Linguistic Analysis of Neural-to-Symbolic Translation Failures in Knowledge Graph-Based Fact Checking

Ivan Alvarenga, Bruno Facco

[1]Independent Researchers – Rio de Janeiro – RJ – Brazil

ivan.alvarenga@pm.me, faccobruno98@gmail.com

Abstract. *Large Language Models are increasingly paired with Knowledge Graphs to verify factual claims, yet such systems are typically evaluated at the response level, leaving open which linguistic phenomena break the neural-to-symbolic translation from a sentence to a verifiable triple. We study this step in isolation with a within-fact design: 26 Wikidata-grounded facts, each expressed across eight linguistic conditions. gpt-4o-mini is evaluated across the full design, and Claude Haiku 4.5 provides a confirmatory comparison on four diagnostic conditions. A symbolic layer then verifies each extracted triple by SPARQL* `ASK`*; keeping the fact constant attributes every failure to the linguistic surface form. For gpt-4o-mini, failures concentrate in constructions that structurally displace the target argument from its canonical position, including exceptor phrases, where it extracts the head noun rather than the excluded entity (46.2% accuracy). Claude Haiku 4.5 likewise performs poorly on exceptor phrases (42.3%), but its errors take a different form: it declines to extract a triple in 15 of 26 cases rather than substituting the head noun. Thus, the exceptor construction is difficult across the two models, whereas the head-noun-substitution signature is model-specific. Neither model substitutes a negated entity's true-valued counterpart in the tested negation conditions. These results are descriptive and do not establish general performance across models or operational fact-checking settings.*

1. Introduction

Large Language Models (LLMs) are increasingly deployed to check factual claims, and a growing body of work grounds that checking in structured knowledge [Lavrinovics et al. 2025]: rather than trusting a model's parametric memory, a system extracts the assertion a sentence makes and verifies it against a Knowledge Graph (KG) such as Wikidata [Vrandečić and Krötzsch 2014], which offers an auditable, up-to-date check on hallucination [Sansford et al. 2024]. But the reliability of such a pipeline rests on a fragile intermediate step that is rarely examined on its own: the *neural-to-symbolic translation* from natural language to a structured triple. If that translation silently misreads the sentence, the symbolic layer verifies the wrong proposition, and the error is invisible in any end-to-end score. Existing KG-grounded verification systems are typically evaluated at the response level, reporting aggregate accuracy over heterogeneous claims [Lavrinovics et al. 2025, Agrawal et al. 2024]; such aggregates can stay high while specific linguistic constructions fail systematically — exactly what fine-grained linguistic diagnostics are designed to expose, and a reason they remain critical even in the LLM era [Opitz et al. 2025].

What is missing is an account of *which* linguistic phenomena cause the translation step to fail and *where* in the pipeline the failure surfaces. Answering that requires separating two confounds that response-level evaluation entangles. First, linguistic difficulty must be isolated from fact difficulty: if negated claims score worse than simple ones, it must be the negation, not a coincidentally harder set of facts. Second, the linguistic parsing of a sentence must be separated from the entity-linking problem of mapping a mention to a canonical identifier, since a strong entity linker can mask the very surface-form difficulties under study. We address both with a *within-fact* design — the same ground-truth fact is expressed across every linguistic condition — and a minimal pipeline that extracts triples *faithfully* (without canonicalising entities) and verifies them with a single SPARQL `ASK` against a known target. The primary study characterises gpt-4o-mini under fixed decoding. To test whether the central exceptor finding is specific to that model, we run a confirmatory comparison with Claude Haiku 4.5 on four diagnostic conditions. This remains a small, controlled study of the translation step, not an evaluation of deployed fact-checkers or a general claim about LLMs.

Our central finding is that failures concentrate not in negation, which the models handle robustly, but in constructions that displace the target argument from its canonical position, leaving the models either to extract a distractor in its place or to extract nothing at all. This frames our central research question: *what linguistic phenomena in natural-language claims cause failures in the neural-to-symbolic translation step of KG-based fact verification, and at which pipeline stage do these failures occur?* Our contributions:

- A **within-fact methodology** that holds the underlying fact constant across 8 linguistic conditions, attributing extraction failures to linguistic surface variation rather than ontological complexity.
- A **two-dimensional error taxonomy** (linguistic phenomenon $\times$ structural failure point) that localises each failure to a single pipeline stage and is populable directly from pipeline logs without manual annotation.
- An explicit **metric-scope analysis** distinguishing what a round-trip `ASK` comparison measures (argument-structure extraction) from what it cannot (negation comprehension), with the negation-substitution error identified as the sole observational signature of the latter.
- **Empirical findings** for gpt-4o-mini across the full design, including near-ceiling performance on canonical argument structure, localized failures for exceptor and reporting-verb constructions, by-design referential-gap failures, no negation-substitution errors, and no prominence gradient; plus a confirmatory Claude Haiku 4.5 comparison showing that exceptor difficulty extends across models while the head-noun-substitution failure form does not.

2. Related Work

KG-grounded LLM verification. A recent survey of knowledge graphs, LLMs, and hallucination frames KG grounding as a leading mitigation but observes that evaluation has largely remained at the response level rather than diagnosing fine-grained failure points [Lavrinovics et al. 2025]. The closest comparable system, GraphEval, builds a KG from a model's output and flags unsupported content [Sansford et al. 2024]. Our work differs in two respects. Verification is *symbolic* — a SPARQL `ASK` against Wikidata rather than a natural-language-inference check between generated text and evidence

— which keeps the decision exact and auditable; and we add an explicit *linguistic* failure analysis, crossing the phenomenon that triggered a failure with the pipeline stage at which it surfaced, which response-level systems such as GraphEval do not provide.

Neuro-symbolic integration. A complementary line studies how symbolic structure can be combined with neural models, whether by integrating logical components into LLM reasoning [Rani et al. 2025] or by leveraging the inferential depth of OWL-based knowledge graphs [Herron et al. 2025]. Relative to this taxonomy, our pipeline is an *inference-time*, post-hoc neuro-symbolic system: the neural component proposes a triple and the symbolic component verifies it, with no joint training. The SPARQL `ASK` is a lightweight but meaningful symbolic step, grounding verification in an external, queryable knowledge base rather than in the model's own parameters.

Linguistic behavioural analysis of NLP systems. Our methodology is most directly indebted to behavioural testing in the style of CheckList, which probes capabilities through targeted, minimally varied inputs rather than aggregate accuracy [Ribeiro et al. 2020]; we adapt that stance from end-to-end predictions to the neural-to-symbolic extraction step specifically, and the within-fact construction parallels minimal-pair benchmarks in linguistics [Warstadt et al. 2020]. The three negation subcategories we test (sentential, exceptor-based, implicit lexical) are drawn from a recent comprehensive taxonomy of negation for NLP, which also motivates the subcategories we exclude as incompatible with atomic binary relations [Petcu et al. 2025]; we use standard terminology for relation extraction and coreference throughout [Jurafsky and Martin 2024]. Finally, that even explicitly structured semantic representations degrade under linguistic variation — as documented for multilingual Abstract Meaning Representation [FitzGerald et al. 2024] — motivates examining the simpler triple-extraction target directly, the level at which KG verification operates.

3. Methodology

3.1. Scoping Decision: Simple Binary Relations Only

Every claim expresses a single binary relation corresponding to a direct Wikidata statement — one subject, one predicate, one object. Compound, qualified, n-ary, and multi-hop facts are excluded. *"Marie Curie was born in Warsaw"* maps cleanly to `(Q7186, P19, Q270)`, whereas *"Curie won the Nobel Prize in Physics in 1903"* carries a temporal qualifier and *"Curie was born in the country whose capital is Warsaw"* requires two chained lookups; both are out of scope. Single-step referential indirection — a referring expression resolvable through one property lookup, such as *"the largest city in Masovian Voivodeship"* — remains in scope and is itself one of the linguistic categories under study. We restrict our scope to nine direct, single-hop Wikidata properties: birthPlace (P19), deathPlace (P20), citizenship (P27), educatedAt (P69), occupation (P106), officialLanguage (P37), locatedIn (P131), authorOf (P800), and memberOf (P463).

3.2. Dataset Design: Fact-Centric, Within-Fact Structure

This is a within-fact rather than between-fact study [Warstadt et al. 2020, Ribeiro et al. 2020]. Ground-truth facts are constructed first, then one claim per

linguistic category is generated for each fact, so the same fact appears across every condition and the only variable is linguistic surface form. This controls for fact difficulty as a confound: worse performance on negation could otherwise reflect harder facts, whereas here the same `(Marie_Curie, P19, Warsaw)` appears in simple, negated, coreferential, and coordinated forms, so extraction differences must come from the surface. Because claims sharing a fact are not independent, the statistical unit of replication is *fact*, not claim, and per-category accuracies are computed across facts; given a small fact count, the study is descriptive — we report counts (and percentages only where per-cell counts are large enough) and avoid inferential tests at the cell level.

Facts were sampled semi-automatically with a purpose-built Wikidata script that took a target property (one of the nine PIDs) and a prominence tier, and returned candidate subjects whose sitelink count fell within that tier and that held a statement for the property. Prominence tiers were operationalized by sitelink-count thresholds: high (> 100 sitelinks), medium (10–100), and low (< 10). From these candidates, facts were manually reviewed and selected for balanced coverage across properties and tiers, yielding 26 facts, so neither property coverage nor prominence tier reflects selection bias and the low-prominence tier contains genuinely obscure entities rather than hand-picked ones.

3.2.1. Fact and Claim Schema

A **Fact** stores the ground-truth triple and the material for its negated variants: subject QID, relation PID, true `object` QID, a pre-selected `negation_object` (a plausible-but-false counterpart), and a prominence `tier`. A **Claim** stores its `sentence`, its `linguistic_category`, a foreign key to its parent fact, and a boolean `is_true`. Crucially, `is_true` records the *expected SPARQL `ASK` result on the correctly-extracted triple*, not the sentence's semantic truth. For *"Marie Curie was born in Warsaw,"* the faithful triple `(Marie_Curie, P19, Warsaw)` yields `ASK = true`, so `is_true = true`. For *"Marie Curie was not born in Rio de Janeiro,"* the faithful triple `(Marie_Curie, P19, Rio_de_Janeiro)` yields `ASK = false`, so `is_true = false` — even though the sentence is itself semantically true. The three negation categories all build their claims around the same `negation_object`, so a single such entity per fact suffices for the whole negation block.

3.3. Linguistic Taxonomy

We define 8 linguistic categories, each posing a distinct surface-form challenge for triple extraction while encoding the same underlying fact. Affirmative categories use the fact's `object`; the three negation categories use its `negation_object`. Table 1 gives an example and the challenge for each.

Under the faithful-extraction principle, the extractor reproduces referring expressions verbatim rather than resolving them. Coreference and referential indirection therefore yield surface strings that are by construction not Wikidata-matchable, so these two categories are expected to fail entity matching by design [Jurafsky and Martin 2024] and are analyzed on the failure-mode axis rather than by accuracy (Section 3.4.3). Coreference uses a two-part structure whose introductory sentence expresses no relation in the closed property set, forcing a faithful extractor onto the second sentence and its pronoun

Table 1. Linguistic categories used in dataset construction

Category	Example	Extraction challenge
Simple (baseline)	Marie Curie was born in Warsaw.	None — baseline condition
Negation — sentential	Marie Curie was not born in Rio de Janeiro.	Polarity inversion
Negation — exceptor	Marie Curie was born in a city other than Rio de Janeiro.	Exclusion expression
Negation — implicit lexical	Historians deny that Marie Curie was born in Rio de Janeiro.	Negation via lexical verb
Hedging / modality	Marie Curie is believed to have been born in Warsaw.	Epistemic embedding
Coreference	Von Neumann was widely admired by his contemporaries. He died in Washington D.C.	Cross-sentence referent resolution
Coordination	Marie Curie, who twice won the Nobel Prize, was born in Warsaw.	Embedded modifier
Referential indirection	Marie Curie was born in the largest city in Masovian Voivodeship.	Single-hop resolution required

or definite-description subject; we record both the compliant unmatchable extraction (the structural finding) and the non-compliant silent resolution (an instruction violation, not a correct verification). Coordination applies the same closed-vocabulary forcing through an out-of-set appositive, but because its subject and object surface as canonical strings it is a matchable-surface category scored by accuracy.

Three restrictions follow from the binary-relation scope. First, of the negation subcategories in the full taxonomy we retain three — sentential, exceptor, and implicit lexical. Quantifier-based negation requires set-level propositions, and antonym-based negation a second proposition plus a natural antonym [Petcu et al. 2025]; both violate the binary-relation scope. Affixal negation is excluded for a different reason: lacking a gentilic for most named entities (*"non-Rio-de-Janeiran"* is ill-formed), it forces the negated target up to a containing entity (e.g., the country, *"non-Brazilian"*), so it negates a *different entity at a different ontological tier* and bundles a morphological-decomposition subtask — a non-comparable cell in a within-fact design. Second, passive voice is excluded, since a genuine active/passive alternation presupposes an agentive predicate whereas the in-scope relations are predominantly stative (*"was born in"* has no active counterpart). These exclusions are reportable: substantial parts of natural-language negation and voice resist clean reduction to atomic triples over stative relations.

3.4. Pipeline Architecture

The pipeline has three stages — faithful extraction, entity matching, and relation verification — mapping a claim to a per-claim correctness measure (whether the SPARQL

ASK on the extracted triple matches the claim's expected is_true) paired with a two-dimensional diagnostic record (linguistic phenomenon × structural failure point). The primary extraction model runs at temperature 0 with settings fixed before the full run; because the endpoint is not fully deterministic at temperature 0, each claim is extracted in five independent passes and the majority triple retained.

For a bounded confirmatory comparison, we also evaluate claude-haiku-4-5-20251001 (Claude Haiku 4.5) on four conditions: simple, sentential negation, exceptor-based negation, and coreference. The confirmatory run uses the identical prompt, closed relation vocabulary, entity matching, SPARQL ASK construction, and five-pass majority-vote procedure as the primary run. It tests the portability of the exceptor finding rather than supplying a baseline (Figure 1).

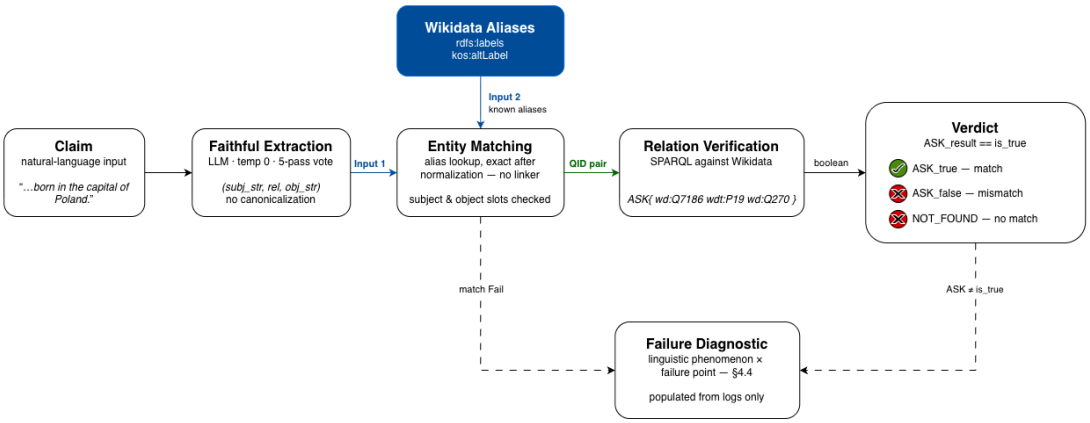


Figure 1. Pipeline architecture overview.

3.4.1. Faithful Extraction

The extraction stage maps each claim to a single (subject, relation, object) triple, reproducing entity expressions exactly as they appear — if a claim contains *"she"* the extracted subject is *"she"*, not *"Marie Curie"*. Silent canonicalisation would conflate two separate capabilities — parsing argument structure and entity linking — and destroy the failure categories under study. Faithful extraction lets the symbolic layer diagnose mismatches visibly.

The principle applies without exception. Even negation cues (*not*, *other than*, *deny*) receive no special handling: the literal entity in the negated position is what the sentence is about and what the extractor produces. The negation does not change which triple is built, only what SPARQL is expected to return, which is already encoded in is_true.

An asymmetry governs the prompt: relations are constrained, entities are not. Relations are drawn from the closed set of nine properties defined in Section 3.1 (the Scoping Decision above); supplying this vocabulary prevents arbitrary paraphrases (*"born in"* vs. *"origin of"*) that would require a normalization step orthogonal to the linguistic questions

at stake. The closed vocabulary also plays a structural role in coreference and coordination: distractor sentences and appositives predicate only relations *outside* the closed set, forcing the LLM onto the target relation. Entity strings are extracted freely as they appear.

3.4.2. Entity Matching

Entity matching is minimal. Because every claim was constructed from a known Wikidata triple, the target QID is known a priori, so matching asks only: *is the extracted string compatible with the known aliases of the target QID?* The pipeline fetches the target's `rdfs:label` and `skos:altLabel` and compares under a fixed normalization policy (NFKC, lowercasing, whitespace collapse, punctuation stripping, symmetric accent folding); the match is exact after normalization, with no edit-distance tolerance that would reintroduce the opacity the design avoids. The object string is matched against the aliases of *both* the fact's `object` and its `negation_object`, recording which slot matched; this single mechanism verifies affirmative and negation claims without a separate polarity branch, and the recorded slot is what later exposes the negation-substitution error. We avoid state-of-the-art entity linkers such as BLINK [Wu et al. 2020], or GENRE [Cao et al. 2021]: such a linker would resolve *"she"* via context and map *"the Polish-born physicist"* to `Q7186` via embeddings, papering over the difficulties this work measures. When a match fails, it is categorized along the two-dimensional error taxonomy (Section 4.4).

3.4.3. Relation Verification

Once both entities match, the pipeline issues a SPARQL query against Wikidata:

```
ASK { wd:<subject_qid> wdt:<property_pid> wd:<object_qid> . }
```

The alias fetch and `ASK` query are each issued once and cached, with failed network calls retried under exponential backoff. Per-claim correctness is the single comparison `ASK_result == is_true`: no SUPPORTED/CONTRADICTED layer and no polarity inference, only whether the round trip from sentence to triple to SPARQL produced the expected boolean. A deployed fact-checker would additionally translate `ASK` into a verdict using the claim's negation status (available from `linguistic_category` or the matched slot), but that step is downstream of the failures studied here. When the comparison fails, the error localizes to exactly one stage — extraction (subject, predicate, or object), entity matching (with a diagnostic flag), or relation mapping — making the two-dimensional matrix populable directly from logs without manual annotation.

Metric scope and analysis families. For the *matchable-surface* categories (simple, the three negations, hedging/modality, coordination), `ASK_result == is_true` measures whether the extractor identified the correct argument structure and whether that triple round-trips. It does *not* measure negation comprehension: since the extracted triple is polarity-free and the expected boolean already encodes the negation, a model that processes a negation cue and one that ignores it both extract the `negation_object`,

Table 2. Round-trip accuracy on the matchable-surface conditions: the extracted triple resolves to a gold-matching SPARQL `ASK` whose result equals `is_true`. gpt-4o-mini covers all conditions; Claude Haiku 4.5 is a confirmatory comparison on the three matchable conditions shown. Dashes indicate conditions not run for Claude. The metric measures argument-structure extraction, not negation comprehension (Section 3).

Condition	gpt-4o-mini	Claude Haiku 4.5
Simple	26/26 (100.0%)	25/26 (96.2%)
Negation – sentential	26/26 (100.0%)	15/26 (57.7%)
Coordination	26/26 (100.0%)	–
Hedging / modality	26/26 (100.0%)	–
Negation – implicit lexical	24/26 (92.3%)	–
Negation – exceptor	12/26 (46.2%)	11/26 (42.3%)
Matchable subtotal	140/156 (89.7%)	–

obtain `ASK = false`, and score correct. The sole observable signature of negation-processing failure is the *negation-substitution error* — the extractor emits the true `object` instead of the `negation_object` — which is why that subtype is featured in the error analysis. The two *referential-gap* categories (coreference, referential indirection) are reported separately: under strict faithful extraction they are *expected* to fail matching, so an accuracy figure there would largely measure the silent-resolution rate rather than anything linguistic, and they are reported on the distribution of match-failure types instead. Reporting both families under one undifferentiated accuracy column would conflate two distinct measurements.

4. Results

We evaluate primarily gpt-4o-mini model (temperature 0; five independent passes per claim, per-claim majority vote), with a confirmatory run with Claude Haiku 4.5, over a within-fact dataset of 26 facts and eight linguistic conditions, yielding 208 claims. As in Section 3, the unit of replication is the fact and we report descriptive counts rather than inferential tests. Following the metric-disaggregation principle of Section 3, we report two metric families separately: round-trip accuracy (`ASK = is_true`) for the *matchable-surface* conditions, and, for the *referential-gap* conditions where strict faithful extraction guarantees a match failure by construction, the distribution of match-failure types and the silent-resolution rate.

4.1. Extraction accuracy on matchable-surface conditions

Table 2 reports round-trip accuracy for the six matchable-surface conditions for gpt-4o-mini and, for the simple, sentential-negation, and exceptor conditions, the confirmatory Claude Haiku 4.5 run. Five cluster near ceiling (92.3–100%); the single sharp exception is exceptor-based negation, at 46.2%. Because the aggregate masks this contrast, we do not foreground a pooled figure. The near-ceiling result for sentential and implicit negation is expected under the metric semantics: a faithful extractor reproduces the negated entity, the `ASK` returns `false`, matching the expected `is_true = false`.

All four ceiling conditions leave both arguments in their canonical subject and

object slots, so faithful extraction recovers them near-automatically. Simple claims establish the baseline; sentential negation inverts the expected boolean without displacing the object; coordination buries its relative-clause predicate outside the closed vocabulary, leaving the main-clause arguments undisturbed; and hedging adds an epistemic layer over the proposition without moving either argument.

4.2. The exceptor construction: a localized object error

For gpt-4o-mini, exceptor-based negation is the only matchable-surface condition to fail at scale. The confirmatory comparison shows that the construction is also difficult for Claude Haiku 4.5, although the two models' failure forms differ. For "*[S] was P-related to a [HEAD] other than [ENTITY]*," the extracted object was the bare *[HEAD]* term (*city*, *university*, *language*, *organization*), which matches no QID alias and yields `NOT_FOUND`; the named entity, buried in the "*other than*" modifier, was systematically passed over. Because each fact is recovered correctly in its simple and sentential conditions, the failure isolates to the exceptor surface form rather than to fact difficulty.

Claude Haiku 4.5 obtained 11/26 (42.3%) on exceptor-based negation, close to gpt-4o-mini's 12/26 (46.2%). The aggregate rates, however, conceal different error forms: gpt-4o-mini produced a bare head noun (e.g., *city* or *university*) in 14/26 exceptor claims, compared with 0/26 for Claude. Claude instead returned a null triple in 15/26 cases, compared with 0/26 for gpt-4o-mini. The exceptor construction is therefore difficult across both models, but the head-noun-substitution signature is specific to gpt-4o-mini.

Claude's sentential-negation control also requires a narrower interpretation: it achieved 15/26 (57.7%) accuracy, and all eleven failures were abstentions rather than mis-extractions — ten null triples and one subject-only extraction, none yielding a verifiable triple and none substituting the negated entity's true-valued counterpart. Its exceptor score is consequently partly depressed by a general negation-related abstention tendency rather than solely by exceptor semantics.

4.3. Referential-gap conditions

For coreference and referential indirection, strict faithful extraction predicts match failure by construction, and the results confirm this with a sharp asymmetry in instruction compliance (Table 3). For referential indirection, the model reproduced the indirect description in every case (e.g., *the largest country in South America*) and never canonicalized it; all 26 claims failed matching, with a silent-resolution rate of zero. For coreference, it reproduced the pronoun or description faithfully in 20 of 26 cases but violated the instruction in the remaining 6 (23.1%), silently resolving the referent. Claude Haiku 4.5 showed the same tendency more strongly, silently resolving the referent in 10 of 26 cases (38.5%). The contrast is itself a finding: single-step coreference occasionally tempts the model to inject world knowledge, whereas multi-hop referring expressions never do. Both support the same conclusion — referentially underdetermined expressions resist canonicalization against a knowledge graph — and the zero rate on referential indirection indicates this resistance is near-total once resolution requires more than a local antecedent.

4.4. Two-dimensional error matrix

Crossing the linguistic condition (why a claim failed) with the structural failure point (what broke in the triple), three observations stand out. First, the failures are not diffuse:

Table 3. Referential-gap conditions. Accuracy is not the operative metric; we report faithful match-failures (the structural finding) against silent resolution of the referent (an instruction violation). Coreference is reported for both models; referential indirection was run for gpt-4o-mini only.

Condition	Model	Faithful match-fail	Silent res. (violation)	Other	Silent-res. rate
Coreference	gpt-4o-mini	20 / 26	6 / 26	–	23.1%
	Claude Haiku 4.5	15 / 26	10 / 26	1 / 26	38.5%
Ref. indirection	gpt-4o-mini	26 / 26	0 / 26	–	0.0%

each failing condition does so through a single dominant mechanism — exceptor negation through 14 object errors (head-noun substitution), implicit lexical negation through 2 subject errors (the matrix subject of the reporting verb, *Scholars*, *Linguists*, taken as the triple subject), and the referential-gap conditions through referential mismatch (20 for coreference, 26 for referential indirection). Second, for gpt-4o-mini, three structural failure types never occur: null extractions (missing triple), triples licensed by non-target material (hallucinated triple), and predicate errors — the last a consequence of the closed relation vocabulary. Third, and most consequential for the central hypothesis, the *negation-substitution* subtype (defined in Section 3) did not occur in any of the 78 negation claims. The same subtype was absent from Claude Haiku 4.5 in both its sentential and exceptor negation conditions (0/26 each).

The absence of negation substitution is a substantive negative result: under faithful extraction the model reproduced the negated entity rather than "answering" the negation, across all three negation subcategories. The design was built to make negation-processing failure visible through this subtype, so its non-occurrence indicates that, for this model and these constructions, the neural-to-symbolic step does not confuse a negated mention with its true-valued counterpart.

4.5. Entity prominence

Stratifying the matchable-surface conditions by subject prominence (tiers defined by Wikidata sitelink counts) shows no accuracy gradient favoring well-known entities: high, medium, and low subjects yield 89.6% (43/48), 89.6% (43/48), and 90.0% (54/60) respectively. The flat curve is consistent with failures driven by linguistic surface form rather than entity memorization.

5. Limitations

Constructed claims. The claims are template-instantiated over known Wikidata triples, giving full control over the fact, the `negation_object`, and the `is_true` label, but not the lexical and syntactic variety of naturally-occurring text; whether the patterns generalise to human-written claims is open, and natural-text replication is immediate future work. **Models and settings.** The full eight-condition analysis uses gpt-4o-mini at temperature 0 with five passes resolved by majority vote. Claude Haiku 4.5 is evaluated only on four confirmatory conditions. This comparison shows that exceptor difficulty is not unique to the primary model, but it is neither a human nor a rule-based baseline and cannot establish whether the observed rates reflect general linguistic difficulty, the prompt

and templates, or the limitations of two particular extractors. Claude's negation-related null outputs further limit direct cross-model accuracy comparisons. **English only.** Multilingual extension, including a Portuguese subset, is left to future work; affixal negation in particular is morphology-dependent and results may not transfer across languages. **Small per-cell counts.** With the fact as unit of replication, the design uses 26 facts across 8 categories and 3 tiers; we report counts rather than percentages where per-cell $n < 10$ and run no inferential tests, so the study is descriptive and makes no claim of proportion stability under a larger sample. **Known-QID assumption and single backend.** Because each claim derives from a Wikidata triple, the target QID is known and matching only checks alias compatibility — far simpler than open-domain entity linking, whose failure modes are orthogonal to those studied here. Verification runs solely against Wikidata, whose coverage is uneven across tiers and relations, so some apparent low-tier failures may reflect coverage gaps rather than extraction difficulty.

6. Conclusion

We asked which linguistic phenomena cause failures in the neural-to-symbolic extraction step of KG-based fact verification, and where those failures occur. To separate linguistic difficulty from fact difficulty, we used a within-fact design: 26 ground-truth facts, each expressed across 8 linguistic conditions, verified by a minimal pipeline — faithful (non-canonicalising) extraction, alias-based matching against a known QID, and a single SPARQL `ASK` against an expected boolean — with a two-metric discipline that keeps matchable-surface and referential-gap phenomena from being averaged.

For gpt-4o-mini, canonical argument structure is highly reliable under this protocol, while failures localise to specific constructions: exceptor phrases often trigger an object error, with the head noun extracted in place of the named entity; implicit lexical negation can trigger an error on the matrix reporting clause; and referentially underdetermined expressions resist canonicalisation by construction. Claude Haiku 4.5 confirms that the exceptor construction is difficult, but not that gpt-4o-mini's head-noun substitution is a cross-model mechanism: Claude primarily abstains on these cases. Across the tested negation conditions, neither model substituted the true object for the negated entity. This supports a narrow conclusion about the observed extraction behaviours, not a general claim that models comprehend negation.

Beyond the specific findings, the contribution is methodological — a within-fact construction that isolates the linguistic variable, a two-dimensional error taxonomy populable from pipeline logs, and an explicit account of what `ASK` = `is_true` can and cannot measure. Next steps follow from the limitations: natural-text replication, additional extraction models, a multilingual extension, and open-domain entity linking.

References

Agrawal, G., Kumarage, T., Alghamdi, Z., and Liu, H. (2024). Can knowledge graphs reduce hallucinations in LLMs? : A survey. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3947–3960. Association for Computational Linguistics.

Cao, N. D., Izacard, G., Riedel, S., and Petroni, F. (2021). Autoregressive entity retrieval.

FitzGerald, J. et al. (2024). MASSIVE multilingual abstract meaning representation: A dataset and baselines for hallucination detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*.

Herron, S. et al. (2025). On the potential of logic and reasoning in neurosymbolic systems using OWL-based knowledge graphs. *SAGE Open Computer Science*.

Jurafsky, D. and Martin, J. H. (2024). *Speech and Language Processing*. 3rd (draft) edition. Available at `https://web.stanford.edu/~jurafsky/slp3/`.

Lavrinovics, E., Biswas, R., Bjerva, J., and Hose, K. (2025). Knowledge graphs, large language models, and hallucinations: An nlp perspective. *Journal of Web Semantics*, 85:100844.

Opitz, J., Wein, S., and Schneider, N. (2025). Natural language processing relies on linguistics. *Computational Linguistics*, 51(3):1009–1032.

Petcu, R., Bhargav, S., de Rijke, M., and Kanoulas, E. (2025). A comprehensive taxonomy of negation for NLP and neural retrievers. *https://aclanthology.org/2025.findings-emnlp.839/*, pages 15511–15533.

Rani, M., Mishra, B. K., and Thakker, D. (2025). Advancing symbolic integration in large language models: Beyond conventional neurosymbolic ai.

Ribeiro, M. T., Wu, T., Guestrin, C., and Singh, S. (2020). Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912. Association for Computational Linguistics.

Sansford, H. et al. (2024). GraphEval: A knowledge-graph based LLM hallucination evaluation framework. *arXiv preprint arXiv:2407.10793*.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Warstadt, A., Parrish, A., Liu, H., Mohananey, A., Peng, W., Wang, S.-F., and Bowman, S. R. (2020). BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Wu, L., Petroni, F., Josifoski, M., Riedel, S., and Zettlemoyer, L. (2020). Scalable zero-shot entity linking with dense entity retrieval. In Webber, B., Cohn, T., He, Y., and Liu, Y., editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.