BD-AgriKG: A Knowledge Graph and Bilingual Text2Cypher Benchmark for Crop Suitability and Productivity in Bangladesh

Md. Aukerul Moin Shuvo **Md. Nazrul Islam Mondal**
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology, Rajshahi-6204, Bangladesh
2204103003@student.ruet.ac.bd, mondal@cse.ruet.ac.bd

Abstract

Agricultural advisory in Bangladesh rests on public statistics locked in PDF yearbooks, and on language technology that rarely speaks Bangla. We present BD-AgriKG, a knowledge graph that integrates five government sources on one administrative spine keyed by standard place codes: upazila-level land suitability for 16 crops, agro-meteorological thresholds, crop calendars, soil associations, and a newly extracted district production panel covering twelve crop years, validated against the yearbooks' own published totals. On top of the graph we release what is, to our knowledge, the first bilingual (English and Bangla) execution-based Text2Cypher benchmark: 460 questions instantiating 26 templates, built Cypher-first so that every item carries an executable gold query and a non-empty executed answer. We report execution-accuracy baselines for seven large language models (LLMs) in both languages. The best model reaches 95.2% in English and 90.9% in Bangla under a strict exact-projection metric whose frontier failures are almost all extra returned columns. Every model scores lower in Bangla, by 1 to 14 points; after multiple-comparison correction the gap is significant for three of the seven models under a template-clustered bootstrap and for one under template-level tests, and for four models, both frontier models among them, two question templates carry most of it. One turn of execution feedback, the simplest use of the graph's own execution signal, recovers 43–73% of the invalid and empty failures of mid-tier models and helps Bangla at least as much as English, though not significantly more, while small open models recover almost nothing. The graph, panel, benchmark and harness are released.

1 Introduction

Bangladesh's agricultural extension workers answer district-level what, where and when questions

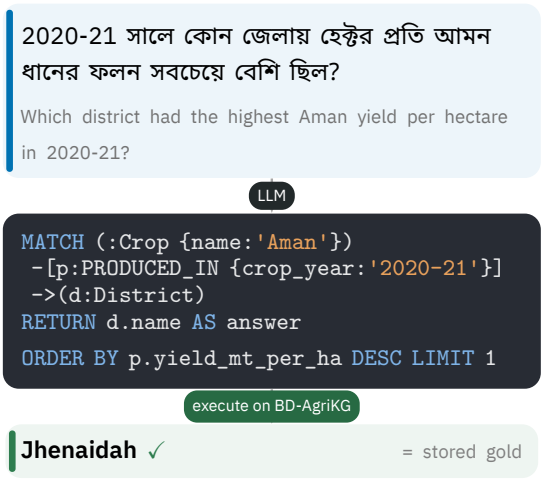


Figure 1: A released benchmark item end to end: the Bangla question and its English twin, a model-generated Cypher query, and read-only execution against the graph, graded against the stored gold result.

every day: which upazila suits a crop, when to sow, what a district produced last year, which pest threatens the main crop. The answers exist in public statistics, land-suitability zoning, crop calendars and soil surveys, but they sit in PDF yearbooks and scattered agency portals rather than in any queryable form. Large language models (LLMs) answer such questions fluently and, on exactly the facts that matter (planting dates, seed rates, input doses), often wrongly (Ibrahim et al., 2024); a recent Bangla agricultural benchmark finds that even retrieval-augmented systems keep a residual hallucination floor on safety-critical values (Reza et al., 2026). Grounding generation in a structured knowledge graph (KG) is the obvious remedy, but no Text2Cypher or KG question answering (KGQA) resource exists in Bangla at all, and the existing Text2Cypher benchmarks are synthetic and English-only (Ozsoy et al., 2025; Ozsoy and Tai, 2025). The measurement matters beyond one-shot question answering (QA). In agentic systems the graph serves as a verifiable tool environment

Proceedings of the 3rd Workshop on KNOwledge GRaphs & Agentic Systems Interplay (NORA 2026), pages 63–76
November 10, 2026 ©2026 Association for Computational Linguistics

in which agents interleave queries with reasoning (Shen et al., 2025; Yang et al., 2024), so the question-to-query step we isolate is the primitive every such action rests on, and execution feedback is a signal the agent itself can observe. Figure 1 shows one released benchmark item end to end.

We make four contributions: (1) BD-AgriKG, a P-code-keyed KG over five public agricultural sources including a 12-year district production panel extracted from PDF yearbooks with checksum validation; (2) the first bilingual execution-based Text2Cypher benchmark (460 items, Cypher-first); (3) execution-accuracy baselines in English and Bangla, an analysis of the Bangla gap, and a one-turn execution-feedback repair experiment; (4) full release of data, extraction pipeline and harness.

2 Related work

Agricultural knowledge graphs. Agricultural KGs are usually extracted from text (Chen et al., 2019). The closest resource to ours is BDAKG (Deb Nath et al., 2024), a Resource Description Framework (RDF) graph of Bangladesh Bureau of Statistics (BBS) data built for online analytical processing (OLAP); it carries production statistics but no land suitability, crop calendar, agrometeorological or pest data, and it has no natural-language layer. BD-AgriKG differs in both respects: it integrates five sources on one administrative spine and is built to be queried in natural language.

GraphRAG in agriculture. Recent graph-based retrieval-augmented generation (GraphRAG) systems pair Neo4j graphs with LLMs for agricultural QA: Crop GraphRAG for pests and diseases (Wu et al., 2025) and a garlic-cultivation assistant (Zhou et al., 2026), both building their graphs by LLM extraction from text and evaluating end-answer accuracy, in Chinese. ShizishanGPT couples an agri LLM with a KG module and tools (Yang et al., 2024). The general GraphRAG line (Edge et al., 2024) likewise assumes an LLM-extracted graph. Our graph is curated from statistical tables, and our evaluation isolates the query-translation step with execution accuracy.

Bangla agricultural NLP. Bangla advisory systems retrieve from documents rather than a graph: KrishokBondhu answers voice queries over scanned handbooks digitised by optical character recognition (OCR) (Ameen et al., 2025), and Reza et al. (2026) release a provenance-traceable Bangla benchmark showing residual chemical-dosage hallucination even with oracle evidence. General-purpose Bangla resources exist for exam QA (Shafayat et al., 2024) and automatic KG construction (Wasi et al., 2024), but none grounds answers in a structured graph. The closest resource is concurrent work: IndicDB (Dawar et al., 2026) evaluates execution-based Text2SQL over 20 relational databases of Indian open data in seven language variants including Bengali, reporting Bengali at 56.2% against 64.7% for English. That 8.5-point Bengali penalty on relational schemas is of the same order as the gaps we measure on a property graph. No Bangla Text2Cypher resource exists.

Text2Cypher. The Neo4j Text2Cypher dataset (Ozsoy et al., 2025) and its multilingual extension (English, Spanish, Turkish; no Bangla) (Ozsoy and Tai, 2025) train and evaluate mostly on synthetic schemas, with string-similarity metrics and, on the subset with live database access, execution-based exact match; CypherBench (Feng et al., 2024) argues for execution-based evaluation on full-scale graphs, which we adopt, following the Text2SQL practice of Spider and BIRD (Yu et al., 2018; Li et al., 2023). Multilingual semantic parsing has been benchmarked for SQL across seven languages, none of them Bangla (Dou et al., 2023), and multilingual KGQA datasets cover nine languages each, again without Bangla (Perevalov et al., 2022; Sen et al., 2022). Translating an English benchmark while keeping its Cypher, as Ozsoy and Tai (2025) do for Spanish and Turkish, would give a larger Bangla set cheaply, but over schemas unrelated to Bangladesh; it would complement a benchmark on a real Bangladeshi graph rather than replace it.

Repair from execution feedback. Feeding execution results back to the generator is an established remedy in code generation (Chen et al., 2024) and Text2SQL, where DIN-SQL adds a self-correction step (Pourreza and Rafiei, 2023) and MAC-SQL a refiner agent that rewrites queries on SQL errors (Wang et al., 2025). We apply the minimal one-turn version of this loop to Text2Cypher and, to our knowledge for the first time, measure it per language.

The gap. Taken together, these lines leave three openings. No agricultural knowledge graph in-

tegrates Bangladeshi land suitability, crop calendars and production statistics on one administrative spine with a natural-language layer. No Bangla resource grounds question answering in a structured graph, and the only Bangla structured-query data targets relational rather than graph schemas. And no Text2Cypher benchmark covers Bangla or a curated government-statistics graph. BD-AgriKG addresses all three: to our knowledge it is the first execution-based Text2Cypher benchmark in Bangla and the first on a real government-statistics property graph.

3 BD-AgriKG

3.1 Administrative spine and entity resolution

Figure 2 summarises how the graph is built. All administrative units are keyed by P-codes from version 3 of the UN OCHA Common Operational Datasets on administrative boundaries (COD-AB). The 466 crop-zoning units of the Bangladesh Agricultural Research Council (BARC) were resolved onto the 507 admin-3 units (upazilas) in four ways: 310 matched exactly after normalisation, 55 through *sadar* naming conventions (the upazila that carries its district's name), 82 through fuzzy matches that were each audited by hand, and 21 manually. Units that merged or split across vintages are handled explicitly (two BARC units map to two admin units each, yielding 468 links), including 11 upazila names that occur in two or three different districts and would silently collide under name-based keys. We have no held-out estimate of resolution accuracy; each of the 468 links is released with the method that produced it.

3.2 Sources and graph shape

The five sources are the BARC crop-suitability zoning, the BBS Yearbook of Agricultural Statistics (production), the BBS crop calendar, the agro-meteorological crop thresholds of the Bangladesh Agro-Meteorological Information Service (BAMIS), and the soil associations of the Soil Resource Development Institute (SRDI). Together they yield a graph of 650 nodes and 21,212 relationships; Appendix A lists every label, relationship type and property. Each of the 7,456 SUITABLE_IN edges links a crop to an upazila and carries the five BARC suitability-class areas together with a derived suitability factor, which is left null where the source area is zero. Each of the 11,632 PRODUCED_IN edges links a crop to a district for one crop year and carries area, production and derived yield, with provenance and a quality flag; rows with zero or missing production stay in the released panel but are not loaded into the graph. The 1,471 HAS_SOIL edges attach soil associations to upazilas, the 82 AFFECTED_BY edges attach pests and diseases to crops, IN_DISTRICT and IN_DIVISION edges carry the administrative hierarchy, and the crop calendar and agro-meteorological thresholds complete the graph as node properties.

3.3 The production panel

The production panel is extracted from ten yearbook editions. The tables come in two families with two to five value columns per year block; most editions print jute in 400 lb bales, which we convert at 181.44 kg per bale, division subtotals are numbered, and long rows are split across lines. Every extracted table is checked against its own printed division and national totals; disagreements are recorded in the released extraction report rather than gating ingestion, and in practice the cross-edition consensus rule below is what resolves them. Where a crop year appears in several editions, we take the newest value that agrees with the median production within 2%; if the newest edition disagrees, we take the median-agreeing value and flag the row, which rejected 264 newest-edition outliers. Twelve adversarial cell-level spot checks against the source PDFs all matched, but a sample that small does not establish a cell-level error rate, so the checksums remain the primary validation. A further 134 source-level anomalies are flagged rather than corrected.

Post-review audit. After the reviews we extended the validation beyond the checksums: crop years present in two editions were reconciled, per-crop plausibility bands applied, and the calendar and pest sheets re-read against the agencies' publications. This found some extraction defects (a unit and a table-parsing error in single yearbook editions, a change of series for chili, a consensus rule that preferred the newer edition, and copy-paste errors in the pest and calendar sheets), all corrected in the released data (v1.1) and listed in its changelog. A minority of gold answers change between v1.0 and v1.1; the changelog lists them. Every result in this paper is on v1.0, which is released beside v1.1; since grading is by execution,

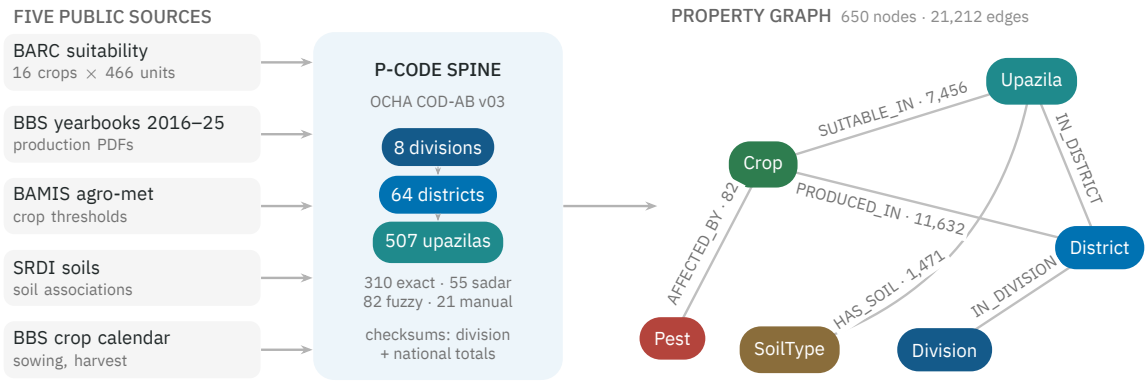


Figure 2: BD-AgriKG construction. Five government sources are resolved onto one P-code administrative spine; every extracted production table is checked against its own printed division and national totals (disagreements are logged; Section 3.3), plus 12/12 cell-level PDF spot checks.

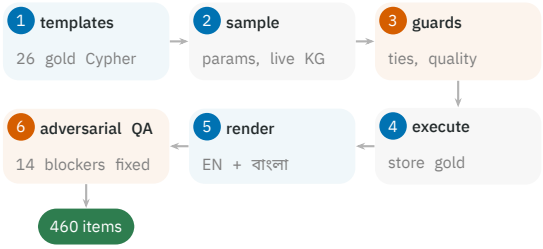


Figure 3: Benchmark construction. Questions are rendered from executed gold queries, never the reverse, so every item is answerable; sampler guards reject ranking ties and quality-flagged data.

a model's verdict measures whether its query retrieves what the graph holds.

4 Benchmark

Construction. The benchmark is built Cypher-first over the live graph (Figure 3). Each of the 26 templates pairs a parameterised gold query with an English and a Bangla question pattern. Parameters are sampled from the graph itself, the gold query is executed and its result stored, and items whose answer is degenerate (empty, tied or resting on quality-flagged data) are rejected. The result is 460 items in six categories: 118 lookup, 90 aggregation, 90 ranking, 72 multi-hop, 54 temporal and 36 comparison. Three readings that the questions leave open are fixed by the gold queries: *most suitable* ranks by `suitability_factor`; *highly suitable* means a factor above 0.7, which the prompt states (Figure 5); and the *top crop* of a district is the one with the largest production in tonnes, so high-tonnage crops such as sugarcane often win. A model that ranks suitability by the very-suitable area instead is graded wrong; across the fourteen runs, 4 of the 160 failing queries that execute on the two suitability templates return that reading, so the ambiguity costs little.

The Bangla side. Bangla questions have Bangla syntax and Bangla crop names but keep three classes of literal in Latin script, namely place names (252 items), crop years (216) and pest or disease names (54), the forms stored in the graph (Limitations discusses this choice). The Bangla templates were LLM-drafted under author supervision and checked for morphology by the Bangla-phrasing agent described below; Appendix B lists all 26 pairs. After the review period the first author, a native speaker, rated every template for fluency and for equivalence with its English twin: 22 are natural and 4 acceptable, none unnatural; 25 ask exactly what the English asks, while *seed req* differs, since the Bangla asks only for seed and the English *planting material* can be read as planting equipment. For *water req* a more natural phrasing was suggested. No template was changed, since the evaluation used them as released; the suggested rewordings are recorded in the released rating sheet.

Quality assurance. As a cross-family adequacy check, all 460 Bangla items were back-translated by Pixtral Large and judged against the English original by Llama 3.3 70B: 422 (92%) were judged equivalent, and the 38 flags, in six templates, are translator errors or idiomatic differences (e.g. *Aman* vs. *Aman rice*) that do not change what is asked (Appendix B). Quality assurance used five independent LLM review agents (Claude family), each with its own adversarial brief: grading exploits, boundary ties, gold-answer audit, Bangla phrasing and source-data cross-checks. Together they attacked the templates and the grader and

surfaced 14 blocking defects, among them grading loopholes, boundary ties, source-data errors and answer leakage. Each defect was confirmed against the source tables before being fixed, the benchmark was then regenerated, and tie and plausibility guards were added to the item sampler.

5 Evaluation

5.1 Protocol

We score execution accuracy. A generated Cypher query runs read-only against the graph with a 15 s timeout, and its result is compared with the stored gold result as a multiset of row values with a relative numeric tolerance of 10^{-6}. Under the strict metric any extra returned column is a failure, write clauses are rejected outright, and models are instructed to return only the values asked for. A gold adapter that replays the reference queries scores 460/460. Because the strict metric fails on any extra returned column, a query that dumps columns to improve its chances cannot pass by construction. All runs are single-pass with a 1,500-token output limit. Non-Anthropic models decode at temperature 0. Anthropic models run without sampling parameters, that is at the API's default temperature, because Claude 5 rejects them and Haiku 4.5 was run the same way for uniformity; their item-level outcomes are therefore not deterministic, and Appendix C measures this for one model with a repeated run. DeepSeek-R1's reasoning trace, which the API returns separately, is discarded before query extraction. Exact model identifiers, run dates and settings are listed in Appendix C. Because the 460 items instantiate 26 templates, we treat templates as the sampling unit. Template-clustered percentile-bootstrap 95% confidence intervals (CIs; 2,000 resamples, no multiple-comparison correction) span ±6–17 points for absolute accuracy, so we do not read small cross-model differences as rankings. The *paired* English–Bangla (EN–BN) gap, computed on identical items, is estimated far more precisely and is the quantity our language claims rest on.

5.2 Baselines

We evaluate seven models on Amazon Bedrock, zero-shot with the schema prompt, covering four vendor families and a large-to-small spread: Claude Opus 5, Sonnet 5 and Haiku 4.5 (one family, three sizes), Llama 3.3 70B and Llama 3.1 8B,

	English			Bangla			
Model	val	str	len	val	str	len	Δ
Opus 5	99.8	**95.2**	99.3	100.0	**90.9**	99.3	+4.3
Sonnet 5	100.0	94.8	99.6	100.0	84.3	97.4	+10.4
DeepSeek-R1	88.0	80.4	81.5	86.7	73.7	77.4	+6.7
Pixtral Large	91.7	78.3	80.0	89.8	64.6	65.4	+13.7
Haiku 4.5	88.3	51.3	52.8	90.0	47.4	50.9	+3.9
Llama 3.3 70B	67.8	34.1	34.1	71.3	28.9	28.9	+5.2
Llama 3.1 8B	85.4	8.9	8.9	86.1	8.0	8.0	+0.9

Table 1: Valid-Cypher rate (val), strict (str) and lenient (len) execution accuracy in percent, zero-shot; Δ is the English-to-Bangla strict gap in points, computed from item counts, so it can differ by 0.1 from the difference of the rounded columns. Opus, Sonnet and Haiku are Claude models. Relative to the reviewed version, the grader now reads a single `collect()` list as rows and matches booleans only to booleans; this changed five DeepSeek-R1 verdicts (two English, three Bangla) and two of its repairs, and no other number.

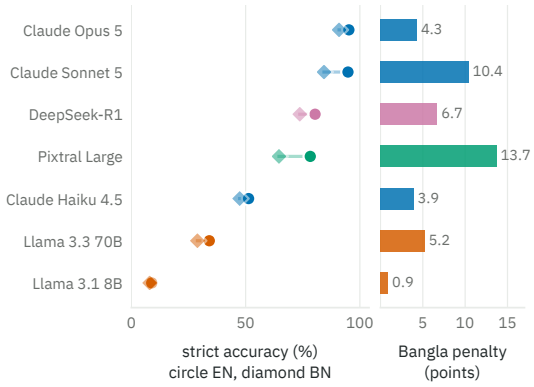


Figure 4: Left: English (circles) and Bangla (diamonds) strict execution accuracy per model. Right: the Bangla penalty isolated, in points; it is largest for Pixtral Large and Sonnet 5 and smallest for Llama 3.1 8B, Haiku 4.5 and Opus 5 (per-category detail in Appendix D).

Pixtral Large 25.02, and the reasoning model DeepSeek-R1. We report the strict metric (exact projection) and a lenient variant that tolerates at most one extra returned value per row, which accepts "winner plus its metric" answers while still rejecting column dumping.

Three findings stand out (Table 1 and Figure 4). First, the benchmark has headroom and spread: the best model solves 19 of 20 English questions while an 8B open model solves fewer than 1 in 10, so zero-shot prompting separates the current model landscape without saturating it (few-shot and fine-tuned baselines are left to future work). Second, every model scores lower in Bangla, but the evidence is of two kinds. The paired template-clustered

95% interval on the EN–BN gap excludes zero for five of the seven models (Opus 5 $[0.2, 10.0]$, Sonnet 5 $[0.9, 21.6]$, DeepSeek-R1 $[1.9, 11.7]$, Pixtral Large $[4.7, 21.1]$, Llama 3.3 70B $[0.7, 13.4]$); it is indistinguishable from zero for the weakest model (Llama 8B, $+0.9$, $[-0.9, 3.7]$) and for Claude Haiku 4.5 ($+3.9$, $[-1.9, 10.5]$). Seven models are seven tests, so we also report the Holm-corrected result, under which three gaps remain significant (DeepSeek-R1, Pixtral Large and Llama 3.3 70B), and two template-level tests, a sign-flip permutation test and a sign test over the 26 templates, under each of which the same three are significant before correction and only Pixtral Large after it (Table 5). The two kinds of evidence separate the models. For Pixtral Large and DeepSeek-R1 the penalty is distributed: 21 and 13 templates favour English, the two largest templates carry 29% and 39% of the gap, and without them the gap is still 10.6 and 4.5 points. For Opus 5, Sonnet 5, Haiku 4.5 and Llama 70B it is concentrated: two templates carry 67–89% of each gap (for Opus 5, *top districts production* and *compare two districts*; for Sonnet 5, *water req of top crop* and *top districts production*; for Llama 70B, mostly *count crops pest*), and without them the gap is 0.5–3.8 points. For these four models the penalty sits in a few templates rather than across the language. With one wording per template we cannot in general separate a template's phrasing from the Bangla construction it uses, but for Llama 70B the wording is visible: the Bangla *count crops pest* question makes the pest its subject where the English is passive, and all 18 of its Bangla queries reverse the AFFECTED_BY edge, against 1 of 18 in English. The gap is not monotone in capability: it is largest for Pixtral Large (13.7 points) and Sonnet 5 (10.4) and smallest for the weakest model (0.9), Haiku 4.5 (3.9) and the strongest (Opus 5, 4.3), though the overlapping intervals do not support an ordering among the gaps. Under the lenient metric the frontier gap nearly vanishes (Opus 5 99.3 in both languages; Sonnet 5 99.6 vs. 97.4), so at the frontier the residual Bangla deficit is mostly output formatting rather than query logic. The three gaps significant under the corrected bootstrap are not mainly formatting: under the lenient metric they are 4.1 points for DeepSeek-R1, 14.6 for Pixtral Large and 5.2 for Llama 3.3 70B. Third, the difference between strict and lenient scores localises failure types. For the open models it is negligible, because their errors are genuinely wrong queries, with hallucinated properties such as `p.crop` or `c.PRODUCED_IN` or missed multi-hop structure; for the frontier models it absorbs almost all errors, which are correct queries with one extra column. Per-category results are shown in Figure 6 (Appendix D) and per-template results in Figure 7 (Appendix E).

Model	fmt	wrong	empty	inv
Opus 5	**6.3**	0.5	0.0	0.1
Sonnet 5	**8.9**	1.5	0.0	0.0
DeepSeek-R1	2.4	6.5	1.4	**12.6**
Pixtral Large	1.3	6.8	**11.2**	9.2
Haiku 4.5	2.5	**31.5**	5.8	10.9
Llama 3.3 70B	0.0	28.5	9.6	**30.4**
Llama 3.1 8B	0.0	28.3	**48.8**	14.5

Table 2: Failure composition as a percentage of all 920 items per model (both languages pooled, so each row sums to the model's pooled failure rate): formatting, wrong result, empty result, invalid Cypher. The dominant mode (bold) shifts from formatting to logic to grounding as capability decreases.

5.3 Error analysis

Re-executing every failed query yields a four-way taxonomy (Table 2): *formatting* (correct answer, extra returned values), *wrong* (executes, non-empty, incorrect), *empty* (executes but matches nothing), and *invalid* (does not execute). Failure modes shift systematically with capability. Frontier models fail almost exclusively by formatting (58 of Claude Opus 5's 64 failures; its single invalid query is a stray colon before `MATCH`). Mid-tier models fail by logic. Claude Haiku 4.5's dominant error is a plausible query that executes and returns the wrong answer. Small open models fail by grounding. Nearly half of Llama 3.1 8B's items execute yet return nothing because they invent schema elements (`c.PRODUCED_IN` as a node property, a `HAS` relationship), which Cypher silently matches to zero rows. DeepSeek-R1 and Llama 3.3 70B additionally produce a large share of invalid Cypher despite the schema being in the prompt.

Why Llama 3.3 70B scores zero on ranking, temporal and multi-hop items. A reviewer asked whether a uniform zero over these 216 items per language could be a pipeline artefact. It is not. In English, 114 of the 216 queries fail to parse and 102 run; of those 102, 89 reference a relationship property through a node variable and the relationship name, as in `ORDER BY c.produced_in.production_mt`. Cypher ac-

cepts this (it reads a property called `produced_in` on the node, which does not exist) and sorts by null, so the query returns an arbitrary district. The Bangla runs show the same idiom in 51 of 114 valid-but-wrong queries. Llama 3.1 8B fails the same items differently: it reverses the PRODUCED_IN edge, so the node it sorts on matches nothing and the query returns no rows (35 English and 22 Bangla queries). Neither error is visible to a syntax check, which is why the taxonomy counts these queries as wrong or empty rather than invalid, and why execution-based grading is necessary here.

5.4 Execution-feedback repair

Because grading is execution-based, the signal that scores a model is also available to an agent at inference time. We test the simplest form of this, a single retry. Every first attempt that was invalid or returned zero rows is retried once: the model receives the same system prompt and question, its previous query, and a fixed sentence carrying either the Neo4j error message (at most 300 characters) or the notice that the query executed but returned zero rows, and is asked for a corrected query under the same rules. Empty results are identified by re-executing the first query with a 6 s timeout; one of the 1,421 retried items timed out at this step and was retried with the timeout message. Wrong-but-non-empty results receive no retry, since no honest signal exists without the gold answer, and no API call failed. Table 3 reports repairable counts, recoveries and strict accuracy after one turn; Appendix F splits them by signal type.

	English			Bangla		
Model	rep.	rec.	acc.	rep.	rec.	acc.
Opus 5	1	1	95.2→95.4	0	0	90.9→90.9
Sonnet 5	0	0	94.8→94.8	0	0	84.3→84.3
Haiku 4.5	72	42	51.3→60.4	81	46	47.4→57.4
DeepSeek-R1	58	37	80.4→88.5	71	52	73.7→85.0
Pixtral Large	74	34	78.3→85.7	114	49	64.6→75.2
Llama 3.3 70B	191	11	34.1→36.5	177	2	28.9→29.3
Llama 3.1 8B	304	1	8.9→9.1	278	0	8.0→8.0

Table 3: One-turn execution-feedback repair. rep.: first attempts that were invalid or returned zero rows (repairable); rec.: recovered to a strictly correct answer; acc.: strict execution accuracy before and after the repair turn (%). Wrong-but-non-empty answers are not retried.

Three observations. First, the ability to use feedback is itself a capability, and it separates the tiers: Claude Haiku 4.5, DeepSeek-R1 and Pixtral Large recover 43–73% of their repairable items, gaining 7–11 points, whereas Llama 3.3 70B recovers 13 of 368 and Llama 3.1 8B one of 582. Within the mid-tier, however, recovery rate does not follow first-pass accuracy (Haiku 58%, Pixtral 44%). The open models largely ignore the message: Llama 70B reproduces the same undefined-variable error in 88 of the 181 retries that carried it, and 189 of its 368 second attempts are still invalid; Llama 8B returns a byte-identical query in 85 of 304 English retries. For Haiku 4.5 the gain is not sampling variance: a repeated Haiku zero-shot run without feedback makes only 10 of the same 72 items correct (Appendix C), against 42 with the error message; 36 items are recovered by feedback only and 4 by resampling only (exact McNemar $p = 1.9 \times 10^{-7}$). We have no such control for DeepSeek-R1 or Pixtral Large. Frontier models have almost nothing to repair (one invalid Opus 5 query, none for Sonnet 5), consistent with their residual errors being formatting rather than grounding. Second, the signal matters (Appendix F, both languages pooled). Haiku fixes 66 of 100 syntax errors but only 22 of 53 empty results, where the notice carries no diagnosis; DeepSeek-R1's failures are almost all syntactic and it fixes 82 of 116; Pixtral shows the opposite profile, repairing 56 of 103 empty results but only 27 of 85 syntax errors, typically by dropping an over-restrictive filter. Third, repair helps Bangla at least as much as English. Bangla first attempts fail more often in repairable ways (81 vs. 72, 71 vs. 58 and 114 vs. 74 items for the three mid-tier models), the Bangla gain exceeds the English gain for all three, and the paired EN–BN gap moves from 13.7 to 10.4 points for Pixtral, 3.9 to 3.0 for Haiku and 6.7 to 3.5 for DeepSeek-R1, whose gap is no longer distinguishable from zero (95% CI $[-1.8, 9.4]$) while Pixtral's remains significant (uncorrected) and Haiku's was not significant before repair either. None of these narrowings is individually significant with 26 template clusters: the 95% intervals on the change are $[-1.9, 8.7]$ for Pixtral, $[-2.2, 8.3]$ for DeepSeek-R1 and $[-2.8, 4.5]$ for Haiku. We therefore read the narrowing as consistent in direction, not as an established effect. Llama 3.3 70B moves the other way (5.2 to 7.2 points) because the few repairs it manages are in English.

6 Conclusion

We released BD-AgriKG, a knowledge graph that makes five scattered Bangladeshi government sources jointly queryable on one administrative spine, together with the first bilingual execution-based Text2Cypher benchmark and baselines for seven LLMs. The benchmark shows a lower Bangla score for every model, significant for three of seven under multiple-comparison control, carried by two templates for four of them and nearly vanishing under lenient grading at the frontier, and an error taxonomy whose dominant failure mode shifts from formatting to logic to schema grounding as capability decreases. The graph, panel, benchmark and harness support two immediate directions: fine-tuning small open models on Bangla Text2Cypher, and generating KG-grounded agricultural reports whose every claim can be verified by executing queries against the same graph. Because grading is execution-based, the benchmark doubles as a verifiable environment for systems that act on execution feedback: one turn recovers 43–73% of the repairable failures of mid-tier models and helps their Bangla at least as much as their English, though not significantly more, while small open models cannot yet use the signal.

Limitations

Suitability reflects 2015-era zoning; production is district-level (no public upazila-level production exists); 20 % of panel cells rest on a single yearbook edition; source misprints are flagged rather than corrected. The benchmark's effective sample size is its 26 templates rather than its 460 items, which our clustered intervals make explicit; each template has one fixed wording and no item is a question asked by a farmer or extension worker, so the results describe these 26 phrasings rather than Bangla Text2Cypher in general; paraphrases and naturally authored questions are future work. Every result is a single run, and the Claude models were sampled at the API's default temperature; the one repeated run (Haiku 4.5, English) moved accuracy by 0.4 points but changed the outcome of 64 of 460 items, so per-template findings carry run-to-run variation that our intervals do not include. Single-value integer answers cluster on a few common values, so a wrong query can occasionally coincide with the right answer. Bangla questions keep place names, years and pest or disease names in Latin script. Code-mixing of this kind is common in Bangladeshi extension material, but several of these pests do have established Bangla names (*Rice Bug* is গান্ধি পোকা, *Aphid* is জাব পোকা), so for the 54 affected items the Latin form is a simplification, and the native-speaker rating of Appendix B judged that an extension worker would write district names, years and pest names in Bangla script as well. Since every Latin-script literal matches the graph verbatim and removes an entity-linking step, fully Bangla surface forms would likely widen the measured gap, an expectation we have not tested. The adversarial QA agents were drawn from a single model family that is also among the systems evaluated; this circularity is mitigated, though not eliminated, by execution-based grading, by manual confirmation of every defect and by the cross-family adequacy check of Appendix B, and the native-speaker rating was made by an author after the review period, not by an independent speaker before evaluation. In particular, Claude-authored phrasing may be easier for Claude models to parse, so comparisons between Claude and non-Claude rows of Table 1 should be read with that caveat. The repair loop is a single turn on failures that carry an execution signal, so wrong-but-non-empty answers are untouched, the only no-feedback control is Haiku 4.5 in English, and the zero-rows notice presumes a non-empty answer exists, which holds for every benchmark item by construction; a deployed agent would need an abstention path. Two baselines we did not run would sharpen the comparison: an existing English Text2Cypher benchmark rendered in Bangla, which would separate the cost of the language from that of this graph's content, and few-shot or fine-tuned models.

Acknowledgements

We thank the three NORA reviewers; their comments led to the corrected significance analysis, the per-template analysis, the Llama 70B inspection and the post-review data audit in this version.

Generative AI use

Generative AI assisted with drafting and revising the manuscript, with literature search, and with parts of the code. The authors reviewed all output and take full responsibility for the content of this paper.

Availability

The graph tables, production panel, benchmark, evaluation harness and all raw per-item model outputs are released at https://github.com/aukerul-shuvo/BD-AgriKG (code under MIT, derived data under CC BY 4.0) and archived with a DOI at https://doi.org/10.5281/zenodo.23203983. The release holds both the v1.0 data on which the results were obtained and the corrected v1.1 data, with a changelog.

References

Muhammad Ibrahim Ameen, Tasnim Islam, Mst. Sanjida Aktar, and Redwan Karim Rafat. 2025. KrishokBondhu: A retrieval-augmented voice-based agricultural advisory call center for Bengali farmers. *Preprint*, arXiv:2510.18355.

Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. 2024. Teaching large language models to self-debug. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Yuanzhe Chen et al. 2019. AgriKG: An agricultural knowledge graph and its applications. In *Database Systems for Advanced Applications (DASFAA)*, LNCS. Springer.

Aviral Dawar, Roshan Karanth, Vikram Goyal, and Dhruv Kumar. 2026. IndicDB: Benchmarking multilingual text-to-SQL capabilities in Indian languages. *Preprint*, arXiv:2604.13686.

Rudra Pratap Deb Nath, Archita Das, Anika Das, and Mahmudul Raihan. 2024. Knowledge graph generation and enabling multidimensional analytics on Bangladesh agricultural data. *IEEE Access*, 12:87512–87531.

Longxu Dou, Yan Gao, Mingyang Pan, Dingzirui Wang, Wanxiang Che, Dechen Zhan, and Jian-Guang Lou. 2023. MultiSpider: Towards benchmarking multilingual text-to-SQL semantic parsing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12745–12753.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From local to global: A graph RAG approach to query-focused summarization. *Preprint*, arXiv:2404.16130.

Yanlin Feng, Simone Papicchio, and Sajjadur Rahman. 2024. CypherBench: Towards precise retrieval over full-scale modern knowledge graphs in the LLM era. *Preprint*, arXiv:2412.18702.

Kadiriye Ibrahim, Kalimuthu Senthilkumar, and Kazuki Saito. 2024. Evaluating responses by ChatGPT to farmers' questions on irrigated lowland rice cultivation in Nigeria. *Scientific Reports*.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, Xuanhe Zhou, Chenhao Ma, Guoliang Li, Kevin Chang, Fei Huang, Reynold Cheng, and Yongbin Li. 2023. Can LLM already serve as a database interface? a BIg bench for large-scale database grounded text-to-SQLs. In *Advances in Neural Information Processing Systems*, volume 36, pages 42330–42357.

Makbule Gulcin Ozsoy, Leila Messallem, Jon Besga, and Gianandrea Minneci. 2025. Text2Cypher: Bridging natural language and graph databases. In *Proceedings of the Workshop on Generative AI and Knowledge Graphs (GenAIK) at COLING*.

Makbule Gulcin Ozsoy and William Tai. 2025. Text2Cypher across languages: Evaluating and finetuning LLMs. *Preprint*, arXiv:2506.21445.

Aleksandr Perevalov, Dennis Diefenbach, Ricardo Usbeck, and Andreas Both. 2022. QALD-9-plus: A multilingual dataset for question answering over DBpedia and Wikidata translated by native speakers. In *2022 IEEE 16th International Conference on Semantic Computing (ICSC)*, pages 229–234.

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 36 (NeurIPS 2023)*, volume 36, pages 36339–36348.

Md. Tanzim Reza, Tahsina Nimi, and Riasat Shahid. 2026. A provenance-traceable multi-task Bengali agricultural benchmark with safety-critical chemical advisory. *Preprint*, arXiv:2606.29243.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1604–1619.

Sheikh Shafayat, H M Quamran Hasan, Minhajur Rahman Chowdhury Mahim, Rifki Afina Putri, James Thorne, and Alice Oh. 2024. BEnQA: A question answering benchmark for Bengali and English. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1158–1177.

Zhili Shen, Chenxin Diao, Pavlos Vougiouklis, Pascual Merita, Shriram Piramanayagam, Enting Chen, Damien Graux, Andre Melo, Ruofei Lai, Zeren Jiang, Zhongyang Li, Ye Qi, Yang Ren, Dandan Tu, and Jeff Z. Pan. 2025. GeAR: Graph-enhanced agent for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12049–12072.

Bing Wang, Changyu Ren, Jian Yang, Xinnian Liang, Jiaqi Bai, LinZheng Chai, Zhao Yan, Qian-Wen Zhang, Di Yin, Xing Sun, and Zhoujun Li. 2025. MAC-SQL: A multi-agent collaborative framework for text-to-SQL. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING)*, pages 540–557.

Azmine Toushik Wasi, Taki Hasan Rafi, Raima Islam, and Dong-Kyu Chae. 2024. BanglaAutoKG: Automatic Bangla knowledge graph construction with semantic neural graph filtering. In *Proceedings of LREC-COLING 2024*, pages 2100–2106.

Jian Wu, Wenzhe Xie, Tao Wang, Xinyu Fan, Yang Li, and Xiangzeng Meng. 2025. Crop GraphRAG: pest and disease knowledge base question answering system for sustainable crop protection. *Frontiers in Plant Science*, 16.

Shuting Yang, Zehui Liu, and Wolfgang Mayer. 2024. ShizishanGPT: An agricultural large language model integrating tools and resources. In *Web Information Systems Engineering (WISE)*, LNCS. Springer.

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.

Xin Zhou et al. 2026. Knowledge-enhanced large language model construction for garlic cultivation using GraphRAG. *Frontiers in Plant Science*.

A Graph schema

The schema is small: 6 node labels, 6 relationship types and 42 label-specific property keys (Table 4). The prompt carries its query-relevant subset verbatim (Figure 5), so the benchmark tests grounding and composition on a fully specified schema rather than schema retrieval. Per graph this is a schema of ordinary size for the task: CypherBench graphs average 7.5 relation types and 18.7 properties (Feng et al., 2024), also given in full in the prompt; the difference is entity scale, 650 nodes here against hundreds of thousands in a Wikidata-derived graph.

B Template inventory

Tables 6 and 7 list one released item for each of the 26 templates in both languages, with the number of items instantiated from it. Bangla items keep place names, years and pest or disease names in Latin script, as these are the forms stored in the graph; crop names are always in Bangla.

Cross-family adequacy check. Each of the 460 Bangla items was back-translated to English by Pixtral Large (given the benchmark's 16-entry crop-name glossary so that vocabulary recall is not what is tested) and Llama 3.3 70B judged whether the back-translation asks for the same entities, quantity and constraints as the original English item; neither model belongs to the family that drafted or audited the Bangla. 422 of 460 items were judged equivalent. The 38 flags fall in six templates: *water req of top crop* (18; Bangla *prodhan foshol* does not mark number and was rendered "major crops", yet every model that answered it in Bangla returned a single crop, and Sonnet 5's

```
You translate natural-language questions about
↪ Bangladeshi agriculture into Cypher queries for a
↪ Neo4j knowledge graph.

Node labels and properties:
- Division {pcode, name}
- District {pcode, name}
- Upazila {pcode, name, area_sqkm, center_lat,
↪ center_lon}
- Crop {name, sowing_time, harvest_time,
↪ seed_requirement_per_acre,
        favourable_temperature, favourable_humidity,
        favourable_soil_temperature, photo_period,
↪ favourable_rainfall,
        water_requirement}
- SoilType {name}
- PestDisease {name}

Relationships:
- (Upazila)-[:IN_DISTRICT]->(District)
- (District)-[:IN_DIVISION]->(Division)
- (Crop)-[:SUITABLE_IN {very_suitable, suitable,
↪ moderately_suitable,
   marginally_suitable, not_suitable, total_area, zone,
↪ suitability_factor}]
  ->(Upazila)   // BARC land-suitability areas in
↪ hectares; factor 0..1
- (Upazila)-[:HAS_SOIL {characteristics}]->(SoilType)
- (Crop)-[:AFFECTED_BY {conditions}]->(PestDisease)
- (Crop)-[:PRODUCED_IN {crop_year, area_acres,
↪ production_mt,
   yield_mt_per_ha, source, quality_flag}]->(District)
  // BBS district production; crop_year like '2023-24',
↪ 2013-14..2024-25

The 16 crops: Aus, Aman, Boro, Wheat, Maize, Potato,
↪ Lentil, Mung, Gram,
Mustard, Groundnut, Chili, Onion, Garlic, Sugarcane,
↪ Jute.
District names use current spellings (Bogura,
↪ Chattogram, Jashore,
Chapainababganj, Barishal, Cumilla, ...).
A crop is called "highly suitable" in a place when
↪ suitability_factor > 0.7.

Rules:
- Output ONLY the Cypher query, no explanation, no code
↪ fences.
- The question may be in English or Bangla; entity
↪ names in the graph are in
  English (crops: Aus, Aman, Boro, ...; districts:
↪ current spellings).
- Read-only queries only (MATCH/RETURN); never CREATE,
↪ MERGE, SET or DELETE.
- Return ONLY the value or values the question asks
↪ for, nothing extra, and
  alias the main answer column AS answer.
```

Figure 5: The system prompt given verbatim to every model, in both languages; the question follows as the user turn. Long lines are broken at the column edge, and each continuation is marked ↪. The schema block is the query-relevant subset of Table 4: provenance properties (`source_span`, `barc_source_name`, and `source` except on `PRODUCED_IN`) are omitted, and the coordinate keys are named `center_lat` and `center_lon` where the graph stores `lat` and `lon`, a mismatch no released item touches.

Label or type	Count	Properties
Division	8	name, pcode
District	64	name, pcode
Upazila	507	name, pcode, area_sqkm, lat, lon
Crop	16	name, sowing_time, harvest_time, seed_requirement_per_acre, water_requirement, favourable_temperature, favourable_rainfall, favourable_humidity, favourable_soil_temperature, photo_period
SoilType	20	name
PestDisease	35	name
IN_DIVISION	64	–
IN_DISTRICT	507	–
SUITABLE_IN	7,456	suitability_factor, very_suitable, suitable, moderately_suitable, marginally_suitable, not_suitable, total_area, zone, barc_source_name, source, source_span
HAS_SOIL	1,471	characteristics, source
AFFECTED_BY	82	conditions, source
PRODUCED_IN	11,632	crop_year, production_mt, area_acres, yield_mt_per_ha, quality_flag, source

Table 4: Node labels (top) and relationship types (bottom) of BD-AgriKG with instance counts and property keys; 650 nodes and 21,212 relationships in total.

Bangla failures on it are an extra returned column, not a plural reading); *sowing time* (8; *ropon* covers both planting and transplanting); *count crops pest* (6; the translator dropped *koyti*, "how many", which is present in the Bangla); *pest conditions* and *water req* (5; Bangla idiomatically adds *dhan*, "rice", to the season names Aman, Aus and Boro); and *suitable and affected* (1; the translator dropped *otyonto*, "highly"). None changes the quantity, entities or constraints asked for, so no template was revised. Per-item back-translations and verdicts are released with the benchmark. This check bounds translation adequacy; it is not a native-speaker fluency rating.

Native-speaker rating. After the review period the first author, a native Bangla speaker, rated each of the 26 templates on two example items with different parameters, for fluency (1 unnatural, 2 acceptable, 3 natural) and for whether the Bangla asks exactly what the English asks. Fluency: 22 templates rated 3 and four rated 2 (*seed req*, *water req*, *division of top producer*, *pests of top crop*), none rated 1. Equivalence: 25 yes and one no, *seed req*, since the Bangla বীজ asks only for seed, while the English *planting material* can be read as planting equipment. Two rewordings were suggested without a change of meaning, পানি কেমন লাগে for পানির চাহিদা in *water req* and দেশে for জাতীয় উৎপাদন in *compare two crops*. The rater also noted that an extension worker would write district names, years and pest names in Bangla script, which is the design choice discussed in Limitations. The filled rating sheet is released with the benchmark; no template was changed after evaluation.

C Reproducibility details

All model runs used the Amazon Bedrock Converse API from a single account in region us-east-1 with cross-region inference profiles. The zero-shot sweep ran on 31 August 2026 and the repair runs on 2 September 2026, each item once, with a 1,500-token output limit. Non-Anthropic models were decoded at temperature 0; Anthropic models were called without sampling parameters. Table 8 gives the exact identifiers.

Because Anthropic models therefore sample at the API default, we repeated the Claude Haiku 4.5 English zero-shot run once (2 September 2026). Aggregate strict accuracy moved from 51.3% to 50.9% (236 vs. 234 correct), but 64 of 460 items changed outcome (31 failures became correct, 33 correct items failed) and only 136 of 460 queries were byte-identical across the two runs. Single-run differences of one or two points between models should therefore not be read as rankings, which is also what the clustered intervals imply. The repeated run doubles as a no-feedback control for Section 5.4: of the 72 Haiku English items that were invalid or empty in the first run, re-sampling alone made 10 correct (8 of 54 invalid, 2 of 18 empty), against 42 with the execution message.

D Per-category results

Figure 6 breaks strict execution accuracy down by question category for every model and language. Lookup questions saturate for every capable model, while comparison and multi-hop questions produce the widest spread between models. Table 5 gives, per model, the EN–BN gap with the interval, the raw and Holm-corrected p-values of three tests, and how much of each gap two templates carry.

E Per-template results

Because the 26 templates are the effective sampling unit (Section 5.1), Figure 7 reports strict execution accuracy for every template, model and language; the item count per template is given in parentheses. One template stands out: *compare two crops* is failed strictly even by the frontier models, which return the winning crop together with

Model	gap	95% CI	bootstrap p raw	bootstrap p Holm	sign-flip p raw	sign-flip p Holm	templates EN>BN / BN>EN / tie	sign test p raw	sign test p Holm	top-2 share	gap w/o top-2
Opus 5	+4.3	$[0.2, 10.0]$	0.044	0.132	0.126	0.496	5 / 1 / 20	0.219	0.474	85%	+0.7
Sonnet 5	+10.4	$[0.9, 21.6]$	0.023	0.092	0.124	0.496	4 / 0 / 22	0.125	0.474	67%	+3.8
DeepSeek-R1	+6.7	$[1.9, 11.7]$	0.006	0.030	0.019	0.113	13 / 3 / 10	0.021	0.128	39%	+4.5
Pixtral Large	+13.7	$[4.7, 21.1]$	0.005	0.030	0.005	0.034	21 / 2 / 3	0.000	0.000	29%	+10.6
Haiku 4.5	+3.9	$[-1.9, 10.5]$	0.215	0.430	0.291	0.582	11 / 4 / 11	0.118	0.474	89%	+0.5
Llama 3.3 70B	+5.2	$[0.7, 13.4]$	0.000	0.000	0.031	0.157	6 / 0 / 20	0.031	0.156	83%	+0.9
Llama 3.1 8B	+0.9	$[-0.9, 3.7]$	0.717	0.717	1.000	1.000	1 / 2 / 23	1.000	1.000	150%	-0.5

Table 5: Tests of the EN–BN strict gap per model (points). The interval and p are the paper's template-clustered percentile bootstrap; the sign-flip test permutes the sign of each template's paired difference; the sign test counts templates by the direction of their difference; Holm corrects each across the seven models. The last two columns give the share of the gap carried by its two largest templates and the gap recomputed without them.

Template (category)	n	English example	Bangla example
harvest time (lookup)	16	When is Onion harvested?	পেঁয়াজ কখন ঘরে তোলা হয়?
pest conditions (lookup)	18	Under what conditions does Rice Bug attack Aman?	কোন অবস্থায় আমন ধানে Rice Bug আক্রমণ করে?
seed req (lookup)	16	How much seed or planting material is needed per acre for Sugarcane?	আখ চাষে একর প্রতি কী পরিমাণ বীজ লাগে?
sowing time (lookup)	16	When is Wheat sown, planted or transplanted in Bangladesh?	বাংলাদেশে গম কখন বপন বা রোপণ করা হয়?
upazila district (lookup)	18	Which district is Mithapukur upazila in?	Mithapukur উপজেলা কোন জেলায় অবস্থিত?
upazila soils (lookup)	18	What soil types are found in Sreebardi upazila?	Sreebardi উপজেলায় কোন কোন ধরনের মাটি পাওয়া যায়?
water req (lookup)	16	What is the water requirement of Groundnut?	চীনাবাদাম চাষে পানির চাহিদা কেমন?
count crops pest (aggregation)	18	How many crops are attacked by Seedling Blight?	Seedling Blight কয়টি ফসলে আক্রমণ করে?
count producing districts (aggregation)	18	In how many districts was Lentil produced in 2015-16?	2015-16 সালে কতটি জেলায় মসুর উৎপাদিত হয়েছিল?
count upazilas (aggregation)	18	How many upazilas are there in Netrakona district?	Netrakona জেলায় কয়টি উপজেলা আছে?
division production (aggregation)	18	How much Groundnut was produced in Chattogram division in 2015-16?	2015-16 সালে Chattogram বিভাগে কত চীনাবাদাম উৎপাদিত হয়েছিল?
national production (aggregation)	18	What was the total national production of Mustard in 2024-25?	2024-25 সালে সারা দেশে সরিষার মোট উৎপাদন কত ছিল?
best upazila suitability (ranking)	18	Which upazila of Brahmanbaria district is most suitable for growing Aus?	Brahmanbaria জেলার কোন উপজেলা আউশ ধান চাষের জন্য সবচেয়ে উপযোগী?

Table 6: Template inventory, part 1 of 2: one released item per template, with the number of items it yields.

the compared production total; the lenient metric of Table 1 accepts these answers, so the template measures output discipline rather than query logic.

F Repair by signal type

Table 9 splits the repairable first attempts of Section 5.4 by the signal the model received (a Neo4j error message for invalid queries, a zero-rows notice for empty results) and reports how many of each were recovered to a strictly correct answer. Tables 2 and 9 classify empty versus invalid by re-executing failed queries with 3 s and 6 s timeouts; two Llama 3.1 8B queries that time out at 3 s return zero rows at 6 s, hence the two-item difference between the tables. The retry turn keeps the system prompt and appends to the original question the text "`Your previous query:`" followed by the first-attempt Cypher, then one of "`Executing your query failed with this error: <message>.`" or "`Executing your query executed but returned zero rows.`", and finally "`Reply with a corrected Cypher query only, following the same`

Template (category)	n	English example	Bangla example
highest yield district (ranking)	18	Which district had the highest Maize yield per hectare in 2024-25?	2024-25 সালে কোন জেলায় হেক্টর প্রতি ভুট্টার ফলন সবচেয়ে বেশি ছিল?
most suitable crop upazila (ranking)	18	Which three crops are most suitable for cultivation in Basail upazila?	Basail উপজেলায় চাষের জন্য সবচেয়ে উপযোগী তিনটি ফসল কী কী?
top crop in district (ranking)	18	Which crop had the highest production in Chapainababganj district in 2014-15?	2014-15 সালে Chapainababganj জেলায় কোন ফসলের উৎপাদন সবচেয়ে বেশি ছিল?
top districts production (ranking)	18	Which three districts produced the most Mustard in 2018-19?	2018-19 সালে কোন তিনটি জেলায় সবচেয়ে বেশি সরিষা উৎপাদিত হয়েছিল?
area trend (temporal)	18	How has the cultivated area of Onion changed over the years in Bhola district?	Bhola জেলায় পেঁয়াজ চাষের জমির পরিমাণ বিভিন্ন বছরে কীভাবে বদলেছে?
best year yield (temporal)	18	In which year did Chuadanga district get its best Mustard yield?	Chuadanga জেলায় কোন বছর সরিষার ফলন সবচেয়ে ভালো হয়েছিল?
production two years (temporal)	18	Compare the production of Garlic in Rajbari district between 2019-20 and 2023-24.	2019-20 এবং 2023-24 সালে Rajbari জেলায় রসুনের উৎপাদন তুলনা করুন।
compare two crops (comparison)	18	Which had the higher national production in 2023-24, Chili or Groundnut?	2023-24 সালে মরিচ নাকি চীনাবাদাম, কোনটির জাতীয় উৎপাদন বেশি ছিল?
compare two districts (comparison)	18	Between Chandpur and Jhalokati, which district produced more Aus in 2019-20?	2019-20 সালে Chandpur নাকি Jhalokati, কোন জেলায় বেশি আউশ ধান উৎপাদিত হয়েছিল?
division of top producer (multihop)	18	In which division is the district that produced the most Jute in 2014-15?	2014-15 সালে সবচেয়ে বেশি পাট উৎপাদনকারী জেলাটি কোন বিভাগে অবস্থিত?
pests of top crop (multihop)	18	Which pests and diseases threaten the top crop of Gazipur district (by 2018-19 production)?	2018-19 সালের উৎপাদন অনুযায়ী Gazipur জেলার প্রধান ফসলটিতে কোন কোন পোকা ও রোগ আক্রমণ করতে পারে?
suitable and affected (multihop)	18	Which crops that are highly suitable somewhere in Feni district are attacked by Flea Beetle?	Feni জেলায় চাষের জন্য অত্যন্ত উপযোগী এমন কোন কোন ফসলে Flea Beetle আক্রমণ করে?
water req of top crop (multihop)	18	What is the water requirement of the top crop of Khulna district (by 2018-19 production)?	2018-19 সালের উৎপাদন অনুযায়ী Khulna জেলার প্রধান ফসলের পানির চাহিদা কেমন?

Table 7: Template inventory, part 2 of 2: one released item per template, with the number of items it yields.

Name in paper	Bedrock model identifier
Opus 5	`us.anthropic.claude-opus-5`
Sonnet 5	`us.anthropic.claude-sonnet-5`
Haiku 4.5	`us.anthropic.claude-haiku-4-5-20251001-v1:0`
DeepSeek-R1	`us.deepseek.r1-v1:0`
Pixtral Large	`us.mistral.pixtral-large-2502-v1:0`
Llama 3.3 70B	`us.meta.llama3-3-70b-instruct-v1:0`
Llama 3.1 8B	`us.meta.llama3-1-8b-instruct-v1:0`

Table 8: Model identifiers used for every run.

`rules.`" The strings are verbatim, including the awkward grammar of the zero-rows variant. Only failed items were retried; no correct item was re-run or re-scored.

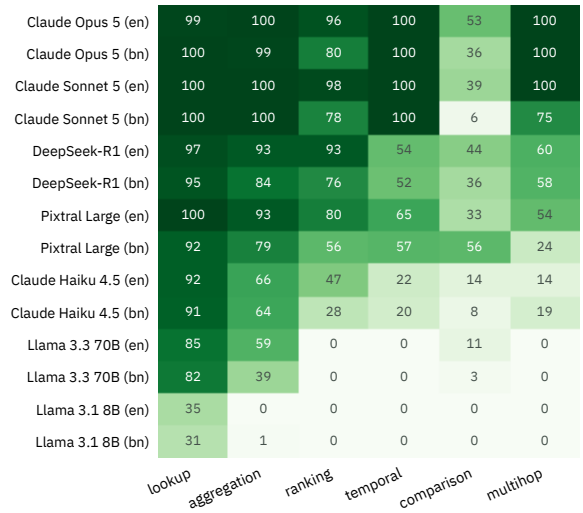


Figure 6: Strict execution accuracy per model, language and question category. Comparison and multi-hop questions separate the models most; lookup saturates for every capable model.

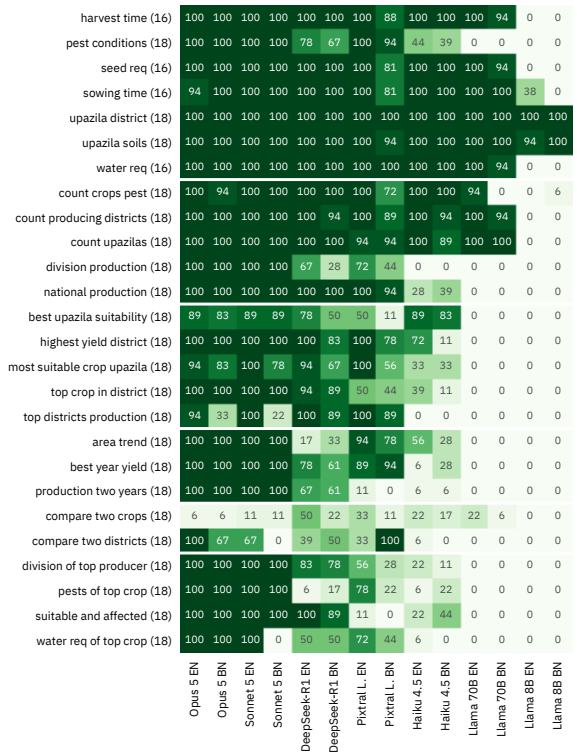


Figure 7: Strict execution accuracy (%) per template (rows, grouped by category) for each model and language (columns).

	English				Bangla			
	invalid		empty		invalid		empty	
Model	n	rec.	n	rec.	n	rec.	n	rec.
Opus 5	1	1	0	0	0	0	0	0
Sonnet 5	0	0	0	0	0	0	0	0
Haiku 4.5	54	36	18	6	46	30	35	16
DeepSeek-R1	55	36	3	1	61	46	10	6
Pixtral Large	38	12	36	22	47	15	67	34
Llama 3.3 70B	148	10	43	1	132	1	45	1
Llama 3.1 8B	67	0	237	1	64	0	214	0

Table 9: Repairable first attempts (n) and recoveries (rec.) by signal type and language; Section 5.4 quotes their row sums.