Evaluating Variance and Reliability of Semantic Operators Under Ambiguous Queries in LOTUS

Geetansh Kumar
Department of Computer Science and Engineering
University at Buffalo, SUNY
geetansh@buffalo.edu

***Abstract*—Large Language Models (LLMs) are increasingly being used as semantic operators for executing natural language queries over unstructured data. Frameworks such as LOTUS enable users to express complex analytical tasks in natural language, allowing LLMs to interpret and execute semantic operations directly over datasets and documents. While this provides flexibility and accessibility beyond traditional query systems, it also introduces challenges related to ambiguity, reasoning consistency, and non-deterministic behavior.**

In this work, we evaluate the reliability and variance of LLM-powered semantic operators implemented through LOTUS under ambiguous and unambiguous query settings. We construct a SQL-based gold standard dataset by combining the latest IMDb dataset with the research-backed JOB benchmark using fuzzy matching techniques. In addition, we create an equivalent document-based dataset by scraping and processing IMDb HTML pages to simulate unstructured retrieval and reasoning tasks. Our experiments analyze output variance across multiple runs, accuracy relative to structured SQL ground truth, and differences in reasoning behavior between ambiguous and unambiguous queries. Our findings show that ambiguity significantly increases output variance and decreases reliability, while large document settings introduce additional instability due to reasoning complexity.

I. Introduction

LARGE Language Models (LLMs) are increasingly being integrated into semantic query processing systems. Frameworks such as LOTUS, allow LLMs to be invoked as semantic predicates or projection functions inside relational query pipelines. Instead of relying purely on symbolic filtering conditions, these systems execute semantic reasoning functions over rows.

In our study, we focus on evaluating the behavior of semantic filter and join operators in LOTUS [1] over unstructured IMDb-based documents represented as dataframe-backed document chunks.

To illustrate the workflow, consider the following SQL query from the JOB [2] benchmark:

```
SELECT DISTINCT t.*
FROM temp_tables.small_title t
JOIN temp_tables.small_title_ratings tr
  ON t.tconst = tr.tconst
JOIN temp_tables.title_mapping m
  ON t.tconst = m.tconst
JOIN temp_tables.small_movie_companies mc
  ON m.job_movie_id = mc.movie_id
JOIN temp_tables.small_company_name cn
  ON mc.company_id = cn.id
WHERE cn.country_code = '[us]'
  AND (t.genres LIKE '%Documentary%'
       OR t.genres LIKE '%Horror%')
  AND tr.averageRating > 5.0
  AND CAST(t.startYear AS INT)
  BETWEEN 2000 AND 2020;
```

The equivalent natural language semantic filter used in LOTUS is: *"{text_cleaned} is about a title made by a U.S. company, Its genre must be either Documentary or Horror. Secondary genres are allowed, but Documentary or Horror must be there as well. Do not accept titles where Documentary/Horror is only implied by tone, themes, or loose description. It should have an average rating above 5. It should also have been released between 2000 through 2020."*

LOTUS executes this workflow by using HTML-based IMDb documents and invoking semantic filter or join operators over dataframe-backed representations. The LLM evaluates whether each document chunk satisfies the semantic predicate and returns matching rows together with reasoning explanations.

This setup introduces ambiguity because semantic interpretation may vary across executions. Terms such as genre relevance, implied themes, or semantic inclusion criteria can lead to different filtering decisions even when the underlying query intent remains fixed.

Unlike traditional deterministic database systems, semantic operator execution may therefore exhibit:

- Run-to-run output variance
- Differences in reasoning behavior
- Non-deterministic semantic filtering

Our work evaluates these behaviors systematically by comparing LOTUS outputs against deterministic SQL-based ground truth derived from the JOB benchmark.

II. Problem Formulation

This work investigates the reliability of LLM-powered semantic operators under ambiguous and unambiguous query settings.

Our primary research questions are:

1) How does ambiguity affect the determinism of semantic operator outputs?
2) How accurately do LOTUS semantic filters reproduce SQL-based ground truth results?
3) Do semantic operators exhibit symmetric probabilistic behavior, or are errors concentrated toward false positives and overestimation?

We evaluate these questions using semantic filter and join operators implemented in LOTUS over unstructured IMDb-based HTML documents.

The study focuses on three primary dimensions:

- Variance across repeated executions
- Accuracy relative to structured SQL ground truth
- Semantic reasoning consistency produced by LOTUS explanations

We hypothesize that ambiguity increases semantic variance and produces an asymmetric error behavior dominated by false positives rather than false negatives.

III. Dataset Construction

Our dataset curation process consists of two aligned representations of IMDb-derived data to support evaluation of semantic operators in LOTUS.

First, we construct a structured relational dataset by combining the IMDb dataset [3] with the JOB benchmark schema. This creates a SQL-executable ground truth where JOB queries are run on an IMDb-enriched relational database.

Second, we construct an unstructured dataset by scraping IMDb pages and converting them into HTML documents containing relevant metadata such as titles, genres, cast, ratings, and production details. These documents are then filtered and chunked to form a queryable corpus for semantic processing.

Finally, both representations are aligned through a shared query design process, where natural language queries are constructed to be semantically equivalent to JOB SQL queries, enabling direct comparison between structured ground truth and LOTUS-based execution outputs.

A. Structured Dataset

To construct a reliable evaluation benchmark, we combine two complementary sources: the latest IMDb dataset and the JOB benchmark dataset.

The IMDb dataset provides up-to-date and detailed metadata including movie titles, cast information, ratings, genres, and production details that closely reflect real-world IMDb records. In contrast, the JOB benchmark provides a well-established relational schema along with rich information about production companies, distribution entities, and structured query patterns widely used in database research.

We combine these datasets to leverage their complementary strengths: IMDb contributes freshness and alignment with real-world data distributions, while JOB provides structural completeness and relational richness, particularly in modeling production and distribution relationships that are not fully captured in IMDb alone. Together, they form a more comprehensive relational dataset suitable for evaluating semantic query execution.

The resulting relational dataset serves as the SQL-based ground truth. JOB benchmark queries are executed over the JOB schema enriched with IMDb metadata to produce deterministic baseline outputs.

```sql
SELECT DISTINCT t.*
FROM temp_tables.small_title t
JOIN temp_tables.small_title_ratings tr
    ON t.tconst = tr.tconst
JOIN temp_tables.title_mapping m
    ON t.tconst = m.tconst
JOIN temp_tables.small_movie_companies mc
    ON m.job_movie_id = mc.movie_id
JOIN temp_tables.small_company_name cn
    ON mc.company_id = cn.id
WHERE cn.country_code = '[us]'
  AND (t.genres LIKE '%Documentary%' OR
       t.genres LIKE '%Horror%')
  AND tr.averageRating > 5.0
  AND CAST(t.startYear AS INT) BETWEEN 2000 AND 2020;
```

	tconst	titleType	primaryTitle	originalTitle
1	tt1363109	tvMovie	Journey to the Edge of the Universe	Journey to the Edge of the Universe
2	tt0810412	short	The Legend of Flashpants	The Legend of Flashpants
3	tt0497557	short	Don't Whistle	Don't Whistle
4	tt0443488	short	Dream on Silly Dreamer	Dream on Silly Dreamer

Fig. 1. SQL Query

B. Mapping Strategy

Exact entity matching across heterogeneous datasets is challenging due to variations in naming conventions, and duplicate or ambiguous titles. In our setting, relying on strict equality would lead to incorrect joins or loss of valid records, especially since multiple movies may share similar titles or exist as remakes, re-releases, or differently formatted entries across datasets.

We align IMDb titles T with JOB titles J by a predicate. For $t \in T$ and $j \in J$,

$$C_{title}(t, j) \equiv \text{lower}(t.primaryTitle) = \text{lower}(j.title)$$
$$C_{year}(t, j) \equiv t.startYear = \text{cast}(j.production\_year, varchar)$$
$$\text{Match}(t, j) \equiv C_{title}(t, j) \wedge C_{year}(t, j)$$

The mapping table is $\{(t, j) \mid \text{Match}(t, j)\}$. Unmatched rows are excluded.

C. Unstructured dataset and HTML processing

To evaluate LOTUS in document-oriented environments, we create an unstructured HTML-based corpus which is similar to structured relational data in terms of information present.

We scrape IMDb pages corresponding to movie titles, cast information, production companies, ratings, and related metadata. The scraped HTML is filtered to retain only semantically relevant information before being used for LOTUS-based querying. The resulting documents simulate realistic web-based information retrieval environments.

Raw HTML contains significant amounts of irrelevant information that increases context size and cost. Therefore, we preprocess and filter the HTML content before experimentation.

Our preprocessing pipeline includes:

- Removing unnecessary scripts, styling, and images — to reduce noise and minimize irrelevant tokens in the document representation.
- Extracting semantically relevant sections such as text, URLs, cast, company, and other metadata — to retain only information useful for semantic querying and operator evaluation.

Content retained (semantically relevant):

- Visible body text (title, plot, genres, ratings, runtime, country, production company, etc.)
- Structured pairs extracted from `<dt>/<dd>` and short `label:` siblings
- Attribute values from `aria-label`, `title`, `alt`, `content`, and `data-*` fields
- Inner text from nodes marked with `data-testid`, `role`, or other `data-*` attributes
- Canonical IMDb URLs and extracted `tt...`, `nm...`, `co...` identifiers (join keys for semantic operators)

Content removed or stripped:

- `<script>` and `<style>` tags
- Advertisement containers (e.g., `cornerstone_slot`, `nas-slot`, `placeholder_pattern`) in strict HTML mode
- Wayback Machine chrome (`wm-ipp`, `donato`, etc.) in HTML mode
- Inline `style` attributes and JavaScript event handlers
- Images, `<link>`, and `<meta>` tags in strict HTML mode (text mode retains more markup for recall)

In text mode we intentionally retain header, footer, and navigation text when it can carry facts used in predicates (e.g., country tags such as `[US]`).

This optimization step is critical because large documents substantially increase LLM context usage, latency, and API cost. We ensure that the processed HTML corpus retains the same semantic information present in the relational dataset to maintain evaluation consistency.

IV. Experimental Setup

A. Overview

Our experimental setup is designed to compare semantic query execution using the *LOTUS framework* against deterministic SQL-based execution over a structured ground truth dataset. Specifically, we evaluate how natural language queries, when executed through semantic filter and join operators in LOTUS, behave relative to equivalent SQL queries executed on a relational IMDb-enriched JOB schema.

The implementation, curated datasets, semantic queries, and experiment scripts are available at: https://github.com/UB-ADBLAB/AmbiguityInSemQP

The evaluation pipeline consists of two parallel execution paths: (1) structured query execution using SQL over the relational dataset to produce ground truth outputs, and (2) unstructured query execution using LOTUS over HTML-based IMDb documents using semantic operators. The outputs from both systems are then compared under multiple evaluation criteria including accuracy, variance, and reasoning consistency.

All the queries we tested are SPJ (Select-Join-Project) queries, so that precision and recall are well defined.

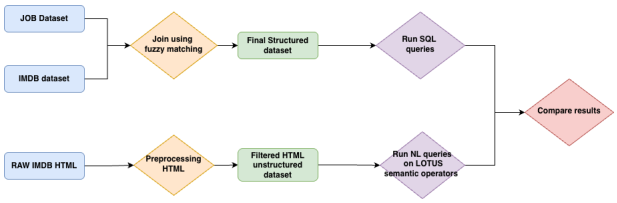

Fig. 2. Flow Diagram

B. Framework

All experiments are conducted using the *LOTUS framework*, which supports semantic filter and join operators that invoke LLMs as predicates over unstructured document chunks.

In our setup, LOTUS operates on an HTML-based IMDb document corpus, where each document is processed in chunks and evaluated using semantic predicates. In parallel, JOB benchmark SQL queries are executed on a structured relational dataset enriched with IMDb metadata, producing deterministic ground truth results for evaluation.

```
PROMPT_UNIFIED_GENRE_12A = (
    """
{text_cleaned} is about a title made by a U.S. company,
and title should capture real-life events or feels like the kind of movie that keeps me up at night.
It should have an average rating above 5.
It should also have been released between 2000 through 2020.
"""
)
```

	id	text_cleaned	explanation_filter	raw_output_filter
0	tt0427969	Cast & crew\nUser reviews\nTrivia\nFAQ\nIMDbPr...	- Production: The page lists U.S. production(d...	Reasoning:\n- Production: The page lists U.S. ...
1	tt0443488	Cast & crew\nUser reviews\nIMDbPro\nAll topics...	- The page describes "Dream on Silly Dreamer" ...	Reasoning:\n- The page describes "Dream on Sil...
2	tt0482572	Cast & crew\nUser reviews\nTrivia\nFAQ\nIMDbPr...	- The page is for Pride and Glory (title shown...	Reasoning:\n- The page is for Pride and Glory ...
3	tt0497557	Cast & crew\nIMDbPro\nAll topics\nDon't Whist...	- The page is for the title "Don't Whistle" (r...	Reasoning:\n- The page is for the title "Don't...
4	tt0810412	Cast & crew\nIMDbPro\nAll topics\nThe Legend o...	- The page lists Country of origin: United Sta...	Reasoning:\n- The page lists Country of origin...
5	tt1363109	Cast & crew\nUser reviews\nTrivia\nIMDbPro\nAl...	- Title: "Journey to the Edge of the Universe"...	Reasoning:\n- Title: "Journey to the Edge of t...

Fig. 3. Semantic Query

C. Evaluation Metrics

We evaluate system behavior along three primary dimensions:

Precision and Recall: We compare LOTUS outputs against deterministic SQL outputs generated from the JOB benchmark queries executed on the structured relational dataset.

Precision and recall are computed as:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

where TP denotes true positives, FP denotes false positives, and FN denotes false negatives.

Reasoning Consistency: In addition to quantitative retrieval metrics, we manually inspect reasoning traces generated by LOTUS semantic operators to analyze how interpretation changes between ambiguous and unambiguous queries.

V. Results

A. Quantitative Results

We evaluate semantic retrieval quality using precision and recall against the SQL-based ground truth.

Precision and recall are computed using:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

The gold standard contains 4 expected rows for each query. Across all evaluated queries, we observe:

$$FN = 0$$

The resulting query-level metrics are shown below:

Query	TP	FP	Precision	Recall	F1 Score
Q1	4	0	1.000	1.000	1.000
Q2	4	2	0.667	1.000	0.800
Q3	4	3	0.571	1.000	0.727
Q4	4	1	0.800	1.000	0.889

TABLE I
Precision, Recall, and F1-score across semantic queries

Average precision:

$$\frac{1.0 + 0.667 + 0.571 + 0.800}{4} \approx 0.760$$

Average recall:

$$1.0$$

Average F1-score:

$$\frac{1.0 + 0.800 + 0.727 + 0.889}{4} \approx 0.854$$

B. Analysis

The precision and recall analysis further reveals that LOTUS rarely misses relevant results, as indicated by consistently perfect recall values. However, ambiguity introduces additional false positives, reducing precision despite retaining correct matches.

This behavior suggests that semantic operators exhibit an asymmetric error profile dominated by overestimation rather than underestimation. Retrieved outputs are frequently equal to or larger than the SQL ground truth, while false negatives remain rare.

We also observe that outputs are concentrated around a subset of dominant semantic interpretations rather than behaving like a probabilistic distribution centered around the true answer. This indicates that semantic variance in LOTUS is not purely random, but instead biased toward semantically broader behavior.

Our overall findings indicate:

- LOTUS performs reliably for well-defined queries. Ambiguity significantly increases variance and inconsistency.
- Large unstructured documents introduce additional instability.
- Semantic outputs exhibit non-standard statistical behavior.

VI. Discussion and Conclusion

The results demonstrate that LLM-powered semantic operators behave fundamentally differently from traditional deterministic database systems. While LOTUS enables highly flexible natural language querying over unstructured data, ambiguity propagates directly into semantic execution behavior. This creates challenges for reproducibility, evaluation, and trustworthiness.

Our experiments show that ambiguity significantly impacts consistency and accuracy, while large document settings introduce additional context-related instability. We also observe that semantic outputs do not behave like a standard probabilistic distribution centered around the true answer. Instead, outputs are concentrated around dominant interpretations and broader semantic matches.

A major observation from our evaluation is the prevalence of false positives and overestimation behavior. Since underestimation occurs less frequently, the resulting error profile behaves more like a one-sided error estimation problem rather than symmetric statistical variance.

Overall, LOTUS demonstrates strong potential for flexible semantic querying, particularly for well-defined and specific natural language queries. However, reliability challenges remain in ambiguous and large-scale document environments. Future work may explore:

- Improved retrieval optimization for large documents
- Deterministic semantic reasoning controls

These findings provide insight into the limitations and future directions of LLM-powered semantic operator systems.

References

[1] "Semantic operators: A declarative model for rich, ai-based data processing. lotus framework documentation: https://lotus-data.github.io/."
[2] "Join order benchmark (job): https://github.com/gregrahn/join-order-benchmark."
[3] "Imdb official dataset: https://datasets.imdbws.com/."