A Survey on Recent Advances in Conversational Data Generation

HEYDAR SOUDANI, Radboud Universiteit, Nijmegen, Netherlands
ROXANA PETCU, University of Amsterdam, Amsterdam, Netherlands
EVANGELOS KANOULAS, University of Amsterdam, Amsterdam, Netherlands
FAEGHEH HASIBI, Radboud Universiteit, Nijmegen, Netherlands

Recent advancements in conversational systems have significantly enhanced human-machine interactions across various domains. However, training these systems is challenging due to the scarcity of specialized dialogue data. Traditionally, conversational datasets were created through crowdsourcing, but this method has proven costly, limited in scale, and labor-intensive. As a solution, the development of synthetic dialogue data has emerged, utilizing techniques to augment existing datasets or convert textual resources into conversational formats, providing a more efficient and scalable approach to dataset creation. In this survey, we offer a systematic and comprehensive review of multi-turn conversational data generation, focusing on three types of dialogue systems: open domain, task-oriented, and information-seeking. We categorize the existing research based on key components like seed data creation, utterance generation, and quality filtering methods, and introduce a general framework that outlines the main principles of conversation data generation systems. Additionally, we examine the evaluation metrics and methods for assessing synthetic conversational data, address current challenges in the field, and explore potential directions for future research. Our goal is to accelerate progress for researchers and practitioners by presenting an overview of state-of-the-art methods and highlighting opportunities to further research in this area.

CCS Concepts: • **Computing methodologies** → **Natural language generation**; • **Information systems** → **Language models;**

Additional Key Words and Phrases: Conversational AI, dialogue system, data augmentation, conversation generation, task oriented dialogues, open domain dialogues, conversational information seeking

ACM Reference Format:
Heydar Soudani, Roxana Petcu, Evangelos Kanoulas, and Faegheh Hasibi. 2026. A Survey on Recent Advances in Conversational Data Generation. *ACM Comput. Surv.* 58, 10, Article 258 (April 2026), 37 pages. https://doi.org/10.1145/3795686

1 Introduction

Conversational AI focuses on enabling machines to interact with humans using natural language [151]. These systems, particularly enhanced by neural networks and deep learning, aim at

Authors' Contact Information: Heydar Soudani(corresponding author), Radboud Universiteit, Nijmegen, Gelderland, Netherlands; e-mail: heydar.soudani@ru.nl; Roxana Petcu, University of Amsterdam, Amsterdam, Noord-Holland, Netherlands; e-mail: r.m.petcu@uva.nl; Evangelos Kanoulas, University of Amsterdam, Amsterdam, Noord-Holland, Netherlands; e-mail: e.kanoulas@uva.nl; Faegheh Hasibi, Radboud Universiteit, Nijmegen, Gelderland, Netherlands; e-mail: F.Hasibi@cs.ru.nl.

ACM 0360-0300/2026/04-ART258
https://doi.org/10.1145/3795686

closely mimicing human conversational behaviors [25]. This enhancement is manifested in the system's ability to handle complicated challenges, such as discussing specialized topics [71, 104, 111], shifting between conversation topics [3], asking proactive questions [139], and integrating emerging topics [120]. To develop systems capable of addressing these sophisticated challenges, a robust and large-scale source of training data is essential.

Crowdsourcing is the primary method for creating conversational datasets, where human workers generate data based on provided instructions [13, 19, 36, 37, 73, 83, 103, 109, 119, 142, 153]. This approach, however, is costly, time-consuming, and challenging to scale or adapt to new domains [46, 107, 116]. It also introduces annotation biases [144] and is ineffective for emerging topics, especially in specialized fields or languages with limited development resources [66, 121]. As an alternative, generating synthetic conversational datasets has emerged as a promising solution. This method involves transforming existing textual resources–like documents, tables, and knowledge graphs–into conversational formats or augmenting existing dialogue data with new instances [120]. Synthetic conversation generation is especially beneficial in resource-scarce domains, enabling models to leverage all available resources to enhance learning [44, 98]. Data Generation denotes the process of expanding or enhancing a dataset by creating entirely new data points, as compared to data augmentation, which involves applying transformations or modifications to existing data points. Generation results in more robust and diverse training samples as compared to augmentation, where the new data points are not present in the original dataset, however, are plausible and relevant.

The history of conversational AI traces back to the 1960s with the creation of the first chatbot, Eliza [133]. Eliza operated on rule-based principles, generating responses based on predefined rules tied to specific keywords or facets. Following this, the rise of systems that integrate rule-based and machine learning components has been observed [25, 94], exemplified by platforms such as Apple's Siri, Amazon's Alexa, and Microsoft's Xiaobing. These systems have found commercial and industrial applications, relying on machine learning algorithms to improve response accuracy by learning from historical user interactions and conversation data [95]. Further advancements occurred with the advent of deep learning-based systems, leveraging large models and extensive datasets to enhance their capabilities [94]. Recently, the emergence of **Large Language Models (LLMs)** has taken this a step further. These models excel in generating contextually relevant, informative, and diverse responses that mirror human language, thanks to their ability to analyze vast amounts of data and the sophistication of their underlying architectures [25, 147].

Dialog systems can be categorized into three paradigms: **Task-oriented Dialogs** (**TOD**), **Open Domain Dialogs** (**ODD**), and Conversational **Information Seeking** (**CIS**) [27, 120, 151]; see the electronic appendix for examples. Traditionally, these systems have been distinguished based on methodological differences. However, with the advancements of LLMs, the boundaries between them are increasingly blurring. The first paradigm, TOD, seeks to execute tasks at the user's request by accurately identifying their intentions and providing appropriate responses [131]. TOD systems are built to comprehend user queries and generate responses that help achieve the user's objective. They find applications in various domains and use cases, such as flight booking, restaurant reservations, hotel accommodations, taxi services, movie ticket purchases, weather inquiries, navigation assistance, scheduling, and customer service [35, 134]. TOD faces multiple challenges, such as ensuring robust and reliable dialogue state tracking [35], integrating external databases for extracting specific conversational entities [143], e.g., restaurant names, flight numbers, and cinema schedules, and optimizing response generation for user satisfaction and task completion [136]. With a focus on these challenges, we propose a taxonomy for TOD generation based on four steps: identifying input sources as external knowledge, such as knowledge bases, schemas, and ontologies, or internal, such as knowledge extracted from training data, dialogue generation methods, training, and quality filtering. This multi-faceted analysis offers an intuition of TOD generation, emphasizing

the critical roles of input handling, generation and training approaches, and quality control in their development and deployment.

The second paradigm, ODD, aims at facilitating casual conversations with users across a broad range of topics and domains, without being confined to specific tasks or objectives [61, 94]. ODD systems encounter several challenges, such as maintaining coherent and contextually appropriate responses [81], ensuring response diversity to engage users [157], displaying proactivity by steering discussions toward certain topics [18, 79, 145], personalizing responses based on user profiles [153], and generating informative replies from external knowledge bases [78]. Addressing these challenges requires systems to be trained with datasets that specifically include them, making synthetic data generation a promising approach. This survey proposes a taxonomy for synthetic dataset creation, structured in three steps: seed generation, turn generation, and quality filtering. This taxonomy is introduced to abstract choices in existing methods in the literature to produce diverse, informative [16, 61], multi-skill [62], and personalized conversational data [54, 66].

The third paradigm, CIS, is designed to assist users in seeking and retrieving information through natural language dialogue interactions. The primary objective of a CIS system is to satisfy the information needs of users by engaging in dynamic conversations, which may include text as well as other modalities such as voice, clicks, or touch [151]. CIS encompasses three main areas: conversational search, conversational **question answering** (**QA**), and conversational recommendation. *Conversational Search* and *Conversational QA* involve interacting with a system in natural language to find specific information, allowing users to pose multiple questions based on their interaction history, with the system providing relevant answers [2, 138]. *Conversational Recommendation Systems* suggest items to users based on their previous interactions, serving as personalized information-seeking tools [83, 130]. CIS systems face several challenges, including maintaining the conversation state to better respond to user needs, managing mixed-initiative interactions where the system sometimes leads the conversation and sometimes responds to user queries, and adapting to user preferences and profiles to enhance satisfaction by personalizing the conversation. Based on these challenges, tasks such as intent prediction, asking clarification questions, target-oriented recommendation, and topic switching are defined. These tasks help clarify uncertainties, lead conversations toward specific items, and adapt the discussion to cover various subjects and entities based on the user's informational needs [3]. Effective handling of these tasks requires a substantial amount of well-curated data for training the models. In this work, we systematically review the literature on information-seeking conversational datasets and provide a three-step framework similar to those used for ToD and ODD. The three-step framework used in each subsection is illustrated in Figure 2.

In this work, we also differentiate between LLM-based and non LLM-based approaches. Specifically, we define ***Language Models (LMs)*** as encoder-only, decoder-only, or encoder–decoder architectures, typically with fewer than one billion parameters. In contrast, we define *LLMs* as foundation models trained on broad, general-purpose corpora with more than one billion parameters, and capable of generalization through prompting, few-shot learning, or **in-context learning** (**ICL**). Based on these definitions, we categorize DA approaches into the following groups and provide detailed coverage of these categories in the corresponding subsections.

— LLM-based approaches: Methods that rely on prompting or in-context learning of LLMs (such as GPT-3, PaLM, or LLaMA) to generate full dialogue responses without supervised fine-tuning.
— Non LLM-based approaches: Methods that use LMs, such as BART and RoBERTa, or do not use LMs. This category also includes methods that use LLMs only for language generation but not for generating the core structure of the dialog; e.g., methods that create TOD intents from a knowledge base but generate language around them using LLMs.

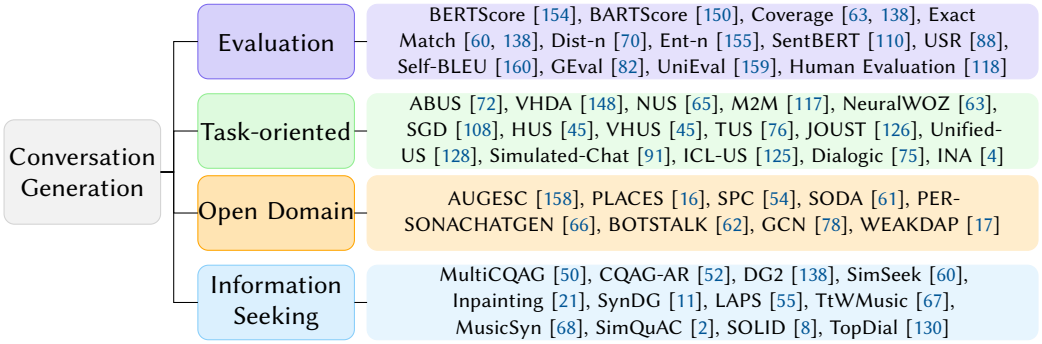

Fig. 1. An overview of multi-turn conversation generation sections and articles.

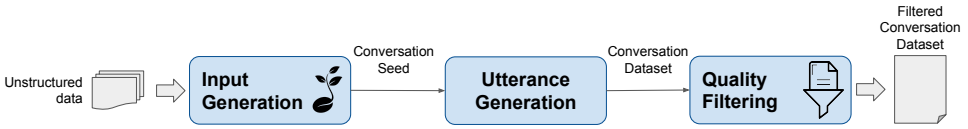

Fig. 2. The general framework for conversation generation in TOD, ODD, and CIS systems. This framework consists of three main components: **(1) Input Generation** creates a dialog seed , **(2) Utterance Generation** uses the dialog seed as an input to a model and generates a multi-turn conversation sample, and **(3) Quality Filtering** eliminates dialogs that do not achieve a desirable representation; filtering can be *lexical* or *consistency* related.

Figure 1 presents an overview of the article's structure. The contributions of this work are summarized as follows:

- We present a comprehensive review of the methodologies employed for creating synthetic conversational data, highlighting the innovative techniques and processes that facilitate their generation.
- We introduce a general framework consisting of three steps: input generation, utterance generation, and quality filtering, which are used to abstract and describe dialogue data generation across three distinct types of dialogue systems: open-domain, task-oriented, and information-seeking.
- We offer a detailed description of data generation evaluation methods, detailing the metrics and criteria used to assess the quality, relevance, and utility of conversational datasets.
- We discuss prospective areas of focus and recent research challenges that have emerged due to the growing demands for data generation.

2 Evaluation

Before exploring data generation methods, it's essential to understand how we can evaluate the outcomes of dialogue generation methods. While this article does not focus on conversation evaluation, we find it necessary to provide a comprehensive overview of how generated conversations are evaluated. There are primarily two approaches for assessing synthetically generated conversational data: ***extrinsic*** and ***intrinsic*** evaluation, described below.

Extrinsic Evaluation measures the quality of the generated data based on the performance of downstream tasks they are used for. Given that the ultimate purpose of synthetic conversational data generation is to augment training data and improve the performance of conversational agents,

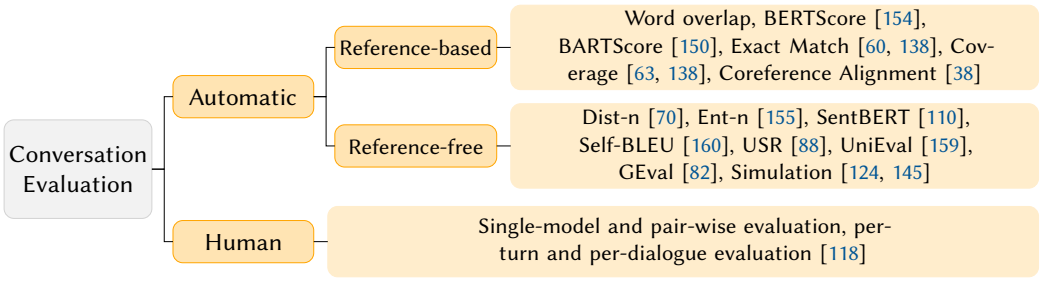

Fig. 3. An overview of methods for evaluating generated multi-turn conversations.

a dialogue model is trained using all or part of the synthetic data and the final performance of the model is used to indirectly assess the quality of the data generation model [60, 140]. For example, AUGESC [158] and SODA [61] aim at creating an ODD conversational dataset for fine-tuning an emotional support conversational system and a social conversational agent, respectively, using synthetically generated data. Similarly, within the CIS system, SOLID [8] focuses on enhancing the intent prediction task with a portion of the generated data and Inpainting [21] highlights that its generated dataset serves as a valuable training resource for ConvQA systems.

Intrinsic Evaluation measures synthetic data quality directly using human or automatic evaluation metrics, assessing various qualities of the generated conversations, such as naturalness, understandability, and coherence. The quality of the generated dialogue is directly evaluated either automatically using various evaluation metrics and through human evaluation. Automatic evaluation metrics of conversations are designed to either compare the model's output against a reference dataset (ground truth) or assess it based on inherent characteristics without reference. To demonstrate generality, existing studies often employ a combination of both intrinsic and extrinsic evaluation methodologies. In this section, we review existing methods for evaluating conversations, generated either during the data augmentation process or by a dialogue model. An overview of these methods is presented in Figure 3 and Table 1.

2.1 Automatic Evaluation

Conversations can be evaluated automatically using reference-based and reference-free approaches.

2.1.1 Reference-based. These methods assess the quality and relevance of generated text by comparing it against a set of predefined reference texts. These metrics are designed to measure how closely the generated dialogue aligns with expected responses, providing insights into the accuracy and coherence of the conversation. Dialogue data evaluation using these methods can be categorized into three main groups based on the metrics employed.

The first group, *Word Overlap Metrics,* assesses the n-gram similarity between the generated text and the ground truth, focusing on the overlap of word sequences to evaluate the quality of generated questions against a predefined standard. Some examples of these metrics include BLEU (1-3), ROUGE-L (R-L), and METEOR.

The second group, *Embedding-based Metrics*, relies on a score derived from comparing the embeddings of the generated text and the ground truth. Two examples from this group are BERTScore [154] and BARTScore [150]. BERTScore uses BERT-based contextual embeddings to compute cosine similarity between words in reference and generated texts [33]. By measuring semantic similarity rather than exact lexical overlap, it better captures meaning beyond word choice or order. BARTScore leverages the BART sequence-to-sequence model [69] to evaluate text using a generative scoring

Table 1. Overview of Evaluation Metrics for Conversational Systems, Categorized to Reference-based and Reference-free Metrics

Metric	Category	Description
Exact Match [138]	Ref-based	A strict accuracy metric that measures the percentage of predictions that match the reference text exactly, with no partial credit.
BLEU	Ref-based	A precision-based metric that measures the overlap of n-grams between a generated text and reference texts, emphasizing exact word matches.
ROUGE-L	Ref-based	A recall-oriented metric that evaluates the longest common subsequence between a generated text and reference texts, capturing sentence-level structure similarity.
METEOR	Ref-based	A metric that scores text similarity based on unigram matches, stemming, synonyms, and word order, balancing precision and recall.
BERTScore [154]	Ref-based	A semantic similarity metric that computes token-level cosine similarity between contextual embeddings from BERT for generated and reference texts.
BARTScore [150]	Ref-based	An evaluation metric that uses the log-likelihood from a pretrained BART model to assess the quality of generated text given the reference or source text.
Dist-n [70]	Ref-free	A diversity metric that calculates the ratio of distinct n-grams to the total number of generated n-grams, measuring lexical diversity.
Ent-n [155]	Ref-free	A diversity metric that computes the entropy of the n-gram distribution in generated text, capturing the unpredictability and variety of word usage.
SentBERT [110]	Ref-free	A semantic similarity metric that uses sentence embeddings from BERT to measure the cosine similarity between generated and reference texts.
Self-BLEU [160]	Ref-free	A diversity metric that computes BLEU scores between each generated sentence and the rest of the generated sentences, where lower scores indicate higher diversity.
USR [88]	Ref-free	An unsupervised and reference-free metric that evaluates dialogue quality based on user satisfaction through multiple model-based evaluators (e.g., relevance, fluency, and engagement).
UniEval [159]	Ref-free	A unified evaluation framework that uses a single model to assess multiple aspects of text generation quality (such as relevance, coherence, and fluency) in a reference-free or reference-based manner.
G-EVAL [82]	Ref-free	An LLM-based evaluation framework that leverages GPT models with carefully designed prompts to assess multiple dimensions of text quality, such as coherence, consistency, and relevance.

approach. It treats evaluation as a generation task, computing scores from the likelihood of generating the reference from the candidate (or vice versa). Compared to BERTScore, which focuses on semantic similarity using embeddings from BERT, BARTScore leverages the generative capabilities of BART to assess text quality from a more holistic perspective, incorporating fluency, coherence, and even factual accuracy [150, 154].

The last group, *Subtask Evaluation Metrics*, assess how well dialogue models handle specific components of conversation generation. For example, Exact Match [138] requires the predicted span to exactly match the reference span, and is mainly used in document-grounded generation. Since this criterion is often too strict for long conversational answers, F1 word overlap is commonly used instead [60]. Another metric is Coverage [63, 138], which is used in both ToD and CIS conversation generation. In CIS, Span Coverage [138] assesses how effectively a model captures the rationale spans within a document. The premise here is that dialogues covering a larger portion of the document indicate a more effective data augmentation method. Span Coverage is calculated by measuring the proportion of the document that the generated dialogue spans cover. In the context of ToD, zero-shot coverage [63] is evaluated, which measures the accuracy ratio between zero-shot learning outcomes in a target domain and a fully trained model that includes that domain. Another metric in this group is Coreference Alignment [38], which is critical in conversational QA where using coreferences, such as pronouns “he” and “she”, is common–almost half of the questions include explicit coreference markers [109]. Specifically, coreference alignment modeling instructs the decoder to focus on the correct non-pronominal coreferent mention in the conversation’s attention distribution to accurately generate the pronominal reference word [38]. To evaluate this

feature, the Precision (P), Recall (R), and F-score (F) of pronouns in the generated questions are calculated and compared with those in the ground truth questions.

2.1.2 Reference-free. These methods assess the generated dialogue without comparing them to ground truth data, evaluating inherent properties such as diversity [70, 155], semantic coherence [110], and user satisfaction [88]. These methods provide a flexible evaluation of the dialogue's originality and engagement, allowing for a broader interpretation of quality without the constraint of matching to a specific ground truth.

For diversity-based metrics, Dist-n [70] calculates the diversity of generated text by determining the ratio of unique unigrams and bigrams to the total number of words generated [100]. Ent-n [155] measures the uniformity of the n-gram distribution across all generated text, revealing the variability in n-gram usage. Another metric for evaluating diversity is Self-BLEU [160]. While BLEU typically measures the similarity between two sentences, Self-BLEU uses one sentence from a set as a hypothesis and the rest as references, calculating a BLEU score for each sentence. The average of these scores is termed Self-BLEU. A higher Self-BLEU score indicates lower diversity among the sentences. Sent-BERT [110] measures semantic diversity by computing the average negative cosine similarity between SentenceBERT embeddings of response pairs. We note that diversity metrics are length-sensitive and often yield higher scores for shorter texts due to reduced word repetition. To mitigate this effect, LAPS [55] applies a word-count cutoff across datasets.

USR [88] is a reference-free dialogue evaluation metric composed of five sub-metrics: Understandable (response coherence), Natural (human-likeness), Maintains Context (conversation continuity), Interesting (engagement), and Uses Knowledge (appropriate factual content). A regression model trained on human judgments combines these components into an overall dialogue quality score. UniEval [159] is another reference-free, aspect-based NLG evaluator that uses T5 to unify multiple evaluation dimensions. It frames each aspect (e.g., coherence, consistency) as a Boolean QA task. For example, it asks whether a summary is coherent with its source document. UniEval is trained on four NLG task types, producing one evaluator per task. For dialogue generation, it follows USR and evaluates Naturalness, Coherence, Engagingness, and Groundedness. G-EVAL [82] evaluates NLG outputs by prompting an LLM with only a task description and evaluation criteria. The model then generates a chain-of-thought outlining evaluation steps, which, together with the prompt, is used to assess the output and produce a form-style rating. The metric can also incorporate the probabilities of the rating tokens to refine the final score.

Simulation is another form of dialogue evaluation, which is primarily used for assessing target-guided open-domain dialogue systems [124, 145]. This method involves two dialogue agents engaging in a conversation, after which the success rate in reaching the predefined target is automatically calculated.

2.2 Human Evaluation

The analysis of a conversational model's performance becomes more comprehensive with the inclusion of human evaluations [118]. Although automatic metrics assess certain aspects of model performance and offer speed, efficiency, and reproducibility, they often correlate weakly with human judgment and may not capture all the nuances of conversational competence [30]. In human evaluation, raters evaluate how effectively models manage realistic and engaging conversations [28]. However, human evaluations have their own limitations. A significant challenge is the lack of comparability across different studies due to variations in experimental settings. These variations can include differing criteria such as naturalness, informativeness, context relevance, and answer accuracy.

Various methods are used for human evaluation of conversations. Smith et al. [118] categorize these methods along two features: per-turn vs. per-dialogue, and pairwise vs. single-model evaluation. Per-turn ratings provide fine-grained analysis, while per-dialogue evaluations better capture overall coherence. Pairwise comparisons highlight subtle differences and reduce scoring bias, whereas single-model evaluations are suitable when direct comparison is unnecessary. Five evaluation techniques have been used by combining these features:

— Pairwise Per-Turn Evaluation (PW-Turn): This technique provides annotations for every turn of a conversation. It requires a crowdworker to choose between a pair of model responses after each message.
— Pairwise Per-Dialogue Evaluation (PW-Dialog): This method asks evaluators to choose between two models by presenting a pair of complete conversations.
— Pairwise Per-Dialogue Self-Chat Evaluation: This approach compares two conversations that involve only the two bots talking to each other, assessing their interaction without human input.
— Single-Model Per-Turn Evaluation (SM-Turn): In this technique, a crowdworker interacts with a conversational agent powered by a single model. The worker annotates each response from the model based on engagement, human-likeness, and interest.
— Single-Model Per-Dialogue Evaluation (SM-Dialog): Similar to SM-Turn, this method involves evaluating a single model's performance over an entire conversation, with the assessment occurring after the conversation has concluded.

After comparing various evaluation techniques, Smith et al. [118] draws several conclusions. Firstly, the PW-Turn are effective when differences in models' responses are easily noticeable, such as when models are trained on different datasets. Secondly, PW-Dialog are most effective when differences between models become apparent over several conversation turns, for example, differences in the average length of conversations. Lastly, single-model evaluations are well-suited for comparing models that are similar but vary slightly in quality, such as models with different numbers of parameters.

3 Task-oriented Conversational Data Generation

TOD represent a specific type of conversational data that contains structured interactions aimed at accomplishing specific tasks or objectives, such as booking a flight ticket or making a restaurant reservation [35, 134]. These dialogues play a pivotal role in areas such as customer service and virtual assistance, where they must adhere to particular requirements and constraints. For instance, successfully booking a restaurant table requires verifying the location's availability, matching the cuisine with the user's preference, and securing a reservation that accommodates the size of the user's party.

TOD dialogues are often used in areas such as customer service and virtual assistance, and consequently, they are constructed around an **entity** such as *Restaurant*, *Customer*, or *Movie*. As a result, factuality is a very important aspect of TOD systems, meaning all generated entity-centric facts must be constructed or verified using existing **predefined knowledge**. The **predefined knowledge** used for fact verification is represented using complex graphical structures, which in addition to entities, also contain information on the relationship between them. The relationship between entities shows dependencies of the task, for example, a *customer* must have at least one attribute clearly defined, such as *name*, before making a *reservation*. Relationships between entities are crucial for generating dialogue sequentially and logically, enhancing the system's ability to manage complex interactions and dependencies effectively.

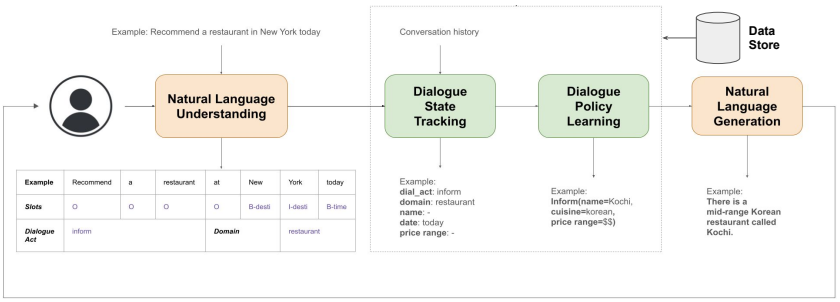

Fig. 4. Architecture of a TOD system, composed of four modules communicating in a pipeline-fashion. The input and output of each module are illustrated, indicating the process of receiving a user conversational turn, identifying intents, slots and values, extracting response attributes from the data store, and generating them into a natural language response.

TOD Pipeline Modules. TOD systems are mainly composed of four modules, namely Natural Language Understanding, Dialogue State Tracking, Dialogue Policy Learning, and Natural Language Generation. These modules can be either compressed into an end-to-end pipeline, or often follow a modular approach as shown in Figure 4. We will describe the function, input, and output of each of these modules:

- **Natural Language Understanding (NLU).** This module receives as input a conversational turn in natural language form. The goal is to process the input and extract intents, slots and values for the identified slots.
- **Dialogue State Tracking (DST).** This module receives as input the conversation history and output of the NLU module (which corresponds to the current turn of the dialog) and produces the necessary slots that should be filled to approach the user goal.
- **Dialogue Policy Learning (DPL).** This module receives as input the slots that must be filled in, and outputs values that would be satisfactory next actions based on the current dialogue state.
- **Data Store.** When predefined knowledge is provided to the system (in the form of a Database, Knowledge base, ontology, or schema), both the DST and DPL modules extract slots and values from these sources. If this predefined knowledge is not provided, extra modules are added to learn how to extract these slots and values from provided training data.
- **Natural Language Generation** receives as input the DPL output, and converts it into natural language representation.

Key terms in TOD. Having introduced the main graphical representations used for entity-centric dialogue generation, we will now define additional key terms that will recurrently appear throughout this section (see an illustrative example in the electronic appendix).

- **Intent.** Represents the possible actions that a user can pursue. Intents are task-specific. Examples are *Book Movie*, *Reserve Restaurant*, *Set Alarm*.
- **Dialogue Act.** A label showing the speech act of a conversational turn; A dialogue act can be associated to either the system or the user, i.e., they can be referred to as **System Act** or **User Act** respectively. Common values are *Inform*, and *Request*.
- **(User) Goal.** A set of slot and slot values that indicate what the user wants to achieve from the conversation. Example: <date="tomorrow", time="8PM", restaurant="LaCongerie", cuisine="french", party_size="6">.

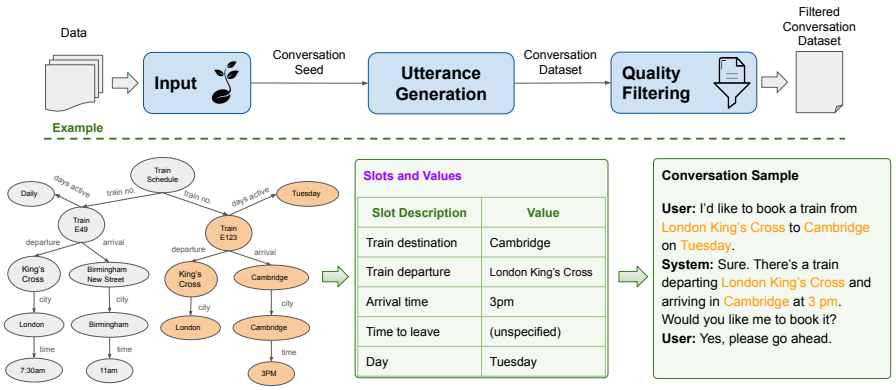

Fig. 5. Overview of data generation in TOD. In this example, the system has access to a knowledge base containing predefined slots and real-world values (e.g., *departure*, *destination*, *time*), which are sampled or extracted as input. The system uses the sampled slots and values to generate natural language utterances. While many methods rely on predefined templates or structured values, not all do; alternative approaches can instead learn slot–value structures directly from training data.

- **Dialogue Frame.** A tuple containing a dialogue act and a slot-value map. Example: *Inform*<date="tomorrow", time="8PM", restaurant="LaCongerie", cuisine="french">.
- **Belief State / Dialogue State.** A tuple of dialogue acts (*Inform*, *Request*) that keeps track of what information has been achieved, and what yet has to be requested in order to reach the user goal. *Inform* represents a set of slots and slot values that indicate the user constrains. *Request* represents a set of slots, without the values, that indicate what the agent wants to extract in the remainder of the dialogue. Example: *Inform*<date="tomorrow", time="8PM", restaurant="LaCongerie", cuisine="french">, *Request*<party_size>.
- **Slot and Value.** A domain or task-specific predefined attribute and its associated piece of information that are monitored using dialogue state tracking. An example of slots and slot values for the restaurant reservation task is the following: *{party_size: two people, cuisine: french, date: January 25th, time: 7:00 PM}*.

Having defined the key components and notation of TOD systems, we outline three stages in the TOD generation process: (1) input, (2) utterance generation, and (3) quality filtering, as shown in Figure 5. In Section 3.1, we provide an overview of different resources that are used as input for TOD systems. We then describe data generation methods for TOD systems in Section 3.2. We first split TOD approaches with respect to the external knowledge available, i.e., the input of the data generation system in the form of slots and their possible values. As presented in Section 3.1, this external knowledge appears in the form of a knowledge base, database, schema, or ontology. If provided, the corresponding slots and slot values are plugged into the dialogue system, and natural utterances are generated around them (see Sections 3.2.1–3.2.3). If they are not provided, the dialogue system learns them through training data (see Sections 3.2.4–3.2.6). However, it is important to note that generating slots and values from the mapped latent space does not guarantee factual/truthful generations. Lastly, we describe the quality filtering step in Section 3.3. An overview alongside existing work is presented in Figure 6.

3.1 Input

There are several key graphical representations that TOD systems need as their input: (1) the **Schema** defines the entities, their attributes (slots) and which attributes link the entities; (2) the

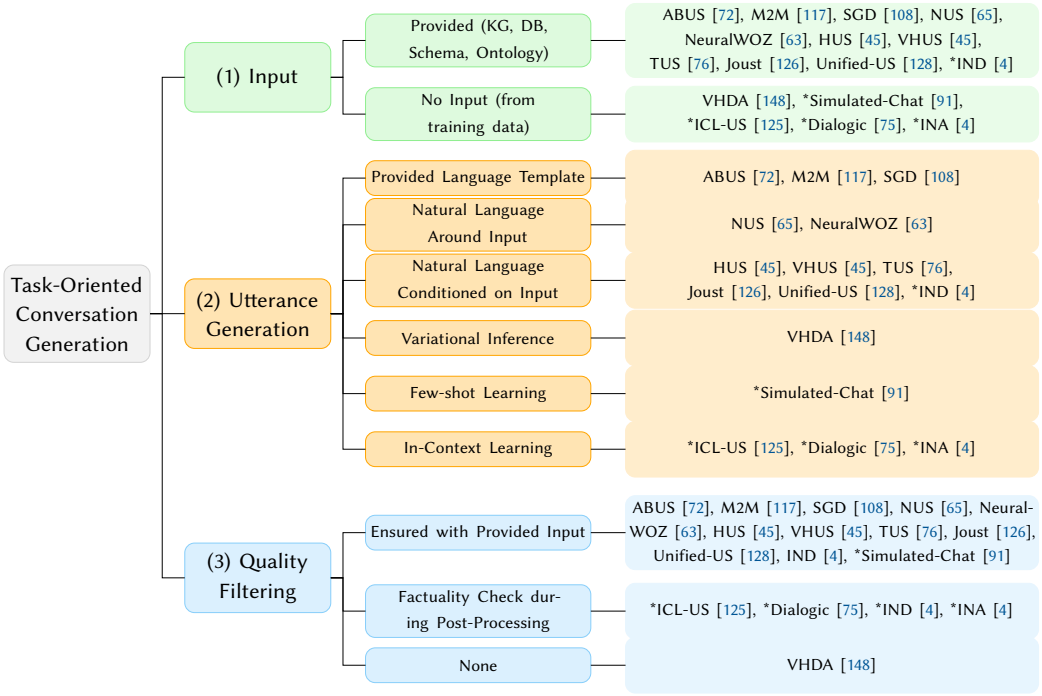

Fig. 6. Overview of approaches used for Task-Oriented Dialogue generation. The generation process is composed of three components, described in Sections 3.1 - 3.3. The models marked with an asteriks * are LLM-based.

Ontology builds on top of the schema by adding information on the relationship between entities, and is often used for querying a knowledge graph; (3) the **Knowledge Graph (KG)** or **Data Base (DB)** serves as a repository of domain-specific information. The KG is an instantiation of an ontology with all available entities, while the DB is an instantiation of a schema. Understanding these elements is essential for developing robust and efficient TOD systems. We will outline each of these components, as they will re-appear in some of the methodologies we further discuss in this study. Concrete examples of schemas, ontologies, and knowledge graphs used in TOD systems are provided in the electronic appendix.

(1) **Schema and Ontology**. These are general structures that describe entities, such as *Restaurant* and *Reservation*, and entity attributes, such as *Location* for *Restaurant*, and *Party Size* for *Reservation*. We differentiate between two commonly used structures:
Schema: Represents a set of entities, where each entity has attributes (slots). The schema models the task in hand, e.g., a restaurant reservation involves information about the date, time, party size, and any special requests.
Ontology: Represents a set of entities, where each entity has attributes (slots), *and the relationship between entities*, for example *Reservation* "is made by" a *Customer*. Often ontologies are considered more semantic-forward than schemas, the reason being the added meaning they inherit through these entity relationships.
(2) **Knowledge Graph and Data Base**. Structured data representations that model real-world entities (instantiated entities), their attributes, and the relationship between them. Knowledge graphs can be class-specific, or integrated (multiple class objects in the same graph).

In addition to these graphical representations, some TOD systems are built to operate without a predefined input. Such methods often rely on training datasets from which they capture domain or task-specific knowledge. Not relying on input offers flexibility, however, it does not guarantee faithful generations or a method of verifying the information.

3.2 Utterance Generation

Various techniques have emerged for constructing dialogue from **provided input** in the form (semi)structured, predefined knowledge, which can be grouped into three main approaches: (1) predefined language templates, (2) wrapping the slots and values in natural language, and (3) generating natural language conditioned on the input. There are also methods that operate **without predefined input**. These methods adopt an approach centered on discovering entity-based slots and values through learning. These strategies typically rely on training datasets often curated by human annotators; alternatively, systems may be trained or fine-tuned on artificially generated dialogues. We identify three distinct approaches for achieving latent-space discovery of TOD-specific entity attributes, (1) variational inference, (2) few-shot learning, and (3) in-context, discussed in Sections 3.2.4–3.2.6. Methods that rely on provided input do not require training, while variational inference, few-shot learning, and in-context learning require training, fine-tuning, or contextual learning at inference time.

We note at the outset that a conversation inherently involves at least two participants, any TOD system requires a conversational partner (something akin to a user) to produce replies. The concept of **simulating** a user to interact with the TOD has been previously established [39, 53, 89, 93, 114]. In TOD data generation, the conversation partner can be a human or a simulator, which can be another pre-trained dialogue system, or a dialogue system that is co-trained alongside the TOD, a paradigm which is often referred to as a two-sided simulation. This method is frequently used in conjunction with reinforcement learning policies, allowing the two simulators to iteratively enhance their performance through mutual interaction. In this article, we focus on the data generation aspects, and refer the reader to the book by Balog and Zhai [10] for further details on user simulation.

3.2.1 Predefined Language Templates. Utilizing predefined language templates, referred to as an *agenda-based* or *schema-based* approach, ensures that the generated conversations follow a predetermined structure. The system then utilizes a series of responses that align with specific conversational goals or tasks, guiding the dialogue according to a pre-established **outline** or **schema**, which have pre-defined language templates for each intent and slot type. For example, the intent *<book_movie>* can be associated with the template *"Book movie with [name="value"] and [date="value"]*. For a specific movie and date, such as *inform<intent=book_movie, name=Inside Out, date=tomorrow>*, the template is filled and generates the turn *"Book movie with name Inside Out and date is tomorrow."* Often, agenda- or schema-based TOD systems contain a human-centered paraphrasing step, or an NLG module, to convert these generations into more diverse and intelligible human dialogue.

One article that follows an agenda-based approach is ABUS [72]. The input consists of a set of rules (a stack-like representation of user states), and example dialogues (used for building an accompanying user simulator). The simulator sends real-time simulated utterances to the system, which is trained to reply using an RL policy. This approach addresses the challenge of training dialogue systems to respond accurately and effectively in real-world scenarios, facilitated by a publicly available simulation framework. A drawback of agenda-based approaches is that they are task-dependent. M2M [117] has access to multiple *APIs* alongside a task specification, where each API has access to a task schema, outlining the scope and restrictions of interactions for the task in question. The generation process of M2M can be divided into two parts: first, using a **task**

specification that contains a schema corresponding to an API; second, using **task-independent information** that maps the task specification to a set of dialogues. Moreover, SGD [108] addresses the fact that in the real world, multiple services have overlapping functionality. The authors advocate for building a single unified dialogue model for all services and APIs by allowing access to a schema that provides a dynamic set of intents and slots, along with their natural language descriptors. Using dynamic schemas for each API allows for sharing knowledge between the services. SGD builds on top of M2M [117] by generalizing it to multiple user-system simulations, called agents, spanning over 26 services covering 16 domains, and resulting in a 16k dialogue dataset.

3.2.2 Natural Language Around Input. As a response to being limited by predefined language templates, Kim et al. [63] define three strategies. First, augmenting the **knowledge base** (**KB**) to enrich the semantic complexity of the dialogue system without altering its operational constraints. Second, modifying the schema itself by either introducing new constraints, removing existing ones, or altering the values these constraints can assume. Third, which is the focal point of this subsection, is directly generating dialogues in natural language form, extending the functionality of dialogue systems to previously unseen domains. This method circumvents the limitations imposed by rigid schema constraints and opens future research directions for more dynamic and versatile interactions.

The **Neural User Simulator** (**NUS**) [65] eliminates hand-crafted rules for natural language generation by utilizing a corpus-driven approach to generate natural language utterances. NUS leverages dynamic goal generation, meaning that the system can dynamically change the user goal with more dialogue context, assuming the user would want to shift their goal mid-conversation. It uses the ontology created with the **dialogue State Tracking Challenge 2** (**DSTC2**) [47] dataset's evolving states, from which the system can extract different slots and, together with the dialogue history, can generate a dynamic goal. NeuralWOZ [63] presents a novel system composed of two sequential neural models, namely the Collector and the Labeler. The Collector constructs dialogues given as input a user goal instruction and API call results. These elements synthesize into a belief state encapsulating both informable and requestable slot and slot values. The Collector generates the dialogue, and Labeler then annotates it with possible slots and slot values from the belief state extracted from a pre-defined list, creating a coherent dataset for dialogue state tracking.

3.2.3 Natural Language Conditioned on Input. The **Hierarchical User Simulator** (**HUS**) [45] is part of the same family as the previously discussed ABUS and NUS. HUS employs a multifaceted encoding scheme to transform different dialogue features in different vector representations: (1) the user goal, (2) the current dialogue turn, and (3) the entire dialogue history. Moreover, HUS has access to a KB from which it extracts information about the task and domain of the dialog. Due to the deterministic nature of HUS, the model will yield the same dialogue each time given identical user goals and system turns. To introduce variability, the authors propose a variational inference framework VHUS, by sampling an unobserved state from a latent space at each turn, with the latent space being positioned between the encoding of the dialogue history and the decoding of the user turn. Compared to VHDA, which is discussed in Section 3.2.4, the latent space in VHUS enriches the dialogue generation with diversity without modifying the underlying user state and goal generation. Therefore, while VHUS contributes to dialogue variety through latent sampling, it does not compromise the factuality of the generated slots and values.

Similarly, TUS [76] maps different inputs to different representations in the feature space, however it also distinguished itself by its unique domain-agnostic feature representations. A similar work is JOUST [126], which combines two simulator agents trained on a collection of source domain dialogues to converse with each other through natural language. The pre-trained agents are fine-tuned in a low resource setting on target domain data using RL. Similarly to TUS, it

is simulation-based, it has access to a knowledge base, it does not generate new values, and keeps track of the dialogue state simultaneously with the natural language dialogue turn generation. Another article that follows analogous work is Unified-US [128], which leverages a simulator model composed of two parts, namely the user and system agents. The authors first pre-train both agents using a multi-domain dialogue dataset without user goals. Subsequently, they are further pre-trained on the same dialogue dataset, however, this time incorporating the user goals, and fine-tuning on a target domain (NeuralWOZ [63]).

IND [4] is a simulation-based generated dataset on negotiation dialogues. The authors create the **Integrative Negotiation Dataset** (**IND**) using GPT-J with human-in-the-loop for quality check. The second part of this study uses the IND dataset to train the negotiating agent INA, which is further discussed in Section 3.2.6. The challenge of this data generation problem is that negotiation strategies are highly context-dependent, involving interactions about aspects such as price, product, delivery, quality, and so on, while also answering with knowledge-grounded information such as manufacturing details about the product, making it quite a challenge for generating high-quality dialogues. The authors construct the dialogues based on the "Integrative" approach to negotiation, i.e., a win-win situation where each party understands the other's needs and the goal is to reach mutual satisfaction.

3.2.4 Variational Inference. VHDA [148] is a method that receives as input human-generated dialogues, and constructs utterance-level natural language outputs together with dialogue acts, modeling latent variables over all dialogue aspects to extrapolate and generate novel slot values. This characteristic allows the system to produce attributes beyond those encountered in the training data, demonstrating an ability to innovate within the dialogue context and suggesting a higher degree of system adaptability and generative capacity. These features highly differentiate VHDA from the previous research. The authors build on top of the idea of VHCR [102], encapsulating multiple encoder and decoder modules, where each models a specific dialogue feature. VHDA can be closely compared to VHUS [45], however, the main and very important difference between the two is that VHUS samples from the latent space of natural language turn generation, while VHDA also models the user goal, and dialogue state in the latent space.

3.2.5 Few-shot Learning. In this section, we discuss articles that leverage the capabilities of LLMs to understand the underlying syntax and semantics of the dialogue data. Here, we differentiate between learning with few-shot examples, and in-context learning. Both techniques require limited data.

— *Few-shot learning*: the ability of a model to generalize when provided a very small dataset for **training** or **fine-tuning**.
— *In-context learning*: the ability of a model to generalize when provided very few examples in the input prompt **without explicit training** or **fine-tuning**.

Simulated-Chat [91] leverages the abilities of large pre-trained models to follow instructions and generate synthetic conversations. The system is split into user and agent simulators. The input of the user simulator is a set of instructions based on which it can generate a dialogue utterance, while the input of the agent simulator is a knowledge base. Given the current dialogue context, the agent generates a belief state. The two simulators hold a conversation conditioned on both the instructions and knowledge base, trained by fine-tuning GPT-2 and Longformer, first on human-generated dialogues, and then on additionally self-generated simulated dialogues. To use this framework for TOD generation, a belief state generator is added after the agent system. This generator produces key-value pairs that are matched against the knowledge base to retrieve relevant information, with the belief state updated using greedy methods.

3.2.6 In-context Learning. ICL-US [125] presents a method to generate TOD dialogues through in-context learning, receiving as input a few example dialogues to perform a new task, without needing any fine-tuning. The framework is composed of a *user simulator* and a *dialogue system*. Given a user goal and a few in-context dialogue examples, ICL-US generates a user utterance per turn. In-context learning requires the dialogue examples to be similar to the desired generation. The authors apply two sampling techniques to select the few-shot dialogues for each generation: (1) random sampling, and (2) sampling based on Jaccard similarity between the user goal and the candidate dialogue goals. Similarly, Dialogic [75] uses as input a few in-context dialogue examples, and a simulator to build TOD data using in-context learning. The authors demonstrate that on MultiWOZ2.3, their approach leads to better performance than when training on human-generated dialogue. Dialogic applies a Controllable Dialogue Generation step: This is an auxiliary step to verify the reliability of the GPT-3 generated dialogue. Once prompted, GPT-3 will generate a belief state and user utterance. If the belief state contains something that the dialogue turn generated utterance does not, it is called over-generation. If the belief state lacks something in the utterance, it is called de-generation. INA [4] leverages GPT-2 for in-context generation of dialogues, using as examples the IND dataset generated in the same article. This is achieved by fine-tuning GPT-2 in a supervised fashion with a cross-entropy loss and with a novel RL reward function using the PPO loss. More details about both IND and INA are presented in Section 3.2.3.

Intent generation. There is a line of work that specializes in data generation for intents in a task-oriented setting. For example, Fang et al. [34] study the challenge of identifying unseen intents that are compositions of previously observed ones, and uses ChatGPT as a data augmentation method to enhance generalizability. Along similar lines, Dai et al. [20] and Yu et al. [149] explore augmentation strategies for rephrasing around intents and expanding domain coverage. Building on this idea, Lin et al. [77] proposes a two-stage pipeline for zero-shot intent detection: utterances are first generated with an open-source LLM, and then refined by a smaller sequence-to-sequence model fine-tuned on seen domains. Their "generate-then-refine" approach improves both the diversity and utility of synthetic data, outperforming zero-shot generation alone. Meanwhile, Sauer et al. [113] tackles the problem of limited labeled data by combining few-shot learning with knowledge distillation for intent classification, showing that compact models can still achieve competitive performance in data-scarce scenarios.

3.3 Quality Filtering

In this section, we explore quality filtering with an emphasis on checking the accuracy of the generated data. Similar to ODD and CIS dialog types, quality filtering focuses on lexical and consistency checks. However, TOD has an extra challenge in filtering dialogs for the quality of the generated intents, entities, and entity attributes. Methods employing a schema, ontology, or knowledge graph in the generation process inherently ensure the accuracy of entities and entity attributes, as they are directly extracted from the input representations. This principle holds for the majority of articles discussed in this study, including ABUS, M2M, SGD, NUS, NeuralWOZ, HUS, VHUS, JOUST, and Unified-US [45, 63, 65, 72, 108, 117, 126, 128]. We consider these to have factuality granted. On the other hand, methods that discover slots and values in the latent space do not have factuality covered. Dialogic [75] controls the generation through an extra step called automatic revision, correcting for potential annotation errors by comparing the GPT-3 generated belief state with the current utterance, and avoiding any de-generation or over-generation issues. ICL-US [125] adds an evaluation step by comparing the dialogue acts extracted from the generated System and User NLU components at each turn. IND and INA [4] apply a human-in-the-loop component where experts make edits at the utterance level of the generated dialogue to ensure

the information is grounded in the provided background database, while also correcting minor grammatical errors. Meanwhile, VHDA [148] ensures the semantic logic of the dialogue turns but does not apply quality checks for filtering generations.

4 Open Domain Conversational Data Generation

ODD represent a specific type of conversational data that contains conversations across a wide variety of topics without being confined to specific tasks or domains [94]. When developing a synthetic dataset for ODD, it is essential to incorporate key features of ODD interactions [32]. Firstly, conversational coherence must be ensured, meaning each turn in the conversation should meaningfully connect to previous ones [94]. Secondly, the dataset should promote response diversity, avoiding bland and repetitive replies like "I don't know" to foster more engaging interactions [157]. Additionally, the dataset should encompass a broad spectrum of topics, enhancing the system's generality. Lastly, it is crucial to generate conversations that aim at eliciting informative responses, ensuring that conversations are both knowledgeable and relevant. For example, in knowledge-grounded dialogue systems, responses should draw on external knowledge bases, while in personalized dialogue systems, they should align with user profiles [33, 43, 51, 52].

This section reviews existing work on generating conversational data for ODD systems and systematically outlines the generation process in three components: input generation, utterance generation, and quality filtering. The first component, *input generation*, is tasked with providing the fundamental information necessary to initiate conversations. It processes a set of unstructured data and produces structured information cards, each serving as the basis for a conversation. In the literature on data generation, these information cards are referred to as "conversation seeds." For instance, when creating a conversational dataset from a **knowledge graph** (**KG**) [61], the input generation component ingests a large KG, samples a triple, and through several steps, transforms it into a concise description that forms the foundation of a sample conversation. The second component, *utterance generation*, utilizes the conversation seed to generate a multi-turn conversation. For example, using the description obtained from a KG triplet, this component instructs an LLM to generate a multi-turn conversation. The third component, *quality filtering*, aims at eliminating samples that do not meet the required quality standards and could potentially degrade the performance of the conversational system trained on this dataset. For instance, if conversation seeds or the resulting dialogue samples fail to meet quality criteria, they are discarded using heuristic or automated tools.

This section provides a detailed explanation of each component and discusses the methods associated with them. Figure 8 summarizes the methods employed in the literature for the aforementioned components, and Figure 7 illustrates the complete generation process with an example.

4.1 Input Generation

Everyday dialogues usually revolve around specific topics, starting with a general question that gradually moves to more specific aspects of the topic [61]. Therefore, to create a conversation sample, it is necessary to first define an information card, referred to as "conversation seed," which specifies the main topic, subtopics, and key topic details for a conversation (see example in the electronic appendix). Thus, the input generation component processes the data source with the goal of producing conversation seeds. The differences among input generation methods stem from the types of input sources used, the techniques employed in the generation process, and the specific content of each conversation seed. The subsequent sections will discuss various methods for generating conversation seeds.

4.1.1 Existing Dialogues. In this method, information from existing crowdsourced conversation datasets is used to generate conversation seeds. These datasets typically include background

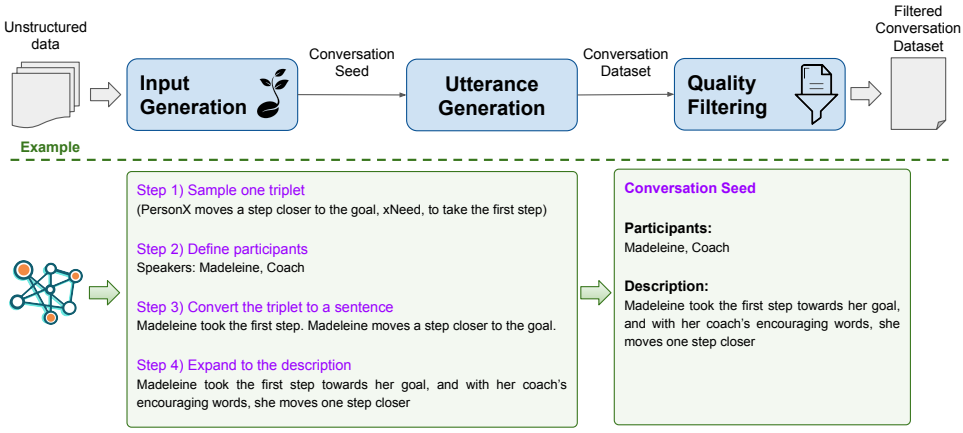

Fig. 7. Overview of data generation in ODD. In this example, adapted from SODA [61], knowledge graph triplets are used to construct a short description that serves as a conversation seed. An LLM is then prompted to generate a conversation based on this description.

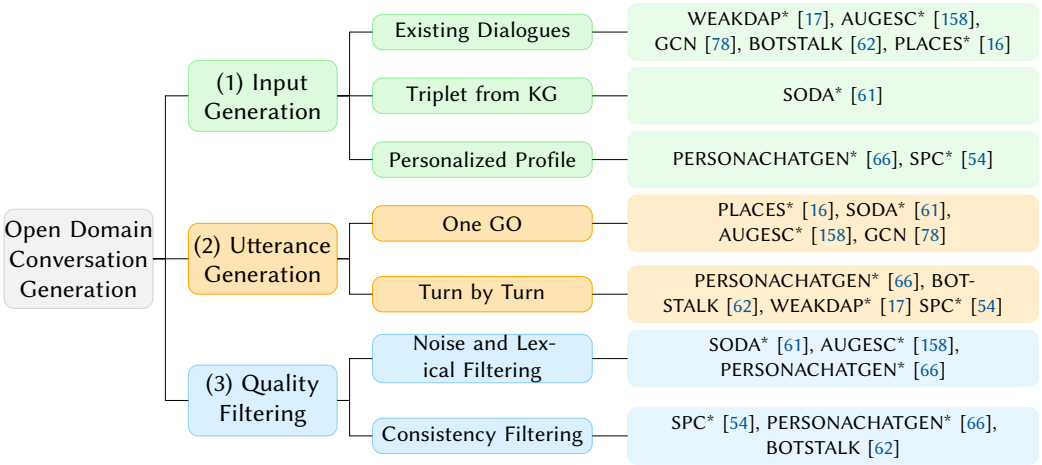

Fig. 8. Overview of approaches used for Open-domain conversation generation. The conversation generation process is comprised of three main components, described in Subsections 4.1–4.3. Methods using LLM are specified with ∗.

information for each conversation, which serves as an effective starting point for creating new synthetic conversations. The process involves selecting a crowdsourced dataset that is well-suited to the specific end task. As a result, conversation seeds in this approach primarily consist of task descriptions, background knowledge, and the initial one or two turns of the conversation.

In this realm, WEAKDAP [17] focuses on classifying emotions and conversational actions and detecting intentions in Spanish, and it utilizes the DailyDialog [73] and FBTOD [115] datasets for building it's conversation seeds. AUGESC [158] aims at generating conversations for **Emotional Support Conversation** (**ESC**) task. It employs the **EmpatheticDialogues** (**ED**) [107] dataset, which offers a rich variety of dialogue descriptions tagged with emotions and detailed descriptions of emotional states. GCN [78] is dedicated to creating knowledge-based conversations and uses

the TopicalChat dataset [42], where each conversation is linked to relevant facts or articles found on Wikipedia, Reddit, or in The Washington Post. BOTSTALK [62], designed for multi-skill conversational systems, integrates multiple behaviors and skills into a single conversation to enable appropriate interactions with diverse users and situations. BOTSTALK selects ConvAI2 [30], **Wizard of Wikipedia** (**WoW**) [31], and ED [107] datasets and combines them, using the conversation description and the first turn of the conversation sample as seeds.

Unlike previous approaches that use utterances from existing conversational datasets, PLACES [16] creates background information and conversations using human workers. Initially, it compiles a list of topics and tasks from the **Feedback for Interactive Talk and Search** (**FITS**) Dataset [141], which includes 52 conversational topics (e.g., "nutrition," "philosophy") and 315 subtopics (e.g., "Italian food," "Soren Kierkegaard"). For each subtopic, PLACES manually generates background information. Additionally, it constructs a series of ten dialogues between two speakers for selected topics, illustrating everyday conversations with proper grammar. Consequently, the conversation seed in PLACES comprises the topic, background information, and a selection of sample conversations, all produced by human effort.

4.1.2 Triplet from KG. In addition to existing conversational datasets, other types of resources, such as KGs, can be used to construct conversation seeds. Each triplet in a KG comprises two entities, referred to as the head and the tail, connected by a relation. The motivation for using textual resources beyond conversational datasets stems from the scarcity of conversational material in some domains. This makes it beneficial to leverage other textual resources, like KGs, for generating conversations. This approach effectively enhances the use of available textual resources in creating conversation datasets. SODA [61] utilizes KG triplets [137] to generate a concise description. The process involves several steps. Initially, SODA samples a socially relevant triplet, since the goal is to generate social conversations. It then transforms the triplet into a simple sentence using predefined templates tailored to each type of relation. To enhance the naturalness of the sentences, SODA substitutes variables representing individuals with the most common names from U.S. Social Security Number applicants.[1] Lastly, using GPT-3.5 [97], SODA expands this sentence into a short description of two or three sentences. The description then serves as the conversation seed.

4.1.3 Personalized Profile. Personalized chat systems are a distinct type of chat system that tailors responses based on a **user profile** (**UP**). Each UP consists of multiple **profile sentences** (**PSs**) that contain personalized information about the user. These PSs enable a dialogue system to generate personalized responses. To generate conversational data for such systems, the fundamental approach involves developing UPs for two agents and then generating conversations based on these profiles. As a result, the conversation seed primarily consists of UPs. The process of generating these seeds begins with a list of topics and unfolds in two main steps. The first step is the preparation or creation of a set of PSs. The second step involves grouping several PSs together to form a complete UP. Ultimately, these user profiles comprise the conversation seed and serve as the foundation for generating personalized conversations.

PERSONACHATGEN [66] generates conversational data for personalized systems by first defining a hierarchical taxonomy of persona topics and using GPT-3 to generate PSs. These PSs are grouped into UPs using contradiction scores, followed by quality control via heuristic filtering and automatic compatibility scoring. **Synthetic-Persona-Chat** (**SPC**) [54] starts from PSs in the PersonaChat dataset [153], expands them using an LLM, and organizes the resulting PSs into coherent UPs with an NLI classifier.

[1] catalog.data.gov/dataset/baby-names-fromsocial-security-card-applications-national-data

4.2 Utterance Generation

The primary goal of this step is to transform a conversation seed into a complete conversation. Thus, the input for this component consists of conversation seeds generated by the previous component, and its output is synthetically generated conversations. For ODD systems, the primary method for generating a conversation involves prompting an LLM with a conversation seed to generate a multi-turn conversation at once. This approach is referred to as *One Go* in this article. Some studies, however, employ a *Turn by Turn* generation approach, where the conversation is constructed sequentially, one turn at a time. We will discuss the reasons why the turn-by-turn method is used.

4.2.1 One GO. This method generates a multi-turn conversation from a conversation seed, primarily by prompting an LLM. There distinct methods are used to generate conversation in one go. The first method uses **In-context Learning** (**ICL**), where a pre-trained LLM is directly prompted without any fine-tuning [16, 61]. PLACES [16] generates conversations by using a prompt that initially includes a topic along with its human-written background information and a few randomly selected conversation samples as few-shot examples. Subsequently, another topic along with its corresponding background information is added to the prompt, and the LLM is tasked with generating a conversation for it. SODA [61] employs GPT-3.5 to create multi-turn conversations from descriptions generated based on a triplet from a KG.

In the second method, an LLM is first fine-tuned and then used to generate conversations. For instance, AUGESC [158] initially fine-tuned GPT-J [129] on a dialogue completion task using 100 samples from the ESConv dataset [81] in one epoch. After fine-tuning, a conversation description along with the first utterance is provided to the LLM, which is then tasked with generating the remainder of the conversation.

The third method employs a Generator-Critic Architecture [84] and uses a combination of prompting and fine-tuning iteratively for generation. In this method, an LLM first generates multiple conversation candidates. Then, a Critic, which is another LLM, evaluates these candidates based on predetermined policies and selects the best ones. These top conversations are then incorporated into the dataset for the next generation cycle. This iterative process of selecting and adding the best candidates gradually enhances the Generator's performance. Similarly, SPC [54] utilizes this approach for personalized conversation generation, prompting the LLM with a pair of user profiles.

4.2.2 Turn By Turn. Some studies generate conversations incrementally, often involving the interaction of two or more LLMs. There are several reasons for using the turn-by-turn method. The first reason is to generate a personalized conversation based on two user profiles. In this scenario, two LLMs, each equipped with a user profile, engage in a dialogue with each other, generating the conversation turn by turn. PERSONACHATGEN [66] utilizes this approach by employing two GPT-3 models prompted by user profiles. The second reason involves merging multiple datasets. In this approach, the choice of which dataset to use is determined turn by turn, based on the dialogue history. BOTSTALK [62] uses this method as it merges three datasets to create a multi-skill conversational dataset. The third reason is to increase the quantity and diversity of the data. By generating dialogues turn by turn, various scenarios can be created for generation. For example, WEAKDAP [17] defines three strategies for generation: First, **Conversation Trajectory Augmentation** (**CTA**) starts with the initial turn and substitutes the subsequent turns with generated ones. Second, **All Turn Augmentation** (**ATA**) generates a turn, considers the current subset as a conversation sample, then retains the original turn and generates the next. This results in $n-1$ new conversations of varying lengths, from 2 to n, for a conversation with n turns. Third, **Last Turn Augmentation** (**LTA**), a specific instance of ATA, focuses on replacing only the last turn of a conversation with a generated utterance.

4.3 Quality Filtering

The primary goal of the filtering component is to eliminate conversations that lack key ODD features such as correctness, consistency, diversity, and informativeness. The existing filtering methods can be broadly categorized into two types. The first type, Noise and Lexical Filtering, primarily focuses on assessing correctness and diversity. In this type, the filter removes samples that exhibit various issues, such as unfinished conversations [158], deviations from desired patterns [61, 66], repetitive patterns [66], inappropriate turn lengths (either too short or too long) [158], conversations with more than two participants [61], dangerous content [66], toxic content with social biases, and offensive material [61, 66]. In the second type of filtering, the focus is on checking the consistency of conversations. BOTSTALK [62] ensures dialogue consistency by examining each newly added turn to verify that it aligns with the rest of the conversation. Similarly, in personalized-based models [54, 66], the consistency between persona sentences in one user profile is also assessed. To achieve this, a **Natural Language Inference** (**NLI**) classifier is primarily utilized.

5 Information Seeking Conversational Data Generation

CIS conversations are multi-turn dialogues between one or more users and an information system, conducted primarily in natural language, whose goal is to resolve evolving information needs [151]. However, CIS systems face several challenges in accurately addressing a user's request. The first challenge to address is ensuring the system maintains control of the conversation, allowing it to progress coherently until it meets the user's needs, while also offering users the flexibility to specify what they want to find or how they want the information to be presented [151]. The challenge intensifies when the user's queries are about various subjects or responses need to be drawn from different resources during the conversation. In such scenarios, the conversation may experience topic shifts, necessitating the system's ability to recognize when the topic changes and accordingly switch the response source [3]. Additionally, users may pose complex questions that require multi-hop reasoning across multiple sources to formulate a response. This demands more sophisticated retrieval models and reasoning capabilities, as the system needs to synthesize information from diverse resources [74, 152]. Another major challenge is query ambiguity, where the system may not understand the user's request or might have multiple potential responses but cannot determine which one is most relevant. In such cases, the system need to ask clarifying questions. This interaction, where both the user and the system can initiate queries and drive the conversation, is known as a mixed-initiative system [26, 139]. To effectively develop a CIS system capable of handling these challenges, it is essential to generate training data that encapsulates these scenarios.

This section reviews existing work on generating conversational data for CIS systems and systematically describes the generation process through three main components: input generation, utterance generation, and quality filtering; see Figure 9. The first component, *input generation*, is responsible for providing the fundamental information needed to initiate conversations; e.g., topic and evidence document. It also defines a conversation flow, which is used to guide the conversation to address users' information needs. Both elements, fundamental information and conversation flow, constitute what is referred to as a "Conversation Seed." For example, a conversation seed may include an entity name with relevant background information, followed by a sequence of intents like [original question, clarifying question, further details]. The second component, *utterance generation*, aims at generating a conversation based on the "Conversation Seed." The third component, *quality filtering*, seeks to eliminate samples that do not meet the required quality standards, which could potentially degrade the performance of the conversational system trained on this dataset. This section provides a detailed explanation of each component and discusses the methods associated with them. The methods employed for these components are illustrated in Figure 10.

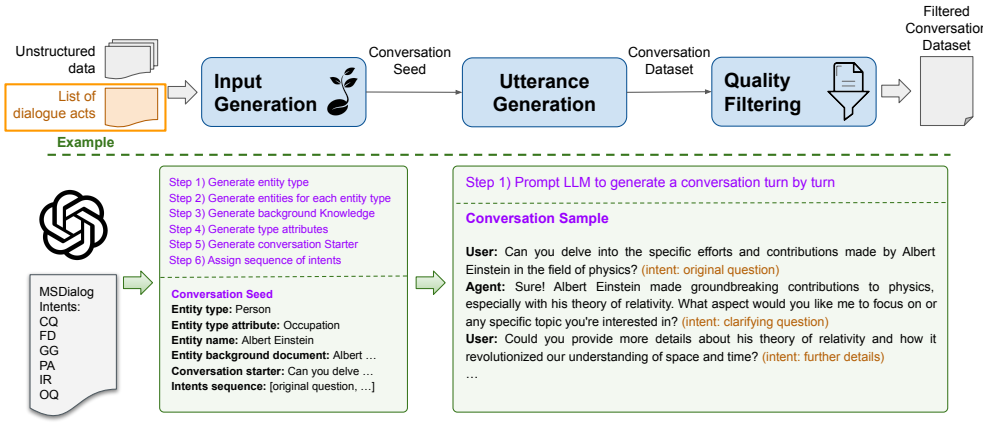

Fig. 9. Overview of data generation in CIS. In this example from SOLID [8], a list of intents from the MSDialog dataset [105] is fed into the input generation component. This component sequentially prompts an LLM to generate the necessary background information and selects a random sequence of intents. After constructing the conversation seed with the background information and intent sequence, it prompts an LLM to generate intent-aware conversation turns sequentially.

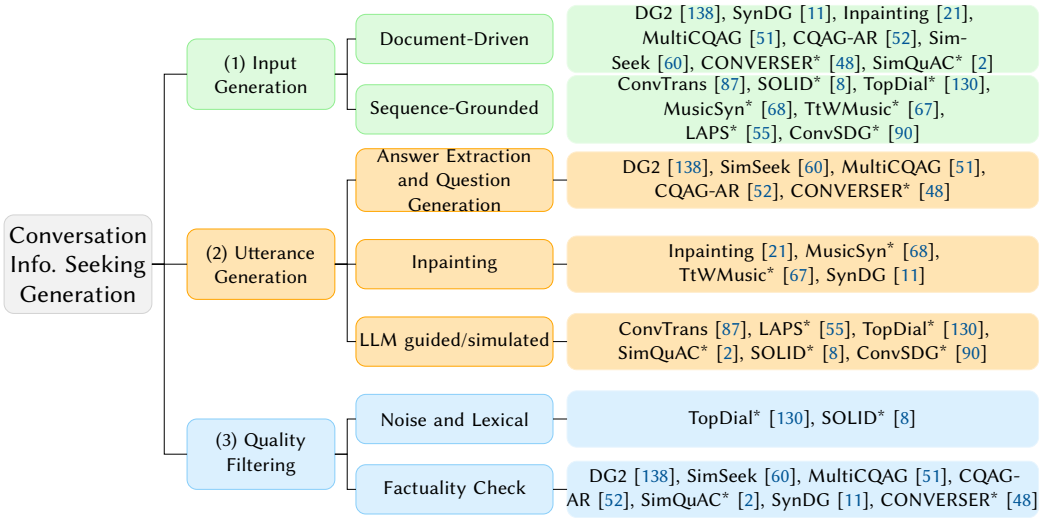

Fig. 10. Overview of methods for information seeking conversational data generation. The conversation generation process is comprised of three main components, described in Subsections 5.1–5.3. Methods using LLM are specificed with ∗.

5.1 Input Generation

CIS conversations are designed to respond to queries or gather information about a topic, typically following a structured sequence. In the literature on conversation generation, it is generally assumed that an information-seeking conversation starts with the user's query, progresses through a series of interactions between the agent and the user, and concludes with the delivery of the final answer. Therefore, each turn in these conversations can be labeled with an act name, resulting in a multi-turn conversation being categorized by a sequence of dialogue acts, commonly referred to as "dialogue

flow." In other words, the dialogue flow offers a comprehensive perspective of the conversation, outlining the role of each turn and how it contributes to reaching the final answer. Thus, a dialogue flow, coupled with information about the topic of conversation and corresponding background information, forms the "Conversation Seed" for CIS systems (see example in the electronic appendix). The conversation sample is generated based on the title and document and is structured around the dialogue flow to meet the user's information needs. In the following, we discuss existing approaches for input generation and categorize them into two main categories of document-driven and sequence-grounded.

5.1.1 Document-driven. High-quality documents, such as those found on platforms like Wikipedia or PubMed, are typically written or edited by experts, who invest significant effort in refining their content and addressing potential reader questions preemptively [21]. These documents often begin with broad overviews of general concepts and gradually delve into more detailed aspects of the subject [33, 38, 43, 138]. The sequential organization of documents coupled with the rich background information of documents serve as a valuable resource for modeling dialogue flow of information-seeking conversations. Therefore, in the document-driven approach, the implicit flow of a document or a ranked list of document passages is used to guide the dialogue. In what follows, we discuss three methods of constructing a dialogue flow in a document-driven manner.

The first document-driven input generation method segments documents into passages and generates dialogue turns by selecting passages sequentially. DG2 [138] follows a turn-by-turn retrieval-based approach, using a passage selector to rank passages based on the conversation history and choose the most relevant one for each turn. SynDG [11] utilizes "knowledge pieces" instead of mere passages. A knowledge piece can be any textual segment; e.g., a sentence from a Wikipedia article or a persona sentence used in a personalized dialogues to describe a user's preferences. Segmenting a document can obscure its positional context, even though dialogues naturally shift toward later, more detailed parts of a document. To address this, DG2 incorporates both dialogue-turn and passage-position information. It marks speaker turns with prompts like "usernum:" or "agentnum:" to track dialogue progression, and assigns each passage an index indicating its location in the document. Combining these signals helps the model begin conversations near the document's start and gradually move toward later sections as the dialogue unfolds. SynDG initially defines a sequence of knowledge pieces as a pre-defined dialogue flow. It applies this pre-defined dialogue flow in two instances: the first instance uses the persona sentences from the PersonaChat dataset [153] as knowledge pieces, and the second instance utilizes passages from the WoW dataset [31].

In the second method of docuemnt-driven input generation, documents are conceptualized as dialogues between the writer and an imaginary reader, where each sentence in the document is viewed as the writer's response to a hypothetical question from the reader. Thus, in this approach, the dialogue flow consists of the document's sentences. This concept, called Inpainting, was first introduced by Dai et al. [21], and subsequently adopted in other studies [67, 68].

As the third method, a specific flow for the conversation is not defined. Instead, a complete document of or background information is provided, and it is left to the "utterance generation" component to decide which part of the document to use for dialogue generation [2, 51, 52, 60].

5.1.2 Sequence-Grounded. In this approach, the conversation seed comprises a topic, its corresponding background information, and a sequence of dialogue acts [83, 130], sometimes referred to as intents [8, 105]. The conversation seed closely resembles that used in ODD but additionally includes the sequence of acts. For this setup, two types of input resources are necessary: a collection of topics with their background knowledge and a list of dialogue acts. Furthermore, given that a sequence of dialogue acts is selected from a pool, verifying the validity of this sequence is crucial.

TopDial [130] generates synthetic recommendation dialogues using a role-playing framework where two LLMs act as the system and the user. Each role is initialized with a conversation seed: the system seed includes background information, an initial utterance, dialogue acts, and a target from DuRecDial 2.0 [83], while the user seed consists of profile attributes and personality features. User personalities are simulated using randomly assigned positive or negative descriptions of the Big-5 traits, openness, conscientiousness, extraversion, agreeableness, and neuroticism [41]. Although only the final target is predefined, the system typically begins by asking preference-related questions and gradually introduces topic-specific attributes as the dialogue unfolds. SOLID [8] generates intent-aware information-seeking dialogues by sequentially prompting an LLM to produce entity types and entities, background documents, and an initial utterance, which together form the conversation seed. The dialogue flow follows intent sequences from MSdialog [105], preserving the original intent distribution. MusicSyn [68] and TtWMusic [67] generate music recommendation dialogues using fixed dialogue act sequences and the CPCD dataset [15]. They construct dialogue flows by embedding artist playlists and sampling sequences based on cosine similarity to maintain topical consistency. ConvTrans [87] reorganizes raw web search sessions into a heterogeneous session graph, where nodes represent queries and edges represent relationships between queries. The graph includes three types of edges (response-induced, topic-shared, and topic-changed) capturing three common types of query relations in conversational search. ConvSDG [90], on the other hand, generates conversations by taking a topic, its corresponding description, and a partial conversation as input.

Some studies do not specify a fixed sequence; instead, outline a set of actions and select one before generating each turn. These studies focus on collaboration between LLM and humans, employing semi-synthetic methods for conversation generation. LAPS [55] uses a set of primary dialogue acts for personalized recommendation tasks, (1) greeting, (2) preference elicitation, (3) recommendation, (4) follow-up questions, and (5) goodbye. It then guides the conversation follow by generating a guidance for each turn, specifying the next action that should be taken in the conversation based on the conversation history and user preferences. This guidance is then utilized by a crowd worker to compose the next utterance for the conversation.

5.2 Utterance Generation

The primary goal of this component is to transform a conversation seed into a multi-turn conversation. However, the content of the seed in CIS differs from that in ODD. The input for this component is the conversation seeds containing a topic and dialogue flow, and its output is a conversational dataset. In the following, we will explore the approaches used for utterance generation.

5.2.1 Answer Extraction and Question Generation. This approach utilizes two sequential components, known as Answer Extraction and Question Generation, which are based on the pipeline method for QA generation introduced by Alberti et al. [5]. Typically, the first component extracts a text span that could potentially serve as an answer, and the second component generates a question corresponding to the extracted answer. DG2 [138] retrieves a passage to use as a conversation seed for generating a turn. It then employs a span extraction component to identify a rationale that could potentially serve as an answer to a question. Following this, it sequentially generates user and agent utterances based on the extracted rationale and the conversation history. MultiCQAG [51] and CQAG-AR [52] employ a similar approach, with the distinction that after extracting the span, they determine which type of question to generate: open-ended, closed-ended, or unanswerable. SimSeek [60] explores two scenarios of generating synthetic conversations from documents. In the first scenario, called "SimSeek-sym," an evidence passage is provided from which an answer

is extracted first, followed by the generation of a question based on the extracted answer. In the second scenario, "SimSeek-asym," only a topic and background information are supplied to the question generator component. Then, an answer finder component attempts to identify the answer within the evidence passage. In this approach, the full Wikipedia passage is considered as the conversation seed. CONVERSER [48] generates a sequence of queries within a session, grounded on one or more passages. It also incorporates passage switching, where the current passage is replaced with a related passage at each turn with a certain probability.

5.2.2 Dialogue Inpainting. The core concept of dialogue inpainting is to reconstruct missing parts of a dialogue [21]; e.g., given a conversation containing agent's responses, the inpainting method generates the corresponding user's questions. Dai et al. [21] train a dialogue inpainting model by fine tuning an LLM on dialogues with randomly masked utterances. The inpainter then learns to predict the missing utterances of a dialogue. Assuming that sentences of a document represent agent responses in an imaginary conversation, the trained inpaiter model then generates user questions for all system responses and transforms a document to a an information seeking conversation. The concept of inpainting can be also seen in SynDG [11], where a fine-tuned sequence-to-sequence converts knowledge pieces (e.g., Wikipedia passages or sentences from a user's profile) into consistent turns of a conversation. Additionally, MusicSyn [68] and TtWMusic [67] apply this technique to generate conversations for a music recommendation system. The system's responses consist of a list of playlists, and the inpainting model is tasked with generating the user's questions to complement these responses.

5.2.3 LLM Guided/Simulated. The third category of CIS utterance generation methods employs LLMs to steer the conversation generation process either by guiding humans or by directly simulating user roles to generate utterances.

In the first approach, the LLM generates guidance, and humans create conversation utterances based on this guidance. LAPS [55] utilizes this approach to generate large scale human-written conversational data for preference elicitation and personalized recommendation. It generates multi-session conversations using an LLM to dynamically provide personal guidance for crowd workers who play both roles of user and agent. The personal guidance is generated dynamically per turn based on the conversation history and elicited preferences of the crowd worker in previous conversation sessions. It enables crowd workers to overcome the high cognitive load of generating conversational utterances in complex and lengthy conversations and craft their own utterances through dialogue self-play. LAPS consists of four key components: dialogue act classification, guidance generation, utterance composition, and preference extraction. For dialogue act classification, LAPS utilizes an LLM to identify the next act that should be taken in the dialogue. The primary dialogue acts in this framework are: (1) greeting, (2) preference elicitation, (3) recommendation, (4) follow-up questions, and (5) goodbye. In the guidance generation component, it uses GPT-3.5-turbo to create personalized guidance, with the template for guidance prompts incorporating dialogue history, preference memory, and the identified dialogue act. The utterance composition is handled by a human agent who crafts the conversation utterances for both the user and the assistant, using dialogue self-play. After the completion of a dialogue session, the preference extraction process extracts user preferences from the dialogue and verify them by the same human agent who managed the conversation.

In the second approach, multiple LLMs serve as both participants of the conversation, with no human involvement. TopDial [130] generates conversations for target-guided recommendation by simulating user, system, and supervisor roles using three instances of ChatGPT. The user is seeded with a profile and personality, while the system is conditioned on domain knowledge, dialogue acts, and the target goal. SimQuAC [2] developed to emulate the crowdsourcing methodology

originally used for creating the QuAC dataset [19], replacing human participants with LLMs. It employs zero-shot learning LLMs to simulate teacher-student interactions. The dialogue revolves around a wikipedia topic; the student only has access to the topic and background information and asks questions pertinent to it. The teacher, with full access to the article text, responds by selecting relevant text spans from the article to answer the student's questions, focusing on providing information directly from the source rather than generating responses from memory.

In the final approach, a single LLM is responsible for generating all utterances in a conversation. For example, SOLID [8] uses an LLM to generate intent-aware information-seeking dialogues from conversation seeds containing topics and intent sequences, with each turn produced via intent-specific instructions conditioned on dialogue history. While the original method generates dialogues turn by turn, its extension SOLID-RL adopts a one-go generation approach, improving naturalness, consistency, and generation speed. SOLID-RL further introduces an RL-guided zero-shot framework to generate entire conversations in a single step. ConvTrans [87] introduces a conversational query rewriter that transforms ad-hoc queries in the graph into natural language conversational queries, thereby bridging the gap between different query forms. It then employs a specialized random walk sampling algorithm to generate pseudo search sessions from the query-transformed session graph.

5.3 Quality Filtering

The primary goal of the filtering component is to remove conversations that lack key CIS features, particularly correctness and consistency, whether at the turn level or throughout the entire conversation. This step improves the quality of synthetic data and prevents the introduction of noise into downstream tasks. We categorize the filtering methods into two groups.

5.3.1 Noise and Lexical Filtering. The first type primarily focuses on ensuring correctness and maintaining diversity. This filtering is typically applied using heuristic methods to check correctness [8, 55, 130]. For instance, TopDial [130] implements fact-checking and makes corrections based on domain-specific knowledge. These filtering methods are commonly used for ODD and are particularly necessary for approaches that rely on LLMs for utterance generation. Since LLMs have a greater degree of freedom in generating utterances, they are more likely to produce inappropriate or irrelevant content. Therefore, Noise and Lexical filtering plays a crucial role in ensuring the quality and consistency of the generated conversations.

5.3.2 Factuality Check. In this type of filtering, the consistency of the generated utterances is evaluated. Document-driven methods predominantly include a factuality checking step to assess the performance of the utterance generator components. A notable technique is roundtrip consistency, initially introduced for QA generation [5] and later adapted for conversation data generation [48, 60, 138]. This method uses a model to verify whether the answer span remains consistent with the original span that prompted the question.

DG2 [138] incorporates passage selector and span extractor components in the utterance generation part. To implement filtering, DG2 develops additional passage selector and rationale extractor models. It then compares the turns generated by the two models and eliminates inconsistent samples. SimSeek [60] also employs additional answer extractor and question generator models but relaxes the filtering criteria from an exact match to word-level similarity (i.e., F1 score) to better accommodate the typically longer answers found in conversational QA as opposed to those in single-turn QA. CQAG-AR [52] focuses on the consistency between an answer and a question after the generation of each turn to avoid propagation errors in subsequent turns of a conversation. It introduces *answer revision* module, which immediately revises the extracted answer after a question is generated. Similarly, MultiCQAG [51] proposes a *Hierarchical Answerability Classification* module to address the error propagation issue by determining whether a question can be answered based

Table 2. Summary of Human-Created and Synthetic Datasets, Including their Tasks/Domains and the Number of Samples

Human-created Datasets			Synthetic Datasets		
Dataset	Task	Size	Dataset	Task	Size
Task-oriented Dialogue System					
DSTC2 [64]	State Tracking	2.5K	Sim-GEN [117]	Movie tickets	120K
DialEdit [86]	Image editing	129	NeuralWOZ [63]	Multi-domain	6K
M2M [117]	Restaurant and Movie	3K	SimulatedChats [91]	Multi-domain	10K
WoZ2.0 [135]	Restaurant Search	1.2K	IND [4]	Negotiation Strategies	4K
MultiWOZ [14]	Multi-domain	10K	DIALOGIC [75]	Multi-domain	8.4K
SGD [108]	Multi-domain	16K			
Open-Domain Dialogue System					
DailyDialog [73]	Emotion and Act Class.	13.1K	AUGESC [158]	Emotional Support	65K
Topical-Chat [42]	Chit-chat	11.3K	BSBT [62]	Blended Skill	300K
ESConv [158]	Emotional Support	1.3K	SODA [61]	Social Dialogue	1.5M
ConvAI2 [62]	Persona-based Chitchat	20K	PERSONACHATGEN [66]	Persona-based Chitchat	1.6K
Wizard of Wikipedia [31]	Knowledge-grounded	22K	Synthetic-Persona-Chat [54]	Persona-based Chitchat	10.3K
Empathetic Dialogues [107]	Empathetic	25K			
Prosocial Dialog [61]	Social Dialogue	58K			
PERSONACHAT [153]	Persona-based Chitchat	11K			
Conversational Information Seeking					
Doc2Dial [36]	Goal-oriented	4.8K	DG2 [138]	Goal-oriented	4.8K
OR-QuAC [19]	CQA	5.6K	WikiDialog [21]	CQA	11.4M
QReCC [7]	CQA	13.6K	WebDialog [21]	CQA	8.4M
CAsT 19-22 [22, 23, 99]	CQA	93	CQAG-AR [52]	CQA	4.7K
iKAT-23 [6]	CQA	25	WIKI-SIMSEEK [60]	CQA	213K
CoQA [52]	CQA	8K	CONVERSER [48]	CQA	31K
TopiOCQA [3]	CQA	4K	SimQuAC [2]	CQA	334
MSDialog [142]	CIS	2.2K	ConvTrans [87]	Conv. Search	75K
MANtIS [103]	Conv. Search	1.3K	SOLISpeak [8]	Conv. Search	316K
DuRecDial [83]	Conv. Recom.	6K	TOPDIAL [130]	Conv. Recom.	18K
CPCD [68]	Conv. Recom.	16K	TtWMusic [68]	Conv. Recom.	1M
MG-ShopDial [12]	E-commerce	64	ConvSDG [90]	Conv. Search	1.7K
LAPS [55]	PMCS	1.4K			

* Personalized Multi-Session Conversational Search.

on the given passage. If a question is deemed unanswerable, the module substitutes the answer with "unknown" to prevent erroneous continuations of conversation.

SimQuAC [2] improves LLM-generated conversations using *Question Validation* and *Answer Validation and Regeneration* modules, ensuring syntactic correctness, coherence, and topic-consistent responses. SynDG [11] implements a *two-stage filtering* process including flow-level and conversation-level checks. At the conversation level, another T5 model [106] is fine-tuned for a task that involves predicting masked utterances, similar to roundtrip consistency. Additionally, another model is fine-tuned to verify the consistency of the dialogue flow before generating the conversation turns.

Compared to the factuality-check filtering in ODD, both approaches share the ultimate goal of ensuring consistency between the turns of a conversation. However, in ODD, the focus is on verifying the consistency of the user persona or the coherence of selected turns originating from different datasets. In contrast, in CIS, the consistency is evaluated across conversation turns and question–answer pairs, as these interactions are grounded in a specific document.

6 Discussion

In this section, we discuss the proposed conversational data generation methods from different aspects.

Human and Syntactically Generated Data: We list the human-curated and LLM-generated datasets for TOD, ODD, and CIS in Table 2, along with their corresponding tasks/domains and sample sizes, to provide an overview of the existing datasets. Generally, human-curated datasets contain fewer samples, while the size of synthetic datasets depends on the authors' choices. Moreover, we observe that synthetic datasets have been created for a wide range of tasks and domains, whereas

the creation of synthetic data often relies on the existence of human-curated data in the same domain or task [8, 61, 130]. In general, synthetic data are useful in domains or tasks where there is a lack of conversational data or only a small amount of training data available. In such cases, other textual sources can be leveraged to generate a large amount of conversational data, thereby strengthening the training process. Another advantage of synthetic data is that it can introduce new challenges that are often absent in human-created data, such as topic shifts and lengthy conversations [61, 62].

Quantity and Quality: Existing studies mainly assess synthetic datasets based on two key dimensions: their quantity and quality. Quantity refers to features such as the number of samples, number of turns, average answer length, and average question length. The quantity of generated samples often depends on the author's choice and the task at hand [2, 21]. For example, SOLID [8] generated 316K samples, which is 158 times larger than its original seed dataset. In contrast, SimQuAC [2] generates a controlled number of samples, matching the size of the original dataset. Other generation features, such as the number of turns and tokens, are typically controlled to align with the original dataset, as these characteristics naturally reflect human-generated conversations [62, 138]. Quality of generated datasets is commonly assessed through metrics such as proactivity, coherence, personalization, target success rate, correctness, naturalness, and completeness [55, 60, 130]. For this evaluation, samples from both LLM-generated conversations and seed data are randomly selected and compared pairwise by either LLM evaluators or human annotators. For instance, TopDial [130] and SimQuAC [2] employed such evaluations and found that generated datasets achieved comparable or slightly higher win rates than the original datasets. Another approach to evaluating dataset quality is to measure performance on downstream tasks, such as passage retrieval or answer generation [138]. A general conclusion across studies is that using only synthetic samples alone cannot outperform scenarios where human-generated samples are utilized. However, effectively combining original data with synthetic data can improve system performance on downstream tasks. This is because the challenges and nuances present in human-generated data remain difficult for models to fully capture, and synthetic data alone cannot fully replicate these complexities [2, 8].

LLM-based vs non-LLM-based Methods: In this section, we discuss the transition of methods from non-LLM-based approaches to those leveraging LLMs. For TOD, early approaches primarily relied on rule-based systems and smaller sequence-to-sequence models. Methods like ABUS [72], M2M [117], and SGD [108] had access to predefined templates and schemas; neural approaches such as NUS [65], HUS [45], and VHDA [148] kept track of the dialogue states using neural architectures, with added modules for language generation. Introducing LLMs made a significant shift in how TOD generations are made, as this type of data relies a lot on exact information, such as restaurant opening hours and real geographical locations. The robustness of LLMs alongside their in-context learning and reasoning capabilities made it possible to generate both dialogue content and slot-value entries.

For ODD, one common approach in the absence of LLMs is to mix existing datasets. In this case, the main consideration is maintaining consistency between consecutive turns that originate from different datasets [62]. Another approach involves training a small model, specifically for the conversation data generation task [78]. In the LLM-based methods for ODD, the process typically involves providing the LLM with a topic and description, and instructing it to generate a conversation sample [16, 17, 153]. In some cases, in-context learning is applied, where the model is given examples and the initial part of an existing conversation, and is then asked to generate the remaining turns [158].

For CIS, most document-driven methods rely on LMs for utterance generation. This is because the generation process is controlled by a sequence of documents, and producing utterances grounded

in these documents can be effectively handled by smaller LMs [48, 60, 138]. However, in sequence-grounded methods, LLMs are predominantly used. This is because such methods require converting a keyword or intent into a coherent utterance, a task that demands the reasoning capabilities of LLMs.

Domain Adaptation: For TOD, several methods show promise for generalization across different domains, although it remains a critical challenge as some approaches require additional domain-specific schemas, and the generations have to be grounded. SGD [108] and M2M [117] demonstrate cross-domain transfer by creating dynamic schemas across services and domains, with SGD demonstrating that many domains actually have an overlap of entities and attributes, making it more easily adaptable. However, this is highly dependent on how domains are semantically close to each other. Nevertheless, these methods need either access to pre-defined domain-specific ontologies and schemas, or having sufficient overlap with a new domain to make use of dynamic modeling of entities and their features. Given LLM models, generalizability seems more straightforward with in-context learning; however, there is still a need for domain-specific examples and a robust way of evaluating generations.

7 Biases and Ethical Considerations

Synthetic dialog generation can potentially introduce biases, such as underrepresentation of real-world topics, user needs, and intents [8, 61]. While many of the approaches reviewed in this work leverage real-world datasets for training a model to generate synthetic dialogs, overly specialized corpora may lead to issues of poor generalizations and generations that are not applicable to broader or unseen scenarios.

LLMs introduce additional biases and may raise ethical concerns by generating dialogs highly influenced by their parametric knowledge acquired during pretraining on often a wide range of unknown data. A primary concern is that LLMs tend to avoid generating content involving long-tail entities, focusing more on information about which it is certain and has been seen more during pretraining [156]. A second challenge lies in generating hallucinated facts. While the generated dialogs can somehow be filtered and their factuality can be checked by comparison with ground truth statements, that often defeats the purpose of LLM-based generations and constrains creativity [49]. A last vital issue is that dialogs may be generated on undesirable subjects, given that LLMs inherently have socio-economic, cultural, racial and gender biases. Ideally, LLM generations would help to enrich the data to a calibrated distribution that can further help to eliminate these biases [146]. However, that is hard to control and evaluate in mass generation.

8 Conclusion

This survey article reviews methods for generating conversational data, focusing specifically on approaches that produce complete multi-turn dialogues. Following prior taxonomies [27, 95, 147, 151], we organize the discussion around ToD, ODD, and CIS systems.

Within the domain of TOD, the advancement and effectiveness of conversational systems, for example in performing specific tasks like booking a flight or making a reservation, heavily rely on the generation of robust and factually accurate dialogues. To address these constraints, research has focused on developing sophisticated data generation and augmentation methodologies that leverage structures such as entities, entity attributes, dialog acts, and intents within complex graphical representations, such as Ontologies, Schemas, and Knowledge Graphs, ensuring the generation of diverse, plausible, and relevant dialogues.

In ODD, general-purpose chatbots such as ChatGPT and GPT-4 have exhibited high quality and performance primarily in general tasks and concepts, rather than in specialized domains [24, 85]. Moreover, access to these models is usually restricted via APIs. To overcome these limitations and

improve the performance of LLMs in specific domains and languages, such as emotional support [158], Spanish intent detection [16], and multi-skill conversations [62], data augmentation followed by fine-tuning has been suggested as an effective approach [98, 121]. Additionally, some crowdsourced datasets may contain outdated concepts and lack newer ones. Synthetic data generation has been proposed to update the knowledge stored in LLMs and mitigate these issues [66].

CIS systems represent a promising type of dialogue systems that encompasses a wide range of tasks, challenges, and applications. The significant number of studies proposing methods for generating conversational data suggests that CIS systems have attracted considerable interest. Various approaches to creating dialogue data have been explored, including document-driven and sequence-grounded methodologies for tasks such as intent prediction, negotiation systems, and recommendation systems. Overall, researchers aim at managing the conversation generation process to better respond to users' needs. They achieve this by defining control variables such as the initial topic utterance and the dialogue flow [2, 8, 55].

9 Future Directions

A crucial aspect of the conversation generation process is the control over the output data, which pertains to the extent to which the generation can be monitored and how many variables can be actively managed. Current research shows that the primary variables controlled are typically the topic and the dialogue flow. Despite the large volume of synthetic data produced, its quality often remains inadequate when evaluated through downstream tasks. This highlights the need to improve the quality of generated data so that even smaller datasets can yield results comparable to or better than those from human-generated data. Enhanced controllability could also increase data diversity, fairness, and safety [96], reduce bias [40], improve factuality using uncertainty quantification methods [122, 123, 127] and create datasets that pose more significant challenges for dialogue systems [112].

Another limitation of synthetic conversational data is the transferability limitation of dialogs across domains. Often models trained on synthetic data from one domain struggle to generalize to other settings due to differences in structure, required knowledge and intent distribution. However, synthetic approaches allow for generation in large quantities, leveraging the possibility of creating cross-domain datasets that can encompass multiple domains. These capabilities are particularly valuable in developing proactive systems that involve clarification questions, dialogues with shifting targets, mixed-initiative interactions, and reasoning over complex questions centered around entities and knowledge graphs [56, 58, 59]. Therefore, future research should enhance existing challenges of conversational AI such as knowledge groundedness, personalization, safety and bias, and focus on refining the quality and control of the generation process to ensure that the resulting datasets align more closely with the desired characteristics.

Recent studies show that using LLM-generated dialogues as training data and relying on LLM-based evaluators can create a self-reinforcing loop in dialogue system development [101, 132]. LLM evaluators often display self-preference bias, consistently rating their own outputs higher than those of other models or humans [9, 29, 57]. This bias, driven by familiarity and lower perplexity, reduces response diversity and amplifies model weaknesses. As a result, dialogue systems trained and evaluated mainly on LLM-generated data may converge toward homogenized behaviors that reflect evaluator preferences rather than true quality improvements. Overcoming this self-reinforcement bias is an important direction for building more robust and unbiased dialogue systems.

Generating large-scale data for complex and personalized tasks, such as tutoring or modeling user personas, remains a challenge [1, 80, 92]. Recent work on scaling persona generation and personalization benchmarks still struggles to capture the evolving nature of user needs and long-context interactions. Tasks like personalized tutoring require adapting to individual learning styles,

emotions, and progress, while persona-driven dialogues need consistent yet flexible trait modeling. Current pipelines often lack diversity, context continuity, and privacy-aware personalization. Developing scalable methods to produce rich, context-aware, and user-aligned data is an important future research.

References

[1] Zahra Abbasiantaeb, Simon Lupart, Leif Azzopardi, Jeffrey Dalton, and Mohammad Aliannejadi. 2025. Conversational gold: Evaluating personalized conversational search system using gold nuggets. In *Proceedings of the Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*. 3455–3465.

[2] Zahra Abbasiantaeb, Yifei Yuan, Evangelos Kanoulas, and Mohammad Aliannejadi. 2024. Let the LLMs talk: Simulating human-to-human conversational QA via zero-shot LLM-to-LLM interactions. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*. 8–17.

[3] Vaibhav Adlakha, Shehzaad Dhuliawala, Kaheer Suleman, Harm de Vries, and Siva Reddy. 2022. TopiOCQA: Open-domain conversational question answering with topic switching. *Transactions of the Association for Computational Linguistics* 10 (2022), 468–483. DOI: https://doi.org/10.1162/TACL_A_00471

[4] Zishan Ahmad, Suman Saurabh, Vaishakh Menon, Asif Ekbal, Roshni R. Ramnani, and Anutosh Maitra. 2023. INA: An integrative approach for enhancing negotiation strategies with reward-based dialogue agent. In *Findings of the Association for Computational Linguistics: EMNLP*.

[5] Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the ACL*. 6168–6173.

[6] Mohammad Aliannejadi, Zahra Abbasiantaeb, Shubham Chatterjee, Jeffrey Dalton, and Leif Azzopardi. 2024. TREC iKAT 2023: A test collection for evaluating conversational and interactive knowledge assistants. In *Proceedings of the SIGIR 2024: Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*. 819–829.

[7] Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2021. Open-domain question answering goes conversational via question rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 520–534.

[8] Arian Askari, Roxana Petcu, Chuan Meng, Mohammad Aliannejadi, Amin Abolghasemi, Evangelos Kanoulas, and Suzan Verberne. 2025. SOLID: Self-seeding and multi-intent self-instructing LLMs for generating intent-aware information-seeking dialogs. *NAACL* 2025 (2025), 6375–6395.

[9] Krisztian Balog, Don Metzler, and Zhen Qin. 2025. Rankers, judges, and assistants: Towards understanding the interplay of LLMs in information retrieval evaluation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3865–3875.

[10] Krisztian Balog and ChengXiang Zhai. 2024. User simulation for evaluating information access systems. *Foundations and Trends in Information Retrieval* 18, 1-2 (2024), 1–261.

[11] Jianzhu Bao, Rui Wang, Yasheng Wang, Aixin Sun, Yitong Li, Fei Mi, and Ruifeng Xu. 2023. A synthetic data generation framework for grounded dialogues. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*. 10866–10882.

[12] Nolwenn Bernard and Krisztian Balog. 2023. MG-ShopDial: A multi-goal conversational dataset for e-Commerce. In *Proceedings of the Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*. 2775–2785.

[13] Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. MultiWOZ: A large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 5016–5026.

[14] Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - A large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the EMNLP 2018*. 5016–5026.

[15] Arun Tejasvi Chaganty, Megan Leszczynski, Shu Zhang, Ravi Ganti, Krisztian Balog, and Filip Radlinski. 2023. Beyond single items: Exploring user preferences in item sets with the conversational playlist curation dataset. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2754–2764.

[16] Maximillian Chen, Alexandros Papangelis, Chenyang Tao, Seokhwan Kim, Andy Rosenbaum, Yang Liu, Zhou Yu, and Dilek Hakkani-Tur. 2023. PLACES: Prompting language models for social conversation synthesis. In *Proceedings of the EACL*. 814–838.

[17] Maximillian Chen, Alexandros Papangelis, Chenyang Tao, Andy Rosenbaum, Seokhwan Kim, Yang Liu, Zhou Yu, and Dilek Hakkani-Tur. 2022. Weakly supervised data augmentation through prompting for dialogue understanding. *CoRR* abs/2210.14169 (2022). https://doi.org/10.48550/ARXIV.2210.14169

[18] Maximillian Chen, Xiao Yu, Weiyan Shi, Urvi Awasthi, and Zhou Yu. 2023. Controllable mixed-initiative dialogue generation through prompting. In *Proceedings of the ACL 2023 (Short Papers)*. 951–966.

[19] 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*. 2174–2184.

[20] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Fang Zeng, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2025. AugGPT: Leveraging ChatGPT for text data augmentation. *IEEE Trans. Big Data* 11, 3 (2025), 907–918.

[21] Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Y. Zhao, Aida Amini, Qazi Mamunur Rashid, Mike Green, and Kelvin Guu. 2022. Dialog inpainting: Turning documents into dialogs. In *Proceedings of the 39th International Conference on Machine Learning (ICML'22) (Proceedings of Machine Learning Research, Vol. 162)*. 4558–4586.

[22] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2020. CAsT 2020: The conversational assistance track overview. In *Proceedings of the 29th Text Retrieval Conference, TREC (NIST Special Publication, Vol. 1266)*. National Institute of Standards and Technology (NIST).

[23] Jeffrey Dalton, Chenyan Xiong, Vaibhav Kumar, and Jamie Callan. 2020. CAsT-19: A dataset for conversational information seeking. In *Proceedings of the 43rd International ACM SIGIR*. ACM, 1985–1988.

[24] María Àngels de Luis Balaguer, Vinamra Benara, Renato Luiz de Freitas Cunha, Roberto de M. Estevão Filho, Todd Hendry, Daniel Holstein, Jennifer Marsman, Nick Mecklenburg, Sara Malvar, Leonardo O. Nunes, et al. 2024. RAG vs fine-tuning: Pipelines, tradeoffs, and a case study on agriculture. arXiv:2401.08406. Retrieved from https://arxiv.org/abs/2401.08406

[25] Yang Deng, Wenqiang Lei, Minlie Huang, and Tat-Seng Chua. 2023. Goal awareness for conversational AI: Proactivity, non-collaborativity, and beyond. In *Proceedings of the ACL 2023 (Tutorial Abstracts)*. 1–10.

[26] Yang Deng, Wenqiang Lei, Lizi Liao, and Tat-Seng Chua. 2023. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In *Findings of the Association for Computational Linguistics: EMNLP (Findings of ACL, Vol. EMNLP 2023)*. Association for Computational Linguistics, 10602–10621.

[27] Yang Deng, Lizi Liao, Wenqiang Lei, Grace Hui Yang, Wai Lam, and Tat-Seng Chua. 2025. Proactive Conversational AI: A comprehensive survey of advancements and opportunities. *ACM Transactions on Information Systems* 43, 3, Article 67 (2025), 67:1–67:45.

[28] Jan Deriu, Álvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. 2021. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review* 54, 1 (2021), 755–810.

[29] Laura Dietz, Oleg Zendel, Peter Bailey, Charles L. A. Clarke, Ellese Cotterill, Jeff Dalton, Faegheh Hasibi, Mark Sanderson, and Nick Craswell. 2025. Principles and guidelines for the use of LLM judges. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval*. 218–229.

[30] Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander H. Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. 2019. The second conversational intelligence challenge (ConvAI2). arXiv:1902.00098. Retrieved from https://arxiv.org/abs/1902.00098

[31] 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 7th International Conference on Learning Representations*.

[32] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 3029–3051.

[33] Xuan Long Do, Bowei Zou, Liangming Pan, Nancy F. Chen, Shafiq Joty, and Ai Ti Aw. 2022. CoHS-CQG: Context and history selection for conversational question generation. In *Proceedings of the COLING*. 580–591.

[34] Yihao Fang, Xianzhi Li, Stephen W. Thomas, and Xiaodan Zhu. 2023. ChatGPT as Data Augmentation for Compositional Generalization: A Case Study in Open Intent Detection. arXiv:2308.13517. Retrieved from https://arxiv.org/abs/2308.13517

[35] Ryan Fellows, Hisham Ihshaish, Steve Battle, Ciaran Haines, Peter Mayhew, and Juan Ignacio Deza. 2021. Task-oriented dialogue systems: Performance vs. quality-optima, a review. arXiv:2112.11176. Retrieved from https://arxiv.org/abs/2112.11176

[36] Song Feng, Kshitij P. Fadnis, Q. Vera Liao, and Luis A. Lastras. 2020. Doc2Dial: A framework for dialogue composition grounded in documents. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*. 13604–13605.

[37] 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*. 6162–6176.

[38] Yifan Gao, Piji Li, Irwin King, and Michael R. Lyu. 2019. Interconnected question generation with coreference alignment and conversation flow modeling. In *Proceedings of the ACL*. 4853–4862.

[39] Kallirroi Georgila, James Henderson, and Oliver Lemon. 2006. User simulation for spoken dialogue systems: Learning and evaluation. In *Proceedings of the Interspeech 2006*.

[40] Emma J. Gerritse, Faegheh Hasibi, and Arjen P. de Vries. 2020. Bias in conversational search: The double-edged sword of the personalized knowledge graph. In *Proceedings of the 2020 ACM SIGIR International Conference on the Theory of Information Retrieval*. 133–136.

[41] Lewis R. Goldberg. 1993. The structure of phenotypic personality traits. *American Psychologist* 48, 1 (1993), 26–34.

[42] Karthik Gopalakrishnan, Behnam Hedayatnia, Qinlang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, and Dilek Hakkani-Tür. 2019. Topical-chat: Towards knowledge-grounded open-domain conversations. In *Proceedings of the Interspeech 2019*. 1891–1895.

[43] Jing Gu, Mostafa Mirshekari, Zhou Yu, and Aaron Sisto. 2021. ChainCQG: Flow-aware conversational question generation. In *Proceedings of the EACL*. 2061–2070.

[44] Aman Gupta, Anup Shirgaonkar, Angels de Luis Balaguer, Bruno Silva, Daniel Holstein, Dawei Li, Jennifer Marsman, Leonardo O. Nunes, Mahsa Rouzbahman, Morris Sharp, et al. 2024. RAG vs fine-tuning: Pipelines, tradeoffs, and a case study on agriculture. arXiv:2401.08406. Retrieved from https://arxiv.org/abs/2401.08406

[45] Izzeddin Gur, Dilek Hakkani-Tür, Gökhan Tür, and Pararth Shah. 2018. User modeling for task oriented dialogues. In *Proceedings of the 2018 IEEE Spoken Language Technology Workshop*. 900–906.

[46] He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling strategy and generation in negotiation dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2333–2343.

[47] Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 263–272.

[48] Chao-Wei Huang, Chen-Yu Hsu, Tsu-Yuan Hsu, Chen-An Li, and Yun-Nung Chen. 2023. CONVERSER: Few-shot conversational dense retrieval with synthetic data generation. In *Proceedings of the Meeting of the Special Interest Group on Discourse and Dialogue*. 381–387.

[49] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems* 43, 2 (2025), 42:1–42:55. DOI: https://doi.org/10.1145/3703155

[50] Seonjeong Hwang, Yunsu Kim, and Gary Lee. 2022. Multi-type conversational question-answer generation with closed-ended and unanswerable questions. In *Proceedings of the AACL/IJCNLP 2022 (AACL/IJCNLP), Short Papers*. 169–177.

[51] Seonjeong Hwang, Yunsu Kim, and Gary Geunbae Lee. 2022. Multi-type conversational question-answer generation with closed-ended and unanswerable questions. In *Proceedings of the AACL*. 169–177.

[52] Seonjeong Hwang and Gary Lee. 2022. Conversational QA dataset generation with answer revision. In *Proceedings of the COLING*. 1636–1644.

[53] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z. Leibo, David Silver, and Koray Kavukcuoglu. 2017. Reinforcement learning with unsupervised auxiliary tasks. In *Proceedings of the International Conference on Learning Representations*.

[54] Pegah Jandaghi, XiangHai Sheng, Xinyi Bai, Jay Pujara, and Hakim Sidahmed. 2024. Faithful persona-based conversational dataset generation with large language models. In *Findings of the Association for Computational Linguistics, ACL (Findings of ACL, Vol. ACL 2024)*. Association for Computational Linguistics, 15245–15270.

[55] Hideaki Joko, Shubham Chatterjee, Andrew Ramsay, Arjen P. de Vries, Jeff Dalton, Faegheh Hasibi, Krisztian Balog, and Arjen P. de Vries. 2024. Doing personal laps: LLM-augmented dialogue construction for personalized multi-session conversational search. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[56] Hideaki Joko and Faegheh Hasibi. 2022. Personal entity, concept, and named entity linking in conversations. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*. 4099–4103.

[57] Hideaki Joko and Faegheh Hasibi. 2025. FACE: A fine-grained reference free evaluator for conversational recommender systems. *CoRR* abs/2506.00314 (2025). https://doi.org/10.48550/ARXIV.2506.00314

[58] Hideaki Joko, Faegheh Hasibi, Krisztian Balog, and Arjen P. de Vries. 2021. Conversational entity linking: Problem definition and datasets. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2390–2397.

[59] Magdalena Kaiser, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Reinforcement learning from reformulations in conversational question answering over knowledge graphs. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 459–469.

[60] Gangwoo Kim, Sungdong Kim, Kang Min Yoo, and Jaewoo Kang. 2022. Generating information-seeking conversations from unlabeled documents. In *Proceedings of the EMNLP 2022*. 2362–2378.

[61] Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, et al. 2023. SODA: Million-scale dialogue distillation with social commonsense contextualization. In *Proceedings of the EMNLP 2023*. 12930–12949.

[62] Minju Kim, Chaehyeong Kim, Yong Ho Song, Seung-won Hwang, and Jinyoung Yeo. 2022. BotsTalk: Machine-sourced framework for automatic curation of large-scale multi-skill dialogue datasets. In *Proceedings of the EMNLP 2022*. 5149–5170.

[63] Sungdong Kim, Minsuk Chang, and Sang-Woo Lee. 2021. NeuralWOZ: Learning to collect task-oriented dialogue via model-based simulation. In *Proceedings of the ACL*. 3704–3717.

[64] Seokhwan Kim, Luis Fernando D'Haro, Rafael E. Banchs, Jason D. Williams, and Matthew Henderson. 2016. The fourth dialog state tracking challenge. In *Proceedings of the International Workshop on Spoken Dialogue Systems*. 435–449.

[65] Florian Kreyssig, Iñigo Casanueva, Paweł Budzianowski, and Milica Gašić. 2018. Neural user simulation for corpus-based policy optimisation of spoken dialogue systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*. 60–69.

[66] Young-Jun Lee, Chae-Gyun Lim, Yunsu Choi, Ji-Hui Lm, and Ho-Jin Choi. 2022. PERSONACHATGEN: Generating personalized dialogues using GPT-3. In *Proceedings of the 1st Workshop on Customized Chat Grounding Persona and Knowledge*. 29–48.

[67] Megan Leszczynski, Shu Zhang, Ravi Ganti, Krisztian Balog, Filip Radlinski, Fernando Pereira, and Arun Tejasvi Chaganty. 2023. Talk the walk: Synthetic data generation for conversational music recommendation. arXiv:2301.11489. Retrieved from https://arxiv.org/abs/2301.11489

[68] Megan Eileen Leszczynski, Ravi Ganti, Shu Zhang, Krisztian Balog, Filip Radlinski, Fernando Pereira, and Arun Tejasvi Chaganty. 2022. Conversational music retrieval with synthetic data. In *Proceedings of the 2nd Workshop on Interactive Learning for Natural Language Processing at NeurIPS*.

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

[70] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the Conference of the North American Chapter of the ACL*. 110–119.

[71] Siheng Li, Yichun Yin, Cheng Yang, Wangjie Jiang, Yiwei Li, Zesen Cheng, Lifeng Shang, Xin Jiang, Qun Liu, and Yujiu Yang. 2023. NewsDialogues: Towards proactive news grounded conversation. In *Findings of the Association for Computational Linguistics*. 3634–3649.

[72] Xiujun Li, Zachary C. Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. 2016. A user simulator for task-completion dialogues. arXiv:1612.05688. Retrieved from https://arxiv.org/abs/1612.05688

[73] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the 8th International Joint Conference on Natural Language Processing*. 986–995.

[74] Yiming Li and Zhao Zhang. 2025. The first place solution of WSDM Cup 2024: Leveraging large language models for conversational Multi-Doc QA. *CoRR* abs/2402.18385 (2024). https://doi.org/10.48550/ARXIV.2402.18385

[75] Zekun Li, Wenhu Chen, Shiyang Li, Hong Wang, Jing Qian, and Xifeng Yan. 2022. Controllable dialogue simulation with in-context learning. In *Proceedings of the EMNLP*. 4330–4347.

[76] Hsien-chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gasic. 2021. Domain-independent user simulation with transformers for task-oriented dialogue systems. In *Proceedings of the SIGDIAL*. 445–456.

[77] I-Fan Lin, Faegheh Hasibi, and Suzan Verberne. 2024. Generate then refine: Data augmentation for zero-shot intent detection. In *Findings of the Association for Computational Linguistics: EMNLP (Findings of ACL, Vol. EMNLP 2024)*. Association for Computational Linguistics, 13138–13146.

[78] Yen Ting Lin, Alexandros Papangelis, Seokhwan Kim, and Dilek Hakkani-Tur. 2022. Knowledge-grounded conversational data augmentation with generative conversational networks. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 26–38.

[79] Anqi Liu, Bo Wang, Yue Tan, Dongming Zhao, Kun Huang, Ruifang He, and Yuexian Hou. 2023. MTGP: Multi-turn target-oriented dialogue guided by generative global path with flexible turns. In *Findings of the Association for Computational Linguistics: ACL 2023*. 259–271.

[80] Ben Liu, Jihai Zhang, Fangquan Lin, Xu Jia, and Min Peng. 2025. One size doesn't fit all: A personalized conversational tutoring agent for mathematics instruction. In *Proceedings of the Web Conference Companion*. 2401–2410.

[81] Siyang Liu, Chujie Zheng, Orianna Demasi, Sahand Sabour, Yu Li, Zhou Yu, Yong Jiang, and Minlie Huang. 2021. Towards emotional support dialog systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 3469–3483.

[82] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the Empirical Methods in Natural Language Processing*. 2511–2522.

[83] Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, and Wanxiang Che. 2021. DuRecDial 2.0: A bilingual parallel corpus for conversational recommendation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 4335–4347.

[84] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. In *Proceedings of the Advances in Neural Information Processing Systems 36*.

[85] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the ACL 2023*. 9802–9822.

[86] Ramesh Manuvirakurike, Jacqueline Brixey, Trung Bui, Walter Chang, Ron Artstein, and Kallirroi Georgila. 2018. DialEdit: Annotations for spoken conversational image editing. In *Proceedings of the Joint Workshop on Interoperable Semantic Annotation*. 1–9.

[87] Kelong Mao, Zhicheng Dou, Hongjin Qian, Fengran Mo, Xiaohua Cheng, and Zhao Cao. 2022. ConvTrans: Transforming web search sessions for conversational dense retrieval. In *Proceedings of the Empirical Methods in Natural Language Processing*. 2935–2946.

[88] Shikib Mehri and Maxine Eskénazi. 2020. USR: An unsupervised and reference free evaluation metric for dialog generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 681–707.

[89] Helen M. Meng, Carmen Wai, and Roberto Pieraccini. 2003. The use of belief networks for mixed-initiative dialog modeling. *IEEE Transactions on Speech and Audio Processing* 11, 6 (2003), 757–773.

[90] Fengran Mo, Bole Yi, Kelong Mao, Chen Qu, Kaiyu Huang, and Jian-Yun Nie. 2024. ConvSDG: Session data generation for conversational search. In *Proceedings of the Web Conference Companion*. 1634–1642.

[91] Biswesh Mohapatra, Gaurav Pandey, Danish Contractor, and Sachindra Joshi. 2021. Simulated chats for building dialog systems: Learning to generate conversations from instructions. In *Proceedings of the EMNLP*. 1190–1203.

[92] Jisoo Mok, Ik-hwan Kim, Sangkwon Park, and Sungroh Yoon. 2025. Exploring the potential of LLMs as personalized assistants: Dataset, evaluation, and analysis. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 10212–10239.

[93] Srini Narayanan and Sheila A. McIlraith. 2002. Simulation, verification and automated composition of web services. In *Proceedings of the 11th International World Wide Web Conference*. 77–88.

[94] Jinjie Ni, Tom Young, Vlad Pandelea, Fuzhao Xue, and Erik Cambria. 2023. Recent advances in deep learning based dialogue systems: A systematic survey. *Artif. Intell. Rev.* 56, 4 (2023), 3055–3155.

[95] Jinjie Ni, Tom Young, Vlad Pandelea, Fuzhao Xue, and Erik Cambria. 2023. Recent advances in deep learning based dialogue systems: A systematic survey. *Artif. Intell. Rev.* (2023), 3055–3155.

[96] Alexandra Olteanu, Jean Garcia-Gathright, Maarten de Rijke, Michael D. Ekstrand, Adam Roegiest, Aldo Lipani, Alex Beutel, Ana Lucic, Ana-Andreea Stoica, Anubrata Das, Asia Biega, Bart Voorn, Claudia Hauff, Damiano Spina, David D. Lewis, Douglas W. Oard, Emine Yilmaz, Faegheh Hasibi, Gabriella Kazai, Graham McDonald, Hinda Haned, Iadh Ounis, Ilse van der Linden, Joris Baan, Kamuela N. Lau, Krisztian Balog, Mahmoud F. Sayed, Maria Panteli, Mark Sanderson, Matthew Lease, Preethi Lahoti, and Toshihiro Kamishima. 2019. FACTS-IR: Fairness, accountability, confidentiality, transparency, and safety in information retrieval. *SIGIR Forum* 53, 2 (2019), 20–43.

[97] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Processing of the Advances in Neural Information Systems 35: Annual Conference on Neural Information Processing Systems*.

[98] Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2024. Fine-tuning or retrieval? comparing knowledge injection in LLMs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 237–250.

[99] Paul Owoicho, Jeff Dalton, Mohammad Aliannejadi, Leif Azzopardi, Johanne R. Trippas, and Svitlana Vakulenko. 2022. TREC CAsT 2022: Going beyond user ask and system retrieve with initiative and response generation. In *Proceedings of the 31st Text REtrieval Conference, TREC (NIST Special Publication, Vol. 500-338)*. National Institute of Standards and Technology (NIST).

[100] Boyuan Pan, Hao Li, Ziyu Yao, Deng Cai, and Huan Sun. 2019. Reinforced dynamic reasoning for conversational question generation. In *Proceedings of the ACL*.

[101] Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. In *Proceedings of the Advances in Neural Information Processing Systems*.

[102] Yookoon Park, Jaemin Cho, and Gunhee Kim. 2018. A hierarchical latent structure for variational conversation modeling. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1792–1801.

[103] Gustavo Penha, Alexandru Balan, and Claudia Hauff. 2019. Introducing MANtIS: A novel multi-domain information seeking dialogues dataset. arXiv:1912.04639. Retrieved from https://arxiv.org/abs/1912.04639

[104] Peng Qi, Nina Du, Christopher Manning, and Jing Huang. 2023. PragmatiCQA: A dataset for pragmatic question answering in conversations. In *Findings of the Association for Computational Linguistics: ACL 2023*. 6175–6191.

[105] Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and characterizing user intent in information-seeking conversations. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 989–992.

[106] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21 (2020), 140:1–140:67. https://jmlr.org/papers/v21/20-074.html

[107] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5370–5381.

[108] Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI*. 8689–8696.

[109] Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *TACL* 7 (2019), 249–266. DOI: https://doi.org/10.1162/TACL_A_00266

[110] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the Empirical Methods in NLP and Int'l Joint Conf. on NLP*. 3980–3990.

[111] Federico Ruggeri, Mohsen Mesgar, and Iryna Gurevych. 2023. A dataset of argumentative dialogues on scientific papers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*. 7684–7699.

[112] Chris Samarinas, Pracha Promthaw, Atharva Nijasure, Hansi Zeng, Julian Killingback, and Hamed Zamani. 2024. Simulating task-oriented dialogues with state transition graphs and large language models. arXiv:2404.14772. Retrieved from https://arxiv.org/abs/2404.14772

[113] Anna Sauer, Shima Asaadi, and Fabian Küch. 2022. Knowledge distillation meets few-shot learning: An approach for few-shot intent classification within and across domains. In *Proceedings of the Workshop on NLP for Conversational AI*. 108–119.

[114] Jost Schatzmann, Blaise Thomson, and Steve J. Young. 2007. Statistical user simulation with a hidden agenda. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*. 273–282.

[115] Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3795–3805.

[116] Karin Sevegnani, David M. Howcroft, Ioannis Konstas, and Verena Rieser. 2021. OTTERS: One-turn topic transitions for open-domain dialogue. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*. 2492–2504.

[117] Pararth Shah, Dilek Hakkani-Tür, Gökhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry P. Heck. 2018. Building a conversational agent overnight with dialogue self-play. *CoRR* abs/1801.04871. http://arxiv.org/abs/1801.04871

[118] Eric Michael Smith, Orion Hsu, Rebecca Qian, Stephen Roller, Y-Lan Boureau, and Jason Weston. 2022. Human evaluation of conversations is an open problem: Comparing the sensitivity of various methods for evaluating dialogue agents. In *Proceedings of the 4th Workshop on NLP for Conversational AI*. 77–97.

[119] Heydar Soudani. 2025. Enhancing knowledge injection in large language models for efficient and trustworthy responses. In *Proceedings of the Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*. 4211.

[120] Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2023. Data augmentation for conversational AI. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 5220–5223.

[121] Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2024. Fine tuning vs. retrieval augmented generation for less popular knowledge. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific*. 12–22.

[122] Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2025. Why uncertainty estimation methods fall short in RAG: An axiomatic analysis. In *Proceedings of the ACL 2025 Findings*. 16596–16616.

[123] Heydar Soudani, Hamed Zamani, and Faegheh Hasibi. 2025. Uncertainty quantification for retrieval-augmented reasoning. arXiv:2510.11483. Retrieved from https://arxiv.org/abs/2510.11483

[124] Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric Xing, and Zhiting Hu. 2019. Target-guided open-domain conversation. In *Proceedings of the ACL*. 5624–5634.

[125] Silvia Terragni, Modestas Filipavicius, Nghia Khau, Bruna Guedes, André Manso, and Roland Mathis. 2023. In-context learning user simulators for task-oriented dialog systems. *CoRR* abs/2306.00774. https://doi.org/10.48550/ARXIV.2306.00774

[126] Bo-Hsiang Tseng, Yinpei Dai, Florian Kreyssig, and Bill Byrne. 2021. Transferable dialogue systems and user simulators. In *Proceedings of the ACL, IJCNLP*. 152–166.

[127] Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, et al. 2025. Benchmarking uncertainty quantification methods for large language models with LM-polygraph. *Transactions of the Association for Computational Linguistics* 13 (2025), 220–248. DOI: https://doi.org/10.1162/tacl_a_00737

[128] Dazhen Wan, Zheng Zhang, Qi Zhu, Lizi Liao, and Minlie Huang. 2022. A unified dialogue user simulator for few-shot data augmentation. In *Proceedings of the EMNLP*. 3788–3799.

[129] Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model.

[130] Jian Wang, Yi Cheng, Dongding Lin, Chak Tou Leong, and Wenjie Li. 2023. Target-oriented proactive dialogue systems with personalization: Problem formulation and dataset curation. In *Proceedings of the EMNLP 2023*. 1132–1143.

[131] Weiyan Wang, Yuxiang Wu, Yu Zhang, Zhongqi Lu, Kaixiang Mo, and Qiang Yang. 2017. Integrating user and agent models: A deep task-oriented dialogue system. arXiv:1711.03697. Retrieved from https://arxiv.org/abs/1711.03697

[132] Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. 2024. Self-preference bias in LLM-as-a-judge. arXiv:2410.21819. Retrieved from https://arxiv.org/abs/2410.21819

[133] Joseph Weizenbaum. 1966. ELIZA – a computer program for the study of natural language communication between man and machine. *Communications of the ACM* 9, 1 (1966), 36–45.

[134] Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve J. Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. 438–449.

[135] Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve J. Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the Conference of the European Chapter of the ACL*. 438–449.

[136] Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve J. Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the EACL 2017*. 438–449.

[137] Peter West, Chandra Bhagavatula, Jack Hessel, Jena D. Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: From general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4602–4625.

[138] Qingyang Wu, Song Feng, Derek Chen, Sachindra Joshi, Luis Lastras, and Zhou Yu. 2022. DG2: Data augmentation through document grounded dialogue generation. In *Proceedings of the SIGDIAL*. 204–216.

[139] Zeqiu Wu, Ryu Parish, Hao Cheng, Sewon Min, Prithviraj Ammanabrolu, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. INSCIT: Information-seeking conversations with mixed-initiative interactions. *Trans. Assoc. Comput. Linguistics* 11 (2023), 453–468.

[140] Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 6268–6278.

[141] Jing Xu, Megan Ung, Mojtaba Komeili, Kushal Arora, Y-Lan Boureau, and Jason Weston. 2023. Learning new skills after deployment: Improving open-domain internet-driven dialogue with human feedback. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*. 13557–13572.

[142] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W. Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 245–254.

[143] Shiquan Yang, Rui Zhang, and Sarah M. Erfani. 2020. GraphDialog: Integrating graph knowledge into end-to-end task-oriented dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 1878–1888.

[144] Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. G-DAug: Generative data augmentation for commonsense reasoning. In *Proceedings of the EMNLP*. 1008–1025.

[145] Zhitong Yang, Bo Wang, Jinfeng Zhou, Yue Tan, Dongming Zhao, Kun Huang, Ruifang He, and Yuexian Hou. 2022. TopKG: Target-oriented dialog via global planning on knowledge graph. In *Proceedings of the COLING*. 745–755.

[146] Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, et al. 2025. Justice or prejudice? quantifying biases in LLM-as-a-judge. In *The Thirteenth International Conference on Learning Representations, ICLR*. OpenReview.net.

[147] Zihao Yi, Jiarui Ouyang, Yuwen Liu, Tianhao Liao, Zhe Xu, and Ying Shen. 2026. A survey on recent advances in LLM-based multi-turn dialogue systems. *ACM Comput. Surv.* 58, 6 (2026), 148:1–148:38.

[148] Kang Min Yoo, Hanbit Lee, Franck Dernoncourt, Trung Bui, Walter Chang, and Sang-goo Lee. 2020. Variational hierarchical dialog autoencoder for dialog state tracking data augmentation. In *Proceedings of the EMNLP*. 3406–3425.

[149] Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2023. Large Language Model as Attributed Training Data Generator: A Tale of Diversity and Bias. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems (NeurIPS'23)*.

[150] Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating generated text as text generation. In *Proceedings of the Advances in Neural Information Processing Systems*. 27263–27277.

[151] Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. 2023. Conversational information seeking. *Found. Trends Inf. Retr.* 17, 3–4 (2023), 244–456.

[152] Jiahao Zhang, Haiyang Zhang, Dongmei Zhang, Liu Yong, and Shen Huang. 2024. End-to-end beam retrieval for multi-hop question answering. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics, 1718–1731.

[153] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too?. In *Proceedings of the ACL 2018*. 2204–2213.

[154] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *Proceedings of the 8th International Conference on Learning Representations*.

[155] Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018. Generating informative and diverse conversational responses via adversarial information maximization. In *Proceedings of the Advances in Neural Information Processing Systems*. 1815–1825.

[156] Qihao Zhao, Yalun Dai, Hao Li, Wei Hu, Fan Zhang, and Jun Liu. 2024. LTGC: Long-tail recognition via leveraging LLMs-driven generated content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19510–19520.

[157] Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Proceedings of the NAACL-HLT*. 1208–1218.

[158] Chujie Zheng, Sahand Sabour, Jiaxin Wen, Zheng Zhang, and Minlie Huang. 2023. AugESC: Dialogue augmentation with large language models for emotional support conversation. In *Findings of the Association for Computational Linguistics: ACL 2023*. 1552–1568.

[159] Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the Empirical Methods in Natural Language Processing*. 2023–2038.

[160] Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *Proceedings of the Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*. 1097–1100.

Received 23 August 2024; revised 15 December 2025; accepted 22 January 2026