Leveraging Low-Level Symbolic Competences for Unsupervised Grounding in Hallucination Detection

Anonymous ACL submission

Abstract

Hallucination—where a language model generates outputs that are factually incorrect or unsupported by the source—is a major challenge for both prompted and fine-tuned language models. Detecting hallucinations is difficult due to the opaque reasoning processes of LLMs, which often provide little insight into why a model's output may be inaccurate.

In this work, we investigate whether an LLM can use an alternative, low level, symbolic competence such as SQL for unsupervised hallucination detection in some high level task. For this, we make an LLM build an SQL database from reference documents. This SQL database is then used for reasoning over the reference and the sampled response in a hallucination detection pipeline that is grounded in the database, thereby providing a neurosymbolic checkup.

On RAGTruth and DiaHalu hallucination detection datasets, we find that our approach improves on direct prediction and competes with state-of-the-art hallucination detection methods, while not requiring domain-specific fine-tuning. Instead it relies on a low-level general competence already present in LLMs. This warrants further investigation of low-level LLM competences in neurosymbolic approaches. [1]

1 Introduction

Large language models (LLMs) excel across natural language processing tasks due to large-scale pretraining and instruction tuning (Brown et al., 2020; Ouyang et al., 2022). Consequently, they form the backbone of modern AI systems by enabling fluent interactions.

Despite this success, LLMs frequently generate factually incorrect outputs unsupported by the source context, a phenomenon known as *hallucination* (Ji et al., 2023; Huang et al., 2024). These hallucinations pose significant risks in high-stakes domains like medicine and law, where misinformation undermines decision-making (Weidinger et al., 2022).

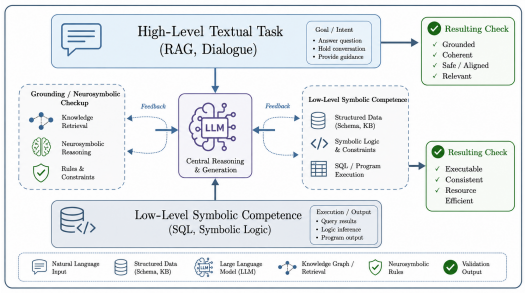

Figure 1: Grounding hallucination detection using low-level symbolic competences.

Detecting hallucinations remains challenging because the opaque reasoning processes of LLMs provide little insight into why an output is inaccurate (Doshi-Velez and Kim, 2017). While existing detection methods achieve strong performance, they often act as black boxes (Manakul et al., 2023; Farquhar et al., 2024; Chen et al., 2024a). This opacity persists even in modern retrieval-augmented generation (RAG; Lewis et al., 2020) systems, offering limited interpretability regarding unsupported claims (Niu et al., 2024; Song et al., 2024; Kovács and Recski, 2025; Dubanowska et al., 2025).

To improve robustness, recent work explores grounding LLM reasoning in structured or symbolic representations (West et al., 2022; Yang et al., 2025). Furthermore, Dennett (1987) argues that explicit task comprehension is not strictly necessary for task competence (Figure 1). Building on this, the TeQoDO pipeline (Vukovic et al., 2025) demonstrates that an LLM's lower-level competence in SQL can be leveraged to construct a reliable knowledge base.

In this work, we investigate whether an LLM can utilise this lower-level competence for unsupervised hallucination detection. We propose **TeQHallu**, a **T**ext-to-**S**QL pipeline for

[1] Code will be released upon publication.

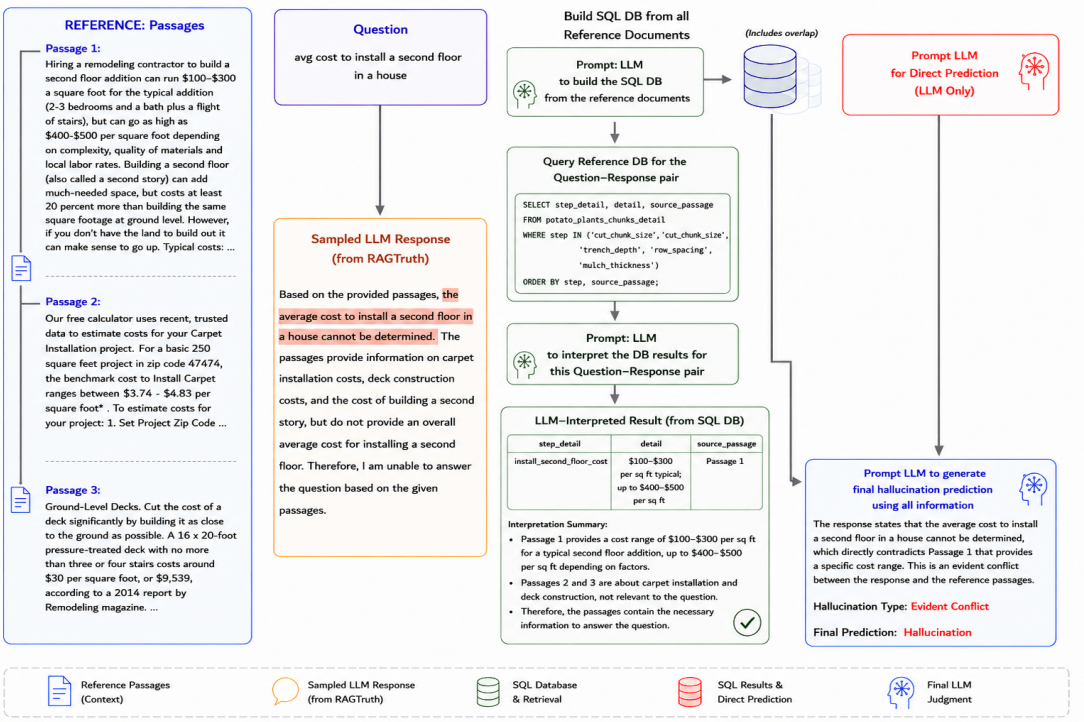

Figure 2: Text-to-SQL grounded hallucination detection with an example from the RAGTruth QA task.

Hallucination detection that builds on TeQoDO. Our approach prompts an LLM to build a relational SQL database from reference documents, which is then used to reason over both the reference and the response. Grounding the detection pipeline in executable SQL queries provides a neurosymbolic (Garcez et al., 2019; d'Avila Garcez and Lamb, 2023) checkup without requiring domain-specific fine-tuning.

Our contributions can be summarised as follows:

- We introduce TeQHallu, an unsupervised text-to-SQL method that leverages low-level LLM competence to structure reference knowledge for hallucination detection.
- We demonstrate that grounding detection in an SQL database provides transparent, inspectable reasoning traces compared to black-box approaches.
- We evaluate our pipeline on the RAGTruth and DiaHalu benchmarks, performing competitively against state-of-the-art methods without domain-specific fine-tuning.
- We highlight neurosymbolic checkups via lower-level, task-agnostic competences as a promising avenue for improving LLM reliability.

2 Related Work

2.1 Hallucination Detection

Hallucination detection identifies factually incorrect or unsupported model outputs to enhance LLM reliability (Ji et al., 2023; Huang et al., 2024). Early strategies primarily relied on sequence-level metrics and heuristic rules to flag improbable generations (Lin et al., 2022).

Recent methods evaluate internal model states and self-consistency, using sampling-based consistency (Manakul et al., 2023), hidden layer signals (Li et al., 2025), or semantic entropy (Farquhar et al., 2024) to estimate uncertainty. In retrieval-augmented generation (RAG) contexts, cross-referencing model outputs with external knowledge serves as the primary safeguard (Lewis et al., 2020; Niu et al., 2024). However, these techniques typically function as "black boxes", yielding binary labels without interpretable reasoning or granular justifications (Sriramanan et al., 2024).

2.2 Text-to-SQL and Structured Reasoning

Text-to-SQL methodologies allow LLMs to use executable queries as a structured reasoning tool, enhancing factual reliability (Liu et al., 2025; Gao et al., 2024). Advanced frameworks like DIN-SQL decompose complex queries into verifiable subtasks to improve generation accuracy (Pourreza and Rafiei, 2023). Whilst Vukovic et al. (2025) utilise

text-to-SQL to construct task-oriented dialogue ontologies, they lack a downstream application for the generated knowledge base. Our framework addresses this gap by repurposing these structured ontologies specifically for hallucination detection.

Neurosymbolic methods integrate language models with symbolic constraints to improve prediction quality, robustness, and interpretability (Princis et al., 2025). Grounding reasoning in executable, symbolic formats yields both tangible performance gains and superior explainability.

However, existing text-to-SQL systems primarily focus on query accuracy rather than hallucination verification (Pourreza and Rafiei, 2023; Gao et al., 2024). The deployment of text-to-SQL pipelines as explicit evaluation mechanisms in dialogue systems remains largely understudied. This motivates the TeQHallu approach, which leverages SQL grounding to provide a factual anchor and a transparent audit trail for model responses.

3 The TeQHallu Framework

TeQHallu is a neurosymbolic framework for unsupervised hallucination detection in retrieval-augmented generation settings. It combines probabilistic language generation with deterministic symbolic verification. Rather than relying solely on neural reasoning, it converts retrieved text into a structured, queryable relational representation. TeQHallu is illustrated in Figure 2 and the full prompt in Appendix A.

3.1 Incremental Database Construction via Multi-step Prompting

TeQHallu constructs a relational database from retrieved passages through multi-step prompting. The process begins by prompting the LLM to inspect the current schema using metadata queries such as `PRAGMA table_info`. This provides symbolic awareness of existing tables, columns, and stored information. The model then analyses retrieved passages to identify information absent from the database. Instead of rebuilding the schema, the framework extracts the symbolic delta between textual evidence and the current representation.

Based on this analysis, the LLM generates database operations to insert new information or extend the schema. The prompting encourages the conversion of natural language variables, measurements, entities, and qualifiers into relational tuples. In our example, the model extracts second-floor cost information, storing numerical ranges with semantic labels and source provenance. The statement that second-floor construction typically costs $100–$300 per square foot, rising to $400–$500, is encoded as structured symbolic evidence. This incremental strategy maintains a high-fidelity relational abstraction of retrieved context without domain-specific supervision.

3.2 Claim Grounding via Symbolic Retrieval

After database construction, the framework grounds claims through symbolic retrieval. The generated response is first analysed to isolate claims requiring factual verification. The system then prompts the LLM to generate targeted `SELECT` queries conditioned on the original question and response. These prompts guide retrieval of database entries most relevant to validation. The resulting SQL queries are executed to obtain symbolic evidence derived directly from the source passages. This transforms verification from implicit reasoning into an explicit retrieval task grounded in database operations.

In the example, the response claims that the average cost of installing a second floor cannot be determined from the passages. The SQL retrieval step instead returns stored cost information from the reference material. The retrieved evidence includes a typical range of $100–$300 per square foot and estimates up to $400–$500. The database therefore contains sufficient information to answer the query. Because the response asserts informational absence despite available evidence, the framework detects a contradiction between the response and symbolic retrieval outputs. The final assessment is thus grounded in the database rather than solely in the model's internal parameters.

3.3 Neurosymbolic Checkup and Alignment

The final stage performs a neurosymbolic checkup that reconciles neural predictions with symbolic retrieval results. An initial hallucination prediction is generated through prompting based only on the LLM's internal reasoning. This serves as a neural-only baseline judgement. A second prompting stage then provides SQL retrieval outputs, interpreted database summaries, and relevant textual context. The model is instructed to compare its preliminary judgement with the retrieved evidence and justify its final decision.

The framework evaluates whether a response is

fully supported, partially supported, unsupported, or contradicted by database results. In Figure 2, the response states that the average installation cost cannot be determined from the passages. This directly conflicts with structured evidence retrieved from the relational database. The framework therefore classifies the response as an *Evident Conflict*. The final output includes both the hallucination prediction and a structured explanation grounded in SQL retrieval results. This multi-stage procedure produces an interpretable reasoning trace with greater transparency than purely neural detection approaches.

4 Experiments

4.1 RAGTruth Dataset

We evaluate TeQHallu on the RAGTruth dataset (Niu et al., 2024), which contains naturally occurring hallucinations across three distinct retrieval-augmented tasks: question answering (MS MARCO; Bajaj et al., 2016), data-to-text generation (Yelp; Yel, 2021), and summarisation (CNN/Daily Mail; See et al., 2017). Each test set contains 900 document-response pairs. The evaluation focus centers on model-generated responses sampled from a diverse array of foundational architectures, including GPT-3.5 (OpenAI, 2023a), GPT-4 (OpenAI, 2023b), and LLaMA variants (Touvron et al., 2023). The dataset captures errors regarding the retrieved document, categorised into four fine-grained types: evident conflict, subtle conflict, introduction of baseless information, and subtle introduction of baseless information. Each response is manually annotated, and following the protocol in (Niu et al., 2024), we evaluate binary response-level hallucination detection to determine whether a hallucination is present or not. We report precision, recall, and F1-score for each individual task alongside macro-averaged results across the entire benchmark.

4.2 DiaHalu Dataset: TOD Portion

To assess performance on multi-turn dialogue data, we utilise the task-oriented dialogue (TOD) portion of the DiaHalu dataset (Chen et al., 2024b). This segment is anchored by structured database intents from MultiWOZ 2.1 (Eric et al., 2020) and DSTC 1.0 (Williams et al., 2013), categorising hallucinations into non-factual, incoherence, irrelevance and over-reliance. Conversations were generated using a self-chat framework with GPT-3.5 (OpenAI, 2023a) and GPT-4 (OpenAI, 2023b) based on distinct prompts for user goals and system responses. The final expert-annotated corpus comprises 210 dialogues, consisting of 87 MultiWOZ-based dialogues, 56 DSTC 1.0-based dialogues, and 67 conversations based on synthetic GPT-4 user intents.

4.3 Implementation Details

Our framework utilises *Gemini 2.5 Flash* (Google, 2025) via the Google Vertex AI API[2] as the main model for both direct hallucination prediction and text-to-SQL grounded reasoning, with open-weight model configurations deferred to Section 4.8 for clarity of the main results. Gemini 2.5 Flash is a general-purpose Mixture-of-Experts (MoE; Jacobs et al., 1991; Shazeer et al., 2017) model optimised for computational efficiency and robust semantic reasoning. The TeQHallu framework adopts the structured database construction, prompting strategy, and SQL interpretation logic established by the TeQoDO framework (Vukovic et al., 2025) to maintain consistency in database synthesis. For the DiaHalu dataset evaluation, the underlying databases are constructed directly from the text of the multi-turn dialogues under inspection.

4.4 Baselines

Initial RAGTruth Baselines Niu et al. (2024) established the benchmark using both prompt-based and fine-tuned models. Zero-shot configurations of GPT-3.5 and GPT-4 establish a baseline performance floor, while SelfCheckGPT (Manakul et al., 2023) identifies factual inconsistencies via stochastic sampling across multiple model generations. The LMvLM framework (Cohen et al., 2023) applies a cross-examination strategy, utilising an external LLM to generate verification questions targeting specific response claims. Furthermore, a fine-tuned Llama-2-13B model (Touvron et al., 2023) serves as a supervised baseline, illustrating the performance advantages of domain-specific training over general-purpose prompting.

SOTA Fine-tuned Baselines We compare TeQHallu against several state-of-the-art architectures optimized for robust hallucination detection. RAG-HAT (Song et al., 2024) utilizes Hallucination-Aware Tuning to generate binary labels and error descriptions for Direct Preference Optimisation (Rafailov et al., 2024). LettuceDetect (Kovács and Recski, 2025) employs a lightweight

[2] https://cloud.google.com/vertex-ai

Methods	Question Answering			Data-to-text Writing			Summarization			Overall Macro		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
(Niu et al., 2024) initial baselines												
Prompt$_{\text{gpt-3.5-turbo}}$	18.8	84.4	30.8	65.1	95.5	77.4	23.4	89.2	37.1	35.8	89.7	48.4
Prompt$_{\text{gpt-4-turbo}}$	33.2	90.6	45.6	64.3	100.0	78.3	31.5	97.6	47.6	43.0	96.1	57.2
SelfCheckGPT$_{\text{gpt-3.5-turbo}}$	35.0	58.0	43.7	68.2	82.8	74.8	31.1	56.5	40.1	44.8	65.8	52.9
LMvLM$_{\text{gpt-4-turbo}}$	18.7	76.9	30.1	68.0	76.7	72.1	23.3	81.9	36.2	36.7	78.5	46.1
Finetuned Llama-2-13B	61.6	76.3	68.2	85.4	91.0	88.1	64.0	54.9	59.1	70.3	74.1	71.8
SOTA fine-tuned baselines												
RAG-HAT (LLaMA-3-8B) (Song et al., 2024)	76.5	73.1	74.8	92.9	90.3	91.6	77.7	59.8	67.6	82.4	74.4	78.0
lettuceedetect-base-v1 (150M)(Kovács and Recski, 2025)	60.6	71.3	65.5	89.3	86.5	87.9	53.9	47.5	50.5	67.9	68.4	68.0
lettuceedetect-large-v1 (396M) (Kovács and Recski, 2025)	65.9	75.0	70.2	90.5	86.7	88.5	64.0	55.9	59.7	73.5	72.5	72.8
ReDeEP$_{\text{LLaMA-2 7B Chat}}$(Sun et al., 2025)	45.3	46.1	45.7	79.3	74.8	77.0	48.4	29.4	36.6	57.7	50.1	53.1
ReDeEP$_{\text{LLaMA-2 13B Chat}}$(Sun et al., 2025)	31.0	75.0	43.9	93.6	94.9	94.2	25.0	78.8	38.0	49.9	82.9	58.7
ReDeEP$_{\text{LLaMA-3 8B Instruct}}$(Sun et al., 2025)	55.1	75.6	63.8	87.1	76.5	81.5	24.8	80.7	37.9	55.7	77.6	61.0
Direct Prediction Baseline$_{\text{Gemini-2.5-Flash}}$	30.1	97.5	45.9	76.6	95.5	85.0	38.0	93.1	54.0	48.2	95.4	61.6
TeQHallu (ours)$_{\text{Gemini-2.5-Flash}}$	42.8	92.5	**58.5**	86.8	90.0	**88.4**	55.5	84.3	**66.9**	61.7	88.9	**71.3**

Table 1: The response-level hallucination detection performance for each baseline method and different models, with overall macro averages over the different tasks from Niu et al. (2024). Song et al. (2024); Kovács and Recski (2025) are fine-tuned methods. ReDeEP results with the token-level method taken from Dubanowska et al. (2025). **Bold** indicates the best F1-score for unsupervised methods.

ModernBERT-based encoder (Fong et al., 2024) for efficient, span-level classification with a reduced parameter footprint. ReDeEP (Sun et al., 2025) decouples external context reliance from internal parametric knowledge by analyzing attention layers across various model scales.

Direct Prediction Baseline To isolate the impact of our symbolic reasoning layer, we evaluate a direct prediction baseline utilising Gemini 2.5 Flash. This baseline employs zero-shot prompting to classify responses as faithful or hallucinated based on the reference text while providing a natural language explanation. It serves as the neural foundation for our framework, representing the model's performance when relying solely on internal reasoning.

DiaHalu Baselines We evaluate several state-of-the-art LLMs and specialised detection frameworks using results directly from the DiaHalu publication (Chen et al., 2024b). Open-source baselines include LLaMa-30B (Touvron et al., 2023) and Vicuna-33B (Chiang et al., 2023), which is fine-tuned for human-like dialogue. Closed-source models comprise Gemini 1.5 Pro (Anil et al., 2023) and GPT-4 (OpenAI, 2023b), which provides a high-performance ceiling for complex logic. Specialised detection methods include SelfCheckGPT (Manakul et al., 2023) and FOCUS (Zhang et al., 2023), which detects hallucinations based on uncertainty. We incorporate the official DiaHalu results across both one-shot and Chain-of-Thought (CoT; Kojima et al., 2022) prompting configurations.

4.5 Results

Table 1 presents response-level hallucination detection performance on the RAGTruth benchmark across question answering, data-to-text, and summarisation tasks. Our proposed *TeQHallu* method significantly outperforms all other unsupervised baselines, including SelfCheckGPT and the zero-shot *Direct Prediction Baseline*, which consistently suffer from low precision and a tendency to over-predict hallucinations. Crucially, by utilising structured verification to eliminate false positives, TeQHallu also outpaces the supervisedly fine-tuned Llama-2-13B model and performs almost on par with heavily supervised state-of-the-art (SOTA) systems like LettuceDetect-large-v1.

While the fine-tuned RAG-HAT model delivers the peak macro F1-score and remains the only configuration to outpace our framework, TeQHallu is highly promising as it achieves this competitive performance entirely out-of-the-box. Ultimately, our approach proves that explicit symbolic reasoning can effectively compensate for the absence of parameter optimisation, delivering supervised-level accuracy without requiring any domain-specific fine-tuning.

4.6 Qualitative Analysis

Beyond quantitative results, TeQHallu provides interpretable insights through a traceable reasoning chain (Figure 3). We employ a multi-colour highlighting scheme where green identifies grounded reference facts, red denotes response hallucinations, and orange highlights the final grounded explanation. The core strength of our framework lies in a deterministic SQL reasoning step that transforms unstructured reference passages into a symbolic `food` table for structured verification. For the "gluten-containing foods" query, TeQHallu executes SQL logic to cross-reference

[INPUT SECTION]

Question: what food contains gluten?

Reference Passages:

Passage 1: Foods may also be labeled as gluten-free. If a product carries a gluten-free label, ...

Passage 2: Gluten-Containing Grains and Their Derivatives. Wheat (wheatberries, durum). Rye. Barley. Triticale. Malt (malted barley flour, malt extract, malt syrup). Brewer's Yeast.

Passage 3: Common Foods That Contain Gluten. Pastas: raviolis, dumplings. Noodles: ramen, udon. Breads and Pastries: croissants, pita.

Sampled Model Response:

Based on the provided passages, here are some foods that contain gluten:
1. Wheat: Wheatberries, Durum, Whole wheat flour, Bran, Germ.
2. Rye: Rye berries, Rye flour.
3. Barley: Barley flour, Malted barley flour, Barley sugar.
...
Note: It's important to note that even if a food is listed as containing gluten, it may still be safe for consumption for individuals with celiac disease...

[HALLUCINATION DETECTION OUTPUTS]

Direct LLM Prediction:

{ "hallucinations in response": true, "type": "Evident Introduction of Baseless Information", "explanation": "The 'Note' section at the end is not present in the reference passages." }

Representative SQL Reasoning:

Query the database for all items explicitly marked as containing gluten:

```
SELECT food_name, category FROM food WHERE contains_gluten = TRUE;
```

SQL Result (excerpt):

food_name	category
Wheatberries	Derivative (Wheat)
Durum	Derivative (Wheat)
Malted barley flour	Derivative (Malt)
Raviolis	Common Foods (Pasta)
Ramen	Common Foods (Noodles)

Interpretation: The food table contains entries derived strictly from the reference text. Items like 'Whole wheat flour' or 'Barley sugar' are missing from the TRUE results, indicating they are ungrounded hallucinations. The SQL-grounded explanation is much more detailed and incorporates this information.

TeQHallu (ours) Prediction:

{ "hallucinations in response": true, "type": "Evident Introduction of Baseless Information", "explanation": "The SQL results confirm only high-level grains and select derivatives. The response introduces unsupported specific items: Whole wheat flour, Bran, Germ, Rye berries, Rye flour, Barley flour, and Barley sugar. Additionally, no health safety 'Note' exists in the structured database." }

Figure 3: Full qualitative example from RAGTruth. Green indicates facts in the reference text; Red indicates hallucinations in the sampled response; Orange indicates the detailed SQL-grounded explanation.

claims against the database. This process reveals that whilst the response correctly identifies primary grains, it hallucinates specific derivatives like "Bran" and "Germ" which are absent from the source text. The neurosymbolic checkup then reconciles these SQL results with the initial neural judgement. In this instance, direct prediction flags the ungrounded health "Note" but overlooks the subtle grain-derivative hallucinations. TeQHallu's final explanation synthesises the symbolic evidence to provide a granular, high-fidelity reasoning trail. This structured grounding ensures each hallucination label is backed by explicit evidence, providing a transparent justification for human oversight.

SQL Accuracy Analysis To evaluate grounding fidelity, we manually audit a random sample of 35 reference documents per task. Database generation is correct only if it captures all source information without data loss. TeQHallu achieves its highest parsing accuracy in Data-to-text, where structured source data aligns naturally with relational schemas, as shown in Table 2. Performance drops

Task	Parsing Accuracy (%)
Question Answering	77.14
Data-to-text	91.43
Summarisation	82.86
Overall	84.76

Table 2: Manual evaluation of SQL parsing accuracy on 35 randomly sampled reference documents per task. Accuracy denotes the percentage of instances where the generated SQL queries fully capture all source text information without data loss.

Method	Precision	Recall	F1
Direct Prediction	48.2	95.4	61.6
Symbolic Prediction	39.9	51.3	39.7
SELECT Only IDs	62.4	80.1	69.7
TeQHallu	61.7	88.9	**71.3**

Table 3: Ablation study on the RAGTruth dataset showing overall macro-average precision, recall, and F1.

Model	Method	Precision	Recall	F1
Gemini-2.5-Flash (Spider EM: 51.3)	Direct Prediction	48.2	95.4	61.6
	TeQHallu	61.7	88.9	71.3
Qwen3 235B/22B (Spider EM: 29.0)	Direct Prediction	50.9	94.0	64.9
	TeQHallu	68.3	67.6	67.2
GPT-OSS 20B/3.6B (Spider EM: 22.4)	Direct Prediction	69.2	69.0	68.7
	TeQHallu	64.0	74.7	68.6
LLaMa 3.3 70B (Spider EM: 35.4)	Direct Prediction	78.4	54.7	64.3
	TeQHallu	76.3	57.4	65.4

Table 4: Model hallucination detection results on RAGTruth, reporting macro-average precision, recall, and F1-scores. Total/active parameters and Spider (Yu et al., 2018) multi-turn text-to-SQL exact match (EM) performance are provided in brackets.

slightly in Summarisation and reaches its lowest point in Question Answering due to the higher semantic variety of the reference texts. Crucially, these manual parsing metrics mirror our downstream hallucination detection F1-scores, following the exact same performance hierarchy. This consistency demonstrates that higher symbolic parsing fidelity directly translates into more robust hallucination detection.

4.7 Ablation Study

We conduct an ablation study to isolate the impact of different grounding configurations on TeQHallu's performance. We compare the baseline neural prediction against a strict intersection over symbolic entity ID sets, which forces deterministic verification but fails to capture the full versatility of hallucinations. The final symbolic prediction prompt generates SELECT queries for entity IDs to ensure claims in the generated response match the exact identifiers stored within the relational schema. While relying solely on neural text-to-text comparisons often misses fine-grained entity mismatches, a strict relational comparison proves too rigid. Instead, evaluating an alternative configuration that feeds these ID sets into the LLM prompt to inform its final decision yields the optimal balance. This soft LLM-driven alignment leverages the precision of symbolic identifiers without sacrificing the model's contextual reasoning capabilities.

4.8 Open-Model Results

We evaluate TeQHallu across open-weight models, including Qwen3 (Qwen-Team, 2025), GPT-OSS (OpenAI, 2025), and LLaMA 3.3 70B (Meta AI, 2025). We also evaluate text-to-SQL exact match performance of these models using the multi-turn setup established by Laban et al. (2025), from whom we also source results. Table 4 summarises macro-averaged performance on RAGTruth across direct prediction and TeQHallu.

Under identical prompting, direct prediction exhibits model-dependent trade-offs: Gemini-2.5-Flash and Qwen3 yield high recall but low precision, whereas LLaMA 3.3 70B prioritises precision. Integrating SQL grounding via TeQHallu consistently corrects this imbalance across all tested backbones by forcing explicit evidence checks. Except for GPT-OSS, which possesses the lowest text-to-SQL capability, this low-level symbolic approach yields the most stable performance and highest F1-scores. This confirms that symbolic grounding and neural judgements are highly complementary, effectively balancing precision and sensitivity.

While backbones with stronger text-to-SQL capabilities yield better overall detection performance, exceptionally high parsing metrics are not required. Overall, these results show that SQL-grounded detection provides a robust corrective signal regardless of the underlying model. Whilst smaller architectures like GPT-OSS gain less from the text-to-SQL layer—aligning with Vukovic et al. (2025)—structured evidence consistently enhances reliability across open and closed LLMs.

4.9 DiaHalu TOD Results

The response-level hallucination detection results on the task-oriented portion of DiaHalu are summarised in Table 5. Our proposed TeQHallu pipeline consistently achieves the highest F1-scores across all evaluated models, outperforming established baselines and prompt-engineering variants. Whilst baseline and CoT configurations exhibit high precision, they suffer from low recall, an imbalance our pipeline corrects. Crucially, TeQHallu maintains this strong performance when constructing the database directly from noisy conversational data, demonstrating robust capabilities on raw dialogues.

For Gemini-2.5-Flash, we also evaluate an alternative configuration (TeQHallu + MWOZ DB) by building a dataset-wide database from MultiWOZ first rather than constructing it incrementally per dialogue. However, this holistic approach degrades performance because cross-dialogue constraints introduce extra contextual information that makes the

Model	Variant	Precision	Recall	F1 Score
Random	–	31.86	48.00	38.30
SelfCheckGPT	BertScore	35.38	30.67	32.86
	NLI	38.84	62.67	47.96
	Prompt	48.00	32.00	38.40
FOCUS	–	34.09	60.00	43.48
LLaMA-30B	–	30.77	5.33	9.09
Vicuna-33B	–	42.86	4.00	7.32
Gemini-1.5 PRO	Base	60.00	36.00	45.00
	w/ CoT	69.77	40.00	50.85
	w/ One-shot	60.87	37.33	46.28
	TeQHallu *(ours)*	57.33	57.33	**57.33**
GPT-4	Base	74.19	30.67	43.40
	w/ CoT	73.17	40.00	51.72
	w/ One-shot	71.87	30.67	42.99
	TeQHallu *(ours)*	53.33	64.00	**58.18**
Gemini-2.5-Flash	Base	36.87	97.33	53.48
	TeQHallu *(ours)*	58.90	57.33	**58.11**
	TeQHallu + MWOZ DB	57.53	56.00	56.76

Table 5: Hallucination detection performance on the DiaHalu TOD portion. Baseline results from Chen et al. (2024b). **Bolding** indicates the best F1-score for each model group using our TeQHallu pipeline.

	NF	Ic	Ir	Ov	ALL
Gemini-1.5 PRO	27.63	42.67	23.53	30.77	31.39
TeQHallu *(ours)*	45.33	50.00	11.11	40.00	**42.38**
GPT-4	33.33	44.78	0.00	0.00	33.87
TeQHallu *(ours)*	40.00	45.00	0.00	40.00	**37.13**
Gemini-2.5-Flash	27.63	42.67	23.53	30.77	31.39
TeQHallu *(ours)*	48.65	47.83	10.53	40.00	**42.95**

Table 6: Fine-grained hallucination-type recognition F1-scores on the TOD portion. Micro average is reported over all types. **Bolding** indicates the best overall performance of each model.

verification logic overly strict.

Table 6 breaks down performance by specific hallucination subtypes, reporting aggregate performance as micro-F1. TeQHallu drastically improves the detection of Non-factual errors (NF) and substantially boosts Overreliance (Or) detection compared to baseline models. Incoherence (Ic) detection remains stable across configurations, whilst Irrelevance (Ir) detection exhibits a minor performance decrease. Despite this minor trade-off, our pipeline achieves the highest overall micro-F1 scores, validating its utility for grounded, conversational verification.

5 Discussion and Future Work

Evaluations on RAGTruth and DiaHalu demonstrate that grounding hallucination detection within structured SQL databases robustly complements direct neural prediction. The TeQHallu pipeline achieves competitive F1-scores against zero-shot baselines and fine-tuned models while enhancing interpretability through traceable SQL reasoning traces. Ablation results highlight that using SQL entity IDs directly for prediction creates a tighter symbolic coupling that is too strict to make the final hallucination prediction. Qualitative analysis confirms that this structured approach identifies subtle hallucinations, such as unsupported grain derivatives, which direct LLM predictions regularly overlook.

A key challenge remains the computational overhead and latency introduced by the multi-stage text-to-SQL generation process. However, this initial translation overhead yields a substantial downstream advantage by shifting the reasoning burden from fuzzy, autoregressive generation to precise database querying. This architectural uniformity enables unified cross-domain auditing without requiring task-specific fine-tuning.

Future work will explore integrating knowledge graphs alongside SQL grounding to broaden the scope of verifiable facts. We also plan to investigate iterative database refinement using detection feedback loops to dynamically improve grounding fidelity over time. Furthermore, improving performance in predictions based on a a strict neurosymbolic check-up could lead to better interpretability. Finally, conducting human-in-the-loop evaluations will be essential to validate the utility of TeQHallu's symbolic explanations in high-stakes legal and financial applications.

6 Conclusion

We introduce TeQHallu, a neurosymbolic framework that enhances hallucination detection by grounding language model responses in structured SQL databases. By transforming reference documents into verifiable evidence, our approach matches or exceeds state-of-the-art performance on the RAGTruth and DiaHalu TOD benchmarks.

Beyond quantitative metrics, TeQHallu provides transparent reasoning traces and human-readable explanations via symbolic SQL checkups. This structured grounding identifies subtle factual inconsistencies that direct neural predictions often overlook, fostering greater interpretability and trust.

Our findings demonstrate the substantial potential of integrating symbolic reasoning with generative models to ensure factuality. Future work will focus on expanding the framework to incorporate diverse structured sources and iterative database refinement for high-stakes, knowledge-intensive applications.

Limitations

While our pipeline demonstrates strong performance, several limitations exist.

First, effectiveness depends on the quality and completeness of reference documents. Incomplete or noisy material may cause the SQL-grounded reasoning to miss hallucinations or produce incorrect explanations, although the results on DiaHalu are promising.

Second, constructing and querying the SQL database introduces computational overhead compared to direct prediction. This may limit applicability in real-time or resource-constrained settings, particularly for large-scale dialogue systems. Furthermore, there can be an information loss when translating unstructured data to SQL, as shown in our manual analysis.

Third, the pipeline relies on the LLM's ability to generate accurate SQL queries. Errors in SQL generation can propagate to the final prediction, potentially undermining interpretability. Furthermore, performance can vary based on exact prompt wording. Vukovic et al. (2025) show that the order in which the documents are presented do not significantly impact the quality of the final ontology.

Finally, while SQL grounding improves explainability, we have not yet conducted large-scale human evaluations. Such assessments are critical to validate how end-users perceive and utilise these transparency benefits, but are out of scope for this paper.

Addressing these limitations—including exploring hybrid knowledge sources, graded scoring, and human-in-the-loop evaluation—remains a priority for future research.

Ethical Considerations

While our framework enhances the reliability of hallucination detection, its deployment involves several ethical considerations. The pipeline relies on reference documents; if these contain biased, harmful, or non-representative information, the generated SQL database and subsequent hallucination checks will reflect those same biases. Automated detection may also provide a false sense of security; users might over-rely on the system's "verified" status, leading to automation bias in critical domains like healthcare or law. Furthermore, the computational cost associated with multi-step SQL generation increases the carbon footprint of the inference process, necessitating a balance between detection accuracy and environmental impact.

Finally, transparent reasoning traces could be exploited by malicious actors to reverse-engineer sensitive reference data or to develop adversarial responses that bypass the structured verification logic. Addressing these risks requires continuous human oversight and robust data governance to ensure that automated fact-checking remains a beneficial tool for digital safety.

References

2021. Yelp open dataset. https://business.yelp.com/data/resources/open-dataset/. Accessed: 2026-02-20.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Maria Schilder, Dzmitry Bahdanau, Phaneendra Bansal, Conor Curtis, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. Ms marco: A human generated machine reading comprehension dataset. In *InCoCo at NeurIPS Workshop on Machine Reading for Question Answering*.

Tom Brown, Benjamin Mann, Nick Ryder, and 1 others. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024a. Inside: Llms' internal states retain the power of hallucination detection. In *The 12th International Conference on Learning Representations (ICLR)*.

Kedi Chen, Qin Chen, Jie Zhou, He Yishen, and Liang He. 2024b. DiaHalu: A dialogue-level hallucination evaluation benchmark for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9057–9079, Miami, Florida, USA. Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, and 1 others. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://lmsys.org/blog/2023-03-30-vicuna/*.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. LM vs LM: Detecting factual errors via cross examination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language*

Processing, pages 12621–12640, Singapore. Association for Computational Linguistics.

Daniel C. Dennett. 1987. *The Intentional Stance*. MIT Press, Cambridge, MA.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Zuzanna Dubanowska, Maciej Żelaszczyk, Michał Brzozowski, Paolo Mandica, and Michal P. Karpowicz. 2025. Representation-based broad hallucination detectors fail to generalize out of distribution. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 17563–17575, Suzhou, China. Association for Computational Linguistics.

Artur S. d'Avila Garcez and Luis C. Lamb. 2023. Neurosymbolic ai: the 3rd wave. *Artificial Intelligence Review*, 56(11):12387–12406.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 422–428, Marseille, France. European Language Resources Association.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Benjamin Fong, Somya Jain, Christopher Eliasmith, and 1 others. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

Dawei Gao, Haibin Wang, Yaliang Li, Xiuyu Sun, Yichen Qian, Bolin Ding, and Jingren Zhou. 2024. Text-to-sql empowered by large language models: A benchmark evaluation. *Proc. VLDB Endow.*, 17(5):1132–1145.

Artur d'Avila Garcez, Luis C. Lamb, and Dov M. Gabbay. 2019. Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *Journal of Applied Logic*, 33:1–17.

Google. 2025. Gemini 2.5 flash model reference. `https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash`. Accessed: 2026-02-13.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.*, 43(2).

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive Mixtures of Local Experts. *Neural Computation*, 3(1):79–87.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.*, 55(12).

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Ádám Kovács and Gábor Recski. 2025. LettuceDetect: A hallucination detection framework for rag applications. *arXiv preprint arXiv:2502.17125*.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. 2025. LLMs get lost in Multi-turn Conversation. *arXiv preprint arXiv:2505.06120*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Jinxin Li, Gang Tu, ShengYu Cheng, Junjie Hu, Jinting Wang, Rui Chen, Zhilong Zhou, and Dongbo Shan. 2025. LLM Hallucination Detection: A Fast Fourier Transform Method Based on Hidden Layer Temporal Signals. *arXiv preprint arXiv:2509.13154*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring How Models Mimic Human Falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Xiaoqing Liu, Shuaichen Shen, Bo Li, Peng Ma, Rui Jiang, Yuxuan Zhang, Jing Fan, Guoliang Li, Nan Tang, and Ying Luo. 2025. A survey of text-to-sql in the era of llms: Where are we, and where are we going? *IEEE Transactions on Knowledge and Data Engineering*, 37(10):5735–5754.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Meta AI. 2025. Llama 3.3 model card and prompt formats. `https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3/`. Accessed: 2026-02-19.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI. 2023a. Gpt-3.5 turbo fine-tuning and api updates. https://openai.com/index/gpt-3-5-turbo-fine-tuning-and-api-updates/. OpenAI Blog.

OpenAI. 2023b. Gpt-4 technical report. https://arxiv.org/abs/2303.08774. ArXiv:2303.08774.

OpenAI. 2025. Introducing GPT-OSS. Blog post on OpenAI website. Accessed 2025-10-30.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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 *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Mohammadreza Pourreza and Davood Rafiei. 2023. DIN-SQL: Decomposed In-Context Learning of Text-to-SQL with Self-Correction. In *Advances in Neural Information Processing Systems*, NeurIPS 2023. Conference paper.

Henrijs Princis, Cristina David, and Alan Mycroft. 2025. Enhancing sql query generation with neurosymbolic reasoning. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, volume 39 of *AAAI Technical Track*, pages 19959–19968. AAAI Press.

Qwen-Team. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations (ICLR) 2017*. Poster presentation.

Juntong Song, Xingguang Wang, Juno Zhu, Yuanhao Wu, Xuxin Cheng, Randy Zhong, and Cheng Niu. 2024. RAG-HAT: A hallucination-aware tuning pipeline for LLM in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1548–1558, Miami, Florida, US. Association for Computational Linguistics.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. 2024. LLM-check: Investigating detection of hallucinations in large language models. In *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024)*. Poster Presentation.

Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, Yang Song, and Han Li. 2025. Redeep: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*. Spotlight.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Renato Vukovic, Carel van Niekerk, Michael Heck, Benjamin Ruppik, Hsien-Chin Lin, Shutong Feng, Nurul Lubis, and Milica Gasic. 2025. Text-to-sql task-oriented dialogue ontology construction. *arXiv preprint arXiv:2507.23358*.

Laura Weidinger, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, Myra Cheng, Borja Balle, Atoosa Kasirzadeh, Courtney Biles, Sasha Brown, Zac Kenton, Will Hawkins, Tom Stepleton, Abeba Birhane, Lisa Anne Hendricks, Laura Rimell, William Isaac, and 4 others. 2022. Taxonomy of risks posed by language models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 214–229, New York, NY, USA. Association for Computing Machinery.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena 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*, pages 4602–4625, Seattle, United States. Association for Computational Linguistics.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 404–413, Metz, France. Association for Computational Linguistics.

Chuanpeng Yang, Yao Zhu, Wang Lu, Yidong Wang, Qian Chen, Chenlong Gao, Bingjie Yan, and Yiqiang Chen. 2025. Survey on knowledge distillation for large language models: Methods, evaluation, and application. *ACM Trans. Intell. Syst. Technol.*, 16(6).

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921, Brussels, Belgium. Association for Computational Linguistics.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 915–932, Singapore. Association for Computational Linguistics.

A Full TeQHallu Prompt

See Figure 4 for the full TeQHallu Prompt

1. **Step 1 – Check existing tables in DB**
   You are given a reference. Generate only SQL queries (valid for the Python sqlite3 package) to check whether the information from the reference is already present in the database. Here is the current list of tables in the DB: {db_result_input}. First, only generate the `PRAGMA(table_info)` queries for the relevant tables. Make sure to highlight SQL using “‘`sql` “‘. Here is the reference: [Reference Text]

2. **Step 2 – Check existing info in DB**
   Query results from queries above: {db_result_input}. Generate only SQL queries (valid for the Python sqlite3 package) to check whether the information from the reference is already present in the database. Do this based on the current set of tables and their corresponding columns seen in the `PRAGMA` query results above. Generate `SELECT` queries now. Make sure to highlight SQL using “‘`sql` “‘.

3. **Step 3 – Insert missing info into DB**
   Query results from queries above: {db_result_input}. Based on the reference and the results of Step 2, generate only SQL queries to insert the missing information into the database. Do not duplicate existing data. If necessary create tables that are not yet in the DB before adding information. Make sure to highlight SQL using “‘`sql` “‘.

4. **Step 4 – Query info for response**
   You are given a sampled response. Generate only SQL queries to retrieve all relevant information from the database that would be necessary to fact-check this response. Make sure to highlight SQL using “‘`sql` “‘. Here is the sampled response: [Response Text]

5. **Step 5 – Detect hallucinations**
   Query results from queries above: {db_result_input}.

   Additionally to the context from the steps above you get:

   - An initial hallucination prediction, which is a preliminary judgement of whether the response contains hallucinations based on the reference and the response only (this prediction does not use any DB-retrieved information).

   Your task is to make a final hallucination determination for the response. A hallucination is any information that contradicts or goes beyond what is supported by the reference. Hallucinations fall into one of the following categories:

   1. Evident Conflict – Direct contradiction or opposition to the reference (e.g., factual errors, incorrect numbers, misspelled names).

   2. Subtle Conflict – Divergence from the intended contextual meaning (e.g., substitutions or rephrasings that change implications or severity).

   3. Evident Introduction of Baseless Information – Clear addition of fabricated or unsupported details not found in the reference.

   4. Subtle Introduction of Baseless Information – Inferred details, assumptions, sentiments, or unverifiable insights beyond what the reference states.

   Evaluation Process:

   1. Consider the initial hallucination prediction as a starting signal.

   2. Independently verify the response against: the original reference (primary authority) and the DB-retrieved information (used to confirm or refute factual claims).

   3. Assess whether the initial prediction remains accurate after considering the DB information. If the initial prediction is correct, retain it. If the initial prediction is inaccurate or incomplete, update or override it.

   4. The final decision must be based on your full evaluation, not solely on the initial prediction. Important Guidelines:

   - Focus only on information grounded in the reference.

   - Do not mark minor wording differences, optional phrases, or harmless stylistic changes as hallucinations.

   - Be lenient about paraphrasing and small changes in phrasing.

   - Only flag information as a hallucination if it clearly introduces a fact, claim, or detail that is unsupported by or contradicts the reference.

   Output Format:

   Return your final answer in exactly one of the following JSON formats:

   `{"hallucinations in response": True, "type": "<category>", "explanation": "<explanation>"}` or `{"hallucinations in response": False}`.

Figure 4: Full prompts for the text-to-SQL hallucination detection pipeline. `db_result_input` contains outputs from preceding SQL executions.