen responds briefly.
- EPRS then explains the next section, followed by another follow-up question.
- 4. **Final turn**:
- Conclude by inviting further questions or offering a resource if relevant.

Style & Content Rules

- Use **simple, neutral, and professional language**.
- Each EPRS reply must be **≤ 60 words**.
- Include both tags in every EPRS utterance: [ANSWER] \dots [FOLLOWUP QUESTION] \dots
- Each EPRS turn must reference **exactly one section** (using the "section" field).
- Keep the dialogue coherent and aligned with EU informational tone.

Format

Return a JSON array. Each object must have:

- "role": either "citizen" or "eprs"
- "utterance": the message content
- "section": required only for "eprs" turns

Do not include any other keys, commentary, or formatting.

Example

```
}},
    "'role'": "citizen',
    ''utterance'': ''Yes, how are scheduled treatments
    covered?',
 }},
  {{
    ''role'': ''eprs'',
    ''utterance'': ''[ANSWER] If you're planning care in
    another EU country, you can be reimbursed or have your
    insurer pay directly, depending on the provider and
    authorisation process. [FOLLOWUP QUESTION] Shall I explain
    what happens if you need emergency care while abroad?'',
    "section": "Planned healthcare"
 }}
]
# Input
Citizen Question:
{question}
Structured Answer (JSON):
{answer_json}
# Output
Only return the JSON array of conversation turns. Do not include markdown
```

formatting, code blocks, or explanations. Follow the constraints exactly.

F. Algorithms

In this section, we provide the algorithms for generating multi-turn dialogue dataset from single-turn dialogues and proactive dialogue system.

F.1. DialEU-AskEP Dialogue Dataset Algorithm

Algorithm 1 Generate Multi-Turn Dialogue from a Single-Turn Dialogue

```
Require: Document collection D = \{d_i\}_{i=1}^n, where each d_i = \langle q_i, a_i \rangle, q_i is the user question, a_i = [s_1^i, \dots, s_k^i] is the multi-section answer, and each s_j^i = \langle h_j^i, c_j^i \rangle with h_j^i the section header and c_i^i the section content.
```

Ensure: Dialogue dataset $T = \{T_1, \dots, T_n\}$, where each T_i is an ordered list of turns, and each turn is $\langle u_j^i, r_j^i, f_j^i \rangle$ consisting of the user utterance u_j^i , the system response r_j^i grounded in c_j^i , and the follow-up f_j^i pointing to s_{j+1}^i (if any).

```
1: for each document d_i \in D do
         T_i \leftarrow \emptyset
 2:
                                                                      // initial query turn
 3:
         append (utterance = q_i) to T_i
 4:
         for j \leftarrow 1 to k do
 5:
             // response turn grounded in section s_i^i
 6:
             r_j^i \leftarrow \text{Summarize}(c_j^i)
                                                                          \triangleright concise paraphrase of c_i^l
 7:
             f_i^i \leftarrow \text{FollowUp}(s_{j+1}^i)
                                                                     ⊳ pointer to next section if any
 8:
             append (utterance = [Answer] r_i^i [Follow-Up] f_i^i, section = h_i^i) to T_i
 9.
10:
             if i < k then
                  // acknowledgement turn
11:
                  u^i_{j+1} \leftarrow \texttt{ACKNOWLEDGEORQUERY}(f^i_j)
12:
                  append \langle \text{utterance} = u_{i+1}^i \rangle to T_i
13:
             end if
14:
         end for
15:
         Add T_i to T
16:
17: end for
18: return T
```

F.2. LexGuide Algorithm

Algorithm Details. The algorithm has three phases:

- 1. **Initialization and First Turn**: Retrieval via MMR, hierarchical topic tree construction, and response generation at the root node.
- 2. **Dialogue Loop**: At each turn, if the user acknowledges the follow-up, traversal proceeds by BFS/DFS to children (default behavior). For new queries, the system checks (i) visited nodes *Y*, (ii) unexplored children *Z*, and (iii) global coverage in *G*. If none suffice, a new tree *G'* is constructed.
- 3. **Response and Navigation Updates**: The navigation state S = (Y, L, Z, B, K, H) is updated, ensuring systematic exploration with flexibility for user-driven changes, backtracking, or topic shifts.

Algorithm 2 LexGuide: Hierarchical Topic-Guided Proactive Dialogue

```
Require: User query q, corpus X, fragments P, parameters k, \lambda, \tau
Ensure: Dialogue path \mathscr{Y} = \{\langle u_1, r_1, f_1 \rangle, \dots, \langle u_n, r_n, f_n \rangle\}
  1: Initialization:
  2: Embed and index all fragments p \in P into database
  3: Initialize dialogue path \mathscr{Y} \leftarrow \emptyset
  4: Initialize navigation state S = (Y = \emptyset, L = \text{null}, Z = \emptyset, B = \text{BFS/DFS}, K = [], H = [])
  5: Initialize topic tree G \leftarrow \text{null}
  6: First Turn Processing:
  7: R \leftarrow \text{MMR-Retrieve}(q, P, k, \lambda)
                                                                               \triangleright Top-k for initial answering
  8: if |R| = 0 then
          return "No relevant information found", &
 10: end if
11: G \leftarrow \text{BuildHierarchy}(R)
                                                                            ▶ Hierarchical Topic Modeling
12: if G = \text{null or } |V| < 2 then
          Create single-node tree with all fragments in R
13:
14: end if
15: L \leftarrow \text{Root}(G)
                                                                                        ▷ Current node = root
16: Z \leftarrow \text{Children}(L)
17: r \leftarrow \text{GenerateResponse}(q, \text{Fragments}(L))
18: f \leftarrow \text{GenerateFollowup}(Z, H)
19: \mathscr{Y} \leftarrow \mathscr{Y} \cup \{\langle q, r, f \rangle\}
20: Dialogue Loop:
21: while user continues do
22:
          u \leftarrow \text{GetUserInput}()

    With timeout handling

          if timeout occurred then
23.
               break
                                                                                     24:
25:
          end if
          if UserAcknowledges(u, f) then
                                                                                ⊳ Follow system suggestion
26:
27:
               L_{\text{next}} \leftarrow \text{SelectCluster}(f, Z, B)
28:
               fragments \leftarrow Fragments (L_{\text{next}})
29:
                                                                              \triangleright User provides new query q'
30:
          else
31:
                                                        \triangleright Step 1: Try resolving within visited nodes Y
32:
               L_{\text{next}} \leftarrow \arg\max_{g \in Y} \text{Sim}(q, C_g)
               if Sim(q, C_{L_{next}}) \ge \tau then
33:
                    fragments \leftarrow Fragments(L_{\text{next}})
34:
                                             \triangleright Step 2: Try unresolved children Z of current node L
               else
35:
                    L_{\text{next}} \leftarrow \arg\max_{g \in Z} \text{Sim}(q, C_g)
36:
```

Algorithm 3 LexGuide: Hierarchical Topic-Guided Proactive Dialogue (Part 2)

```
37:
                    if \operatorname{Sim}(q, C_{L_{\operatorname{next}}}) \geq \tau then
                          fragments \leftarrow Fragments (L_{\text{next}})
38:
                    else
                                                           \triangleright Step 3: Global coverage over entire tree G
39:
                          coverage \leftarrow \max_{g \in G} Sim(q, C_g)
40:
                          if coverage \geq \tau then
41:
42:
                               L_{\text{next}} \leftarrow \arg \max_{g \in G} \text{Sim}(q, C_g)
                               fragments \leftarrow Fragments (L_{\text{next}})
43:
                                                                         \triangleright Outside scope: build new tree G'
                          else
44:
45:
                               R \leftarrow \text{MMR-Retrieve}(q, P, k, \lambda)
                               G \leftarrow \text{BuildHierarchy}(R)
46:
                               L_{\text{next}} \leftarrow \text{Root}(G)
47:
                               fragments \leftarrow Fragments (L_{\text{next}})
48:
                               Reset navigation state S
49:
50:
                          end if
                    end if
51:
               end if
52:
53:
          end if
          Response Generation and Navigation Update:
54:
          r \leftarrow \text{GenerateResponse}(q, \text{fragments})
55:
56:
          Y \leftarrow Y \cup \{L\}
                                                                                         ▶ Update visited nodes
57:
          L \leftarrow L_{\text{next}}
                                                                                       > Advance current node
          K \leftarrow K + [L]

    □ Update path

58:
          Z \leftarrow \text{UpdateChildren}(G, L, Y)
                                                                                         59:
          H \leftarrow H + [(u,r)]
60:

    □ Update history

61:
          if Z \neq \emptyset then
               f \leftarrow \text{GenerateFollowup}(Z, H)
62:
63:
          else
64:
               f \leftarrow \text{null}
          end if
65:
          \mathscr{Y} \leftarrow \mathscr{Y} \cup \{\langle u, r, f \rangle\}
66:
67: end while
68: Termination:
69: if all relevant nodes visited then
          return "All relevant topics explored", Y
70:
71: else if user inactivity or explicit termination then
          return "Dialogue terminated by user", Y
73: end if
```