



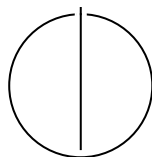
SCHOOL OF COMPUTATION, INFORMATION
AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

**From Predicting Human Ratings to Measuring
True Quality: LLM Accuracy and Bias in
Aspect-Based Sentiment Analysis**

Victor Friedrich





SCHOOL OF COMPUTATION, INFORMATION AND
TECHNOLOGY — INFORMATICS

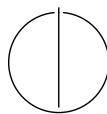
TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

**From Predicting Human Ratings to Measuring True
Quality: LLM Accuracy and Bias in Aspect-Based
Sentiment Analysis**

**Von der Bewertungsvorhersage zur Qualitätsmessung: Genauigkeit
und Fehlerstruktur von LLMs bei der aspektbezogenen
Bewertungsanalyse**

Author:	Victor Friedrich
Examiner:	Dr.-Ing. Jörg Ott
Supervisor:	Ashmi Banerjee
Submission Date:	04.05.2026



I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, 04.05.2026

Victor Friedrich

Acknowledgements

I would like to thank my advisor for their support during this thesis and for granting me the creative freedom to pursue a project of my own proposal. Furthermore I am grateful for the support and encouragement of my family and friends during my studies.

Use of Generative AI Tools

Generative AI tools, including large language models, were used throughout the preparation of this thesis as assistive tools. All AI-generated outputs were reviewed, verified, and revised by the author. The selection of sources, methodological decisions, implementation, analysis, interpretation of results, and final text are the responsibility of the author. All cited sources were checked independently.

Abstract

This thesis asks how numeric scores produced by LLMs should be evaluated when the human scores used for comparison are themselves noisy and unreliable. The empirical setting is Hostelworld, where users rate several hotel aspects, including location, cleanliness, facilities, and security, which are then averaged into an overall score used to rank properties. These human ratings appear strongly shaped by reviewers’ general impression of a property: about 40% of reviewers give the same score to every aspect, and the average correlation between aspects is 0.73. As a result, the seven aspect scores often behave more like one general rating than like seven separate evaluations.

Using review text from 342 properties, the study tests whether LLMs can produce aspect scores that separate the hotel aspects more clearly than users’ own numeric ratings. It also tests three design choices that artificially inflate the correlations between aspects: asking the LLM to score several aspects in a single response, changing the wording of the scoring instruction, and showing the LLM the full list of rated aspects through the required output format.

The results show that all LLM configurations produce substantially lower correlations between aspects than the original human ratings, and that this reduction is structured rather than arbitrary. It follows the same pattern found among reviewers who do not rate all aspects uniformly: location separates most strongly from the other aspects, subjective aspects separate more moderately, and cleanliness, facilities, and security remain the most closely related group. Revealing the full set of aspects in the output format increases correlations between aspects even when each aspect is scored in a separate call, which indicates that the output format is not a neutral technical detail but part of the measurement itself.

A second model, DeepSeek V3.2, reproduces the main construct validity findings but its variation across reruns is too large to support fine-grained comparisons between scoring designs. In an additional sampling-reliability experiment, the scoring is repeated with randomly drawn smaller samples of reviews, to test the stability of LLM-derived scores compared to the human ratings. We find that the construct-validity advantage of LLM scoring holds only where review volume is sufficient and the aspect is frequently discussed. Finally, we evaluate which scoring method produces more fine-grained ratings. Although LLM scores have a wider spread than human ratings, both Claude and DeepSeek anchor on a small set of preferred digits, reducing their effective output resolution.

Overall, the thesis presents two main findings. First, LLMs can provide more independent aspect scores than humans when their ratings are mostly shaped by a single overall impression. This advantage is limited, however, by reduced stability under low review counts and by lower numerical precision. Second, an LLM judge’s awareness of the broader evaluation framework biases the scores it assigns, even when each aspect is scored separately.

Kurzfassung

Diese Arbeit geht der Frage nach, wie sich numerische Bewertungen von Large Language Models sinnvoll evaluieren lassen, wenn die menschlichen Referenzwerte selbst unzuverlässig sind. Als Kontext dient Hostelworld: Dort bewerten Nutzerinnen und Nutzer verschiedene Aspekte einer Unterkunft – darunter Lage, Sauberkeit, Ausstattung und Sicherheit –, die zu einer Gesamtbewertung verrechnet werden und als Rankinggrundlage dienen. Diese Bewertungen werden stark vom allgemeinen Eindruck geprägt: Rund 40 % der Bewertenden vergeben für sämtliche Aspekte dieselbe Punktzahl, und die durchschnittliche Korrelation zwischen den Aspekten liegt bei 0,73. Die sieben Aspektbewertungen verhalten sich somit eher wie eine einzige Gesamtbewertung als wie eigenständige Messungen.

Auf Basis von Bewertungstexten zu 342 Unterkünften wird untersucht, ob Aspektbewertungen von LLMs die einzelnen Hotelaspekte differenzierter abbilden. Geprüft werden zudem drei Designentscheidungen, die die Korrelationen zwischen den LLM-Bewertungen beeinflussen: mehrere Aspekte in einer einzigen Anfrage bewerten zu lassen, die Bewertungsanweisung zu verändern sowie dem Modell über das geforderte Ausgabeformat die zu bewertenden Aspekte offenzulegen.

Alle untersuchten LLM-Konfigurationen erzeugen deutlich geringere Korrelationen zwischen den Aspekten als die menschlichen Bewertungen, und dies folgt dem menschlichen Muster aus numerischen Bewertungen: Lage ist dekorreliert mit allen anderen Aspekten, subjektive Aspektkategorien sind moderat korreliert, während Sauberkeit, Ausstattung und Sicherheit stark korreliert bleiben. Wird dem Modell die vollständige Aspektliste über das Ausgabeformat offengelegt, steigen die Korrelationen selbst dann, wenn jeder Aspekt separat bewertet wird.

DeepSeek V3.2 bestätigt die zentralen Ergebnisse zur Konstruktvalidität, das Modell schwankt über wiederholte Durchläufe jedoch zu stark für Vergleiche zwischen Bewertungsdesigns. Ein ergänzendes Experiment zur Stichprobenzuverlässigkeit zeigt, dass der Vorteil der LLM-Bewertung nur erhalten bleibt, wo genügend Reviews vorliegen und der jeweilige Aspekt häufig erwähnt wird. Zwar haben LLM-Bewertungen eine größere Standardabweichung als die Menschlichen, doch sowohl Claude als auch DeepSeek konzentrieren sich in ihren Ausgabewerten auf wenige bevorzugte Ziffern, was ihre effektive Resolution einschränkt.

Insgesamt zeigt die Arbeit zwei zentrale Ergebnisse. Erstens können LLMs unabhängige Aspektbewertungen liefern als Menschen — deren Urteile überwiegend von einem einzigen Gesamteindruck geprägt sind. Sie sind jedoch begrenzt durch eine geringere Stabilität bei kleinen Reviewzahlen und durch niedrigere numerische Präzision. Zweitens, LLM-Judges bewerten Aspekte weniger unabhängig, wenn sie Kenntnis von den bewerteten Aspekten haben.

Contents

Acknowledgements	iii
Abstract	v
Kurzfassung	vi
1. Introduction	1
1.1. Problem Statement and Scope	1
1.2. Research Questions	2
1.3. Structure	2
2. Literature Review	4
2.1. Aspect-Based Rating Prediction	4
2.2. LLMs for Rating Prediction and Evaluation	5
2.2.1. The LLM-as-a-Judge Paradigm	6
2.2.2. LLMs for Numeric Prediction in Hospitality	6
2.2.3. Implications for This Thesis	8
2.3. Measurement Theory: Key Concepts	8
2.4. Measurement Properties of Human Ratings	9
2.4.1. Reliability	9
2.4.2. Construct Validity	9
2.4.3. Discriminative Power	9
2.5. Measurement Properties of LLM-Based Scoring	10
2.5.1. Reliability	10
2.5.2. Construct Validity	10
2.5.3. Discriminative Power	11
3. Implementation	12
3.1. Design Logic	12
3.2. Score Dimensions and Score Scale	13
3.3. Experimental Conditions	14
3.4. The Schema-Leak Mechanism	15
3.5. Review Sampling	16
3.6. Prompt Caching and Cost	16
3.7. Runtime Controls and Reproducibility	17
3.8. Platform Selection and Data Source	17
3.8.1. Why Hostelworld	17

3.8.2. Data Acquisition	18
4. Construct Validity of LLM Aspect Ratings	19
4.1. Agreement with Human Ratings	19
4.2. Halo Effects in the Human Rating Baseline	21
4.2.1. Magnitude of the Halo Effect	21
4.2.2. Identical Aspect Scores and the No-Flat Benchmark	21
4.2.3. Selective Correlation Across Aspect Pairs	22
4.2.4. Pipeline Design Affects Halo	23
4.3. The Reduction Is Structured, Not Noisy	25
4.3.1. Aggregate Trajectory Alignment	25
4.4. Within-Hotel Profile Recovery	26
4.4.1. Centred Profile Agreement	27
4.5. Summary	27
5. Cross-Model Robustness and Operational Stability	29
5.1. Construct Validity Replication	29
5.2. Rerun Noise and Ablation Interpretability	31
5.3. Null Emissions as Operational Failures	32
5.4. Reduced Instruction-Following	33
5.5. Effective Inference Cost	33
5.6. Summary	35
6. Reliability Under Reduced Review Evidence	37
6.1. From Rerun Stability to Sampling Reliability	37
6.2. Review Subsampling Design	37
6.3. Reference Scores and Reliability Metrics	38
6.4. Convergence to the Large-Sample Reference	39
6.5. Aspect-Level Reliability and Evidence Sparsity	40
6.6. Summary	41
7. Informativeness of Scores	42
7.1. Digit Anchoring in LLM Score Outputs	42
7.2. Comparing Variants Formally	43
7.2.1. Qualitative Analysis	43
7.2.2. Last-Digit Entropy	44
7.2.3. Full Value Entropy	44
7.3. Summary	45
8. Conclusion	46
8.1. Limitations	47
8.2. Implications	47
8.3. Future Work	48

A. Appendix	49
A.1. Search Queries	49
A.2. Schema	49
List of Figures	51
List of Tables	52
Bibliography	53

1. Introduction

Two guests can stay at the same hostel on the same night and reach opposite conclusions about it. One leaves satisfied because the bar downstairs ran late and the common area was full of people; the other leaves frustrated for the same reason. Neither evaluation is wrong, but a single overall rating that averages them produces a number whose meaning is unclear. Aspect ratings are intended to preserve what the overall average loses: they ask reviewers to score specific parts of the stay, such as cleanliness, location, staff, and atmosphere.

This matters most for travellers who are unlikely to engage carefully with review text. A backpacker booking a few days ahead, often while already travelling, may have limited motivation to read through hundreds of reviews.

In practice, the feature falls short of what it promises. Reviewers do not evaluate aspects independently. Their overall impression of a property tends to drive most of the individual scores they assign, a pattern long documented in survey research as the halo effect. On Hostelworld, the platform examined in this thesis, roughly 40% of reviewers give identical scores to every aspect, and the seven aspects correlate at an average of 0.73 across properties. The consequence is that aspect ratings end up communicating broad sentiment about a property rather than separable information about its individual qualities.

This becomes an issue when travellers compare properties. Two properties may have the same cleanliness score, but in one case cleanliness may be the property's weakest aspect, while in the other it may be its strongest. Under the halo effect, the second is more likely to actually be clean, but this is not recognisable from the absolute score values.

1.1. Problem Statement and Scope

A possible solution is to predict aspect quality directly from review text, where guests describe their experience in a less constrained form than a numeric scale permits. This idea is not new. Aspect-based sentiment analysis has been studied extensively in NLP, but the model was typically evaluated by how well it reproduces the score a user provided. This thesis changes that target. If user ratings are noisy, compressed, and halo-contaminated, then reproducing them is not necessarily the same as measuring aspect quality.

This thesis therefore treats numeric aspect ratings and text-derived LLM scores as two competing measurement instruments. The comparison is not whether an LLM can predict user scores, but whether scores derived from review text behave like a better measure of aspect quality than the numeric ratings they are usually benchmarked against.

Two scenarios are possible. In the first, numeric ratings remain the better instrument. They force reviewers to evaluate every aspect, while written reviews do not. A guest may rate

an aspect poorly without mentioning it, or leave broadly positive feedback that does not distinguish between a merely good stay and an exceptional one. In these cases, the numeric score contains information that the text leaves out.

In the second scenario, review text provides the better basis for scoring as it mitigates two issues of numeric scoring: First, the rating scale is typically limited to five values, and most reviewers use the highest two. Second, every score reflects an individual reviewer's standard, and these standards differ. A model reading the underlying text could produce scores at a higher granularity, and with reviews evaluated under a consistent standard.

The claim is hard to test directly. The true quality of a hotel is not observable, and external proxies cannot capture many important nuances. Distance to the city centre is a poor measure of location wherever the centre is not the most desirable area. Crime statistics do not capture what reviewers describe under "security", which often reflects the presence of lockers, key cards, and whether the front desk is staffed at night. Validation against an external ground truth is therefore not realistic.

This thesis therefore tests whether a scoring method reduces known distortions in human ratings, especially halo and scale compression, without losing alignment with property quality. A method that does so can be interpreted as a closer approximation of true quality.

1.2. Research Questions

RQ1. When user ratings are themselves a halo-contaminated measurement instrument, can LLM-based scoring of the same review text produce a more differentiated measure, one that decouples aspects which should plausibly vary independently while preserving correlations among aspects that should plausibly move together?

RQ2. What features of the scoring pipeline determine whether an LLM functions as a discriminating evaluator or as a single-impression rater? In particular, does the model's awareness of the broader evaluation framework induce halo even when aspects are scored in separate inference calls?

RQ3. Where are the limits of LLM-based aspect scoring as a measurement instrument? Specifically: how much review evidence is required for stable scores, and what is the effective resolution of the numeric outputs the model produces?

1.3. Structure

The thesis proceeds as follows. Chapter 2 reviews prior work on aspect-based rating prediction and LLM-based evaluation, and introduces the measurement-theoretic concepts used to compare human ratings and LLM-derived scores. Chapter 3 describes the scoring pipeline. It presents the pipeline not as a neutral implementation step, but as an experimental design that isolates three features which may artificially increase similarity between aspect scores.

Chapter 4 evaluates the construct validity of the Claude-based scores. It first establishes the strength of the halo effect in the human rating baseline, then tests whether LLM-based scoring reduces this effect without producing arbitrary or unstructured score profiles. Chapter 5

examines whether the same findings hold for a second model, and evaluates whether the ablation design remains interpretable when model behaviour is less stable.

Chapters 6 and 7 examine two further properties that affect how LLM-derived scores should be interpreted. Chapter 6 tests their reliability under reduced review evidence, while Chapter 7 analyses the effective resolution of the numeric outputs. Together, these chapters examine the conditions under which LLM-derived aspect scores can be interpreted as a more differentiated measurement of hotel quality than the numeric ratings they are compared against.

2. Literature Review

2.1. Aspect-Based Rating Prediction

Prior literature on aspect-level rating prediction in hospitality shows several recurring patterns, identified in Table 2.1. First, many models do not predict aspect ratings independently from review text, but instead use the overall rating as an anchor and estimate aspect scores as adjustments around that anchor. Second, many studies apply restrictive filtering, keeping only reviews with complete optional aspect ratings or with a minimum review length. This likely results in a sample with more explicit textual evidence, making the task easier than it would be in the broader review population. Third, more recent work within the reviewed set increasingly evaluates prediction at the review level rather than the hotel level, despite the latter being more directly relevant for practical purposes. Finally, the reviewed studies generally treat user-provided numeric ratings as target labels without examining whether these labels may themselves be noisy, biased, or internally inconsistent.

Table 2.1.: Summary of aspect-prediction studies.

Paper	Eval Target	Overall-Rating used as Input?	Key Filtering
Wang et al. [1], LARA	Hotel	Yes	Discards sentences that cannot be assigned to any aspect: 235,793 crawled reviews, 1,850 hotels / 108,891 reviews after preprocessing
Lu et al. [2]	Hotel + Review	Yes, at the hotel level	Hotel Level: ≥ 10 reviews per hotel; removes reviews with < 50 or > 300 words. Review Level: 5,000 reviews with overall stars roughly balanced
Wang and Ester [3], SATM	Hotel	Overall rating distribution	LARA dataset. 7 aspects chosen and required for all reviews; only evaluates reviews with ≥ 3 sentences in a review. Final dataset 1,850 hotels / 61,306 reviews
Xue et al. [4], ARIH/ARID	Review	Yes	4 aspects chosen and required for all reviews; only evaluate reviews with ≥ 10 opinion comments
Wang et al. [5]	Review	Yes	5 aspects chosen and required for all reviews
Laddha and Mukherjee [6]	Review	None	6 aspects; only considers reviews where at least 2 sentences mention an aspect and only evaluates those reviews on that aspect
Hasan and Ding [7]	Review	None for aspect prediction	6 aspects; filters reviews shorter than 10 words

2.2. LLMs for Rating Prediction and Evaluation

Earlier hospitality aspect-prediction research was developed under the constraints of traditional NLP methods, which were most effective when sentiment was stated directly. They were less suited to reviews containing mixed evidence, indirect evaluations, or contextual trade-offs, such as “the room was very clean, but everything else felt neglected” or “the location is far from the centre, yet perfect for the beach and nightlife” [8]. Large language models

make a different use of review text plausible because they can infer aspect quality more holistically from the full context of a review. This shifts the problem from label replication alone toward evaluation: rather than only predicting observed user ratings, the model can be used to generate aspect scores from textual evidence. In this sense, the present thesis connects to the LLM-as-a-judge literature.

2.2.1. The LLM-as-a-Judge Paradigm

The LLM-as-a-judge literature asks whether language models can turn complex text into structured evaluations. Recent surveys document the expanding scope of this paradigm and note that a common way to benchmark LLM judges is to measure alignment with human preferences. [9, 10]. More critical work argues that this validation target can itself be unstable: Chehbouni et al. [11] argue that correlation with human judgement has become a primary validation criterion without sufficient attention to whether the human judgements themselves validly capture the intended construct, while Bavaresco et al. [12] caution that correlation with human judges may not always be the right validation target, since apparent agreement may reflect shared biases or unreliability of human annotators rather than valid capture of the underlying construct.

In this thesis, the model scores each hotel directly from text and produces scores for several aspects without using a reference answer. In the LLM-as-a-judge literature, this is closest to an absolute, single-response, reference-free, multi-criterion setting. The terminology is useful because it links the design to concrete validity risks: calibration problems, weak discriminative power, criterion entanglement, and systematic evaluator bias. Prior work identifies closely related LLM-judge designs as vulnerable to these problems, making an analysis of them important [13, 14, 10, 15].

2.2.2. LLMs for Numeric Prediction in Hospitality

A targeted Scopus and Google Scholar search focused on identifying work most similar to this thesis found 473 records after deduplication, of which 48 were retained after title and abstract screening (Appendix A.1). The most relevant literature falls into three groups: papers that use LLMs to directly predict the overall rating from only the text, papers that use coarse LLM-derived signals combined with other non-textual inputs to make predictions, and other papers that cannot be grouped under one clean label.

Overall rating prediction. The small body of work using LLMs for direct numeric prediction focuses on overall ratings only and generally evaluates success in terms of agreement with observed user ratings. Roumeliotis et al. [16] report 67% exact-match accuracy for GPT-4o on five-class hotel rating classification in a test-set that was balanced to have equal reviews for each rating label; Nnanna et al. [17] perform a similar task and find a Spearman rank correlation of 0.703 between predicted and actual ratings, $MSE = 0.629$, $MAE = 0.451$. Vink et al. [18] appear relevant at first as they present “an LLM fine-tuned on 200 reviews”, though

this is better described as them using an LLM-written conventional supervised pipeline for comparison.

LLM-derived aspect signals. A related set of papers uses LLMs to extract aspect, sentiment, or preference signals from reviews for downstream recommendation, rating-prediction, or rating-explanation tasks. However, these studies generally do not treat LLM-generated aspect scores as standalone measurements; the LLM output is an intermediate feature that is combined with other modelling components to predict or explain overall ratings [19, 20, 21, 22, 23].

In the literature identified through the targeted search (Appendix A.1), no paper uses LLMs to generate aspect-level numeric ratings from review text alone.

LLM-derived signals of social atmosphere in hostels. McMurry [24] provides a closely related but methodologically distinct case. The paper evaluates GPT-4o-mini as a baseline for inferring socialness from hostel review text. This is similar to the evaluation of the (social) “atmosphere” aspect conducted for this thesis, but it differs from the present thesis in its target variable: rather than predicting observed platform ratings, it evaluates performance against manually annotated socialness labels. The paper argues that socialness is a distinct text-grounded construct, while Hostelworld’s atmosphere rating is a different and likely noisier measure because it is shaped more by expectation alignment than socialness.

LLMs as measurement instrument: papers with similar framing. Morini-Marrero et al. [25] compare intra-rater reliability of ChatGPT with three human raters on hotel reviews. ChatGPT matched guests’ ratings slightly less often than humans (73% vs. 75 to 79%) but was more consistent when re-rating disputed reviews one month later. The authors also interpret ChatGPT’s tendency to compress scores toward roughly 4 and 5 as a “more balanced” perspective. An alternative interpretation is that this pattern reflects the score-compression bias documented elsewhere in the LLM-judge literature (Section 2.5.3).

Liu [26] shows in a rubric-conditioned labelling study that LLM labels can be more internally consistent and easier for downstream models to learn from than human labels in subjective text annotation. The same author applies a similar idea to patient medication reviews in a 2026 study [27]. Instead of asking the LLM to assign one overall sentiment label to a review, it first asks the model to judge separate aspects of the review and then combines those aspect judgements. This produces training labels that are more stable and easier for later models to learn from than labels based only on user ratings or on a single overall LLM judgement.

Wang et al. [28] evaluate LLMs as essay raters through a psychometric framework and show that criterion entanglement is a shared problem for human and LLM raters, not unique to either.

2.2.3. Implications for This Thesis

Taken together, the ABSA and LLM-as-a-judge literatures reveal a mismatch between the standard evaluation of hotel aspect prediction and the possibilities opened by LLMs. Hospitality research typically treats user ratings as fixed targets and asks whether a model can reproduce them, while a part of the LLM-as-a-judge literature stresses that agreement with observed labels is not sufficient when the labels themselves may be noisy or biased. Hotel reviews provide a useful setting to test this issue directly, because human and LLM scores can be compared on the same review texts at scale without requiring a separate annotation dataset. The question is therefore not only whether an LLM can reproduce observed aspect ratings, but whether it functions as a more differentiated and less halo-contaminated measurement instrument.

2.3. Measurement Theory: Key Concepts

Comparing both scores requires operating within an established framework that specifies what counts as a better scoring instrument. Without it, any apparent advantage of one over the other could be attributed to the particular metrics chosen for the comparison. This section therefore treats aspect-level scoring as a measurement problem and draws on its concepts to evaluate the two instruments systematically.

Xiao et al. [29] argue that psychometric concepts such as reliability, construct validity, and measurement error can be used to assess how well human raters and LLM-based evaluators score machine-generated text.

Chehbouni et al. [11] apply this measurement-theoretic lens specifically to LLM-based judges, cautioning that their rapid adoption has outpaced rigorous evaluation of their reliability and validity as measurement instruments.

Reliability concerns whether the instrument produces stable results under comparable conditions. Here, this means whether the same property, rated by different reviewers or by the same LLM under different prompt conditions, receives consistent scores [29, 11].

Construct validity concerns whether scores capture the intended construct rather than a different or more general one. Applied to hotel aspect ratings, this means that the instrument should distinguish between aspects such as cleanliness, location, staff, and atmosphere. The scores should not correlate so highly that they no longer provide separable information [30, 31].

Discriminative power asks whether scores spread widely enough to distinguish between the items being rated. An instrument that assigns nearly the same value to everything is reliable in a trivial sense but carries no usable signal. This concept is not one of the main categories used by Xiao et al. [29] or Chehbouni et al. [11], but it is part of the broader psychometric measurement literature. Ferguson [32] and Hankins [33, 34] treat discrimination

as a distinct measurement property because reliability does not by itself guarantee that a measure can separate cases in a useful way.

2.4. Measurement Properties of Human Ratings

This section examines whether human aspect ratings provide a reliable measure of the true underlying property quality. Some of the biases discussed here may also be reflected in the review text itself, and therefore constrain any text-based scoring instrument, including LLMs.

2.4.1. Reliability

Single hospitality ratings contain substantial variance not attributable to stable hotel characteristics. In a multilevel decomposition of 3.49 million TripAdvisor reviews, Radojevic et al. [35] found that hotel attributes explained only about 27% of total rating variance; visitor nationality, trip characteristics, and encounter-level residuals accounted for the rest. [35] Much of this non-hotel variance is structured. Rater nationality and trip type shift scores systematically, so the variance does not cancel through aggregation the way random measurement error would.

Second, ratings are not independent. Cicognani et al. [36] show the effects of social influence in hospitality, highlighting that above-average prior ratings increase hotel ratings whereas below-average prior ratings do not produce a comparable effect.

2.4.2. Construct Validity

In hospitality aspect ratings, the main threat to construct validity is the weak empirical separation of supposedly distinct aspects. Nicolau et al. [37] tested this directly using location as a focal attribute, on the grounds that a hotel's location is fixed and cannot change. In a sample of 21,338 Booking.com hotels, ratings on the other attributes explained nearly 50% of the variance in location scores. In a large-scale study of Qunar hotel reviews, 77% of reviews have an equal score to every aspect, making these informationally equivalent to a single overall score [38]. The remaining 23% who used diversified ratings are the only cases in which aspect-level data adds information beyond the overall rating. Wirtz [39, 40], working in general consumer research, identify the conditions that amplify halo: evaluative purpose, few rated attributes, lower involvement, and delayed measurement. All of these characterise the online review setting.

Multi-aspect ratings may therefore largely replicate a single latent factor.

2.4.3. Discriminative Power

Hospitality ratings often fail to discriminate because most properties are pushed into a narrow, top-heavy range of scores. On Airbnb, roughly half a star separates 95% of all properties [41]; on Booking.com the compression is comparable, on TripAdvisor less extreme [42]. This is not a reflection of uniform quality. When participation is mandatory rather than self-selected,

rating distributions become approximately normal rather than J-shaped [43], suggesting that the skew of online hospitality ratings is driven largely by who chooses to review.

Platform design can amplify the problem: Booking.com’s original system used a 2.5–10 scale presented in a way that most researchers and users interpreted as a conventional 1 to 10 scale. When it later switched to a direct 1–10 rating, scores dropped by 0.15–0.22 points, confirming that scale design rather than quality had caused parts of the prior inflation [44, 45]. This is relevant to our analysis of Hostelworld, where a 2–10 scale is presented as a 1–10 scale and we normalise accordingly in Chapter 3.

2.5. Measurement Properties of LLM-Based Scoring

2.5.1. Reliability

LLM scoring is sensitive to prompt wording. Sclar et al. [46] report that formatting variants changed accuracy by up to 76 percentage points on earlier models; the effect appears smaller on newer models, but it does not disappear [47], and Hwang et al. [48] show that LLM judges remain vulnerable to framing-induced inconsistencies. Zhang et al. [49] find that well-specified human-authored rubrics yield dramatic improvements over self-generated criteria. Beyond prompt sensitivity, LLM judgements do not act with consistent preferences: the SAGE framework [50] shows that even top models fail to maintain consistent preference orderings in nearly a quarter of difficult cases, though human judgements exhibit comparable inconsistency on the same items. Reducing the model’s temperature setting improves self-consistency, but even at temperature 0 the model output is not deterministic [51, 52]. Human ratings are not prompt-dependent in the same mechanical sense, but analogous framing effects like question wording, scale anchoring and question ordering are well documented in survey methodology [53]. The key difference is that this source of instability is easier to test for LLMs: prompt sensitivity can be measured cheaply by running variant prompts, whereas framing effects in human ratings require new surveys.

2.5.2. Construct Validity

Because LLMs generate tokens autoregressively, scoring multiple criteria in a single response means earlier-produced numbers become context for later ones. Stureborg et al. [15] show that attributes evaluated later within the prompt-response correlate less well with expert annotations, and Li et al. [54] find that joint scoring leads to overly similar scores across criteria. This sequential anchoring is mechanistically distinct from halo. It is a generation artefact that can be reduced through prompt design (scoring each aspect independently), but its observable consequence is identical: inflated inter-aspect correlations. An unexamined question is whether the model’s awareness of the broader evaluation framework is itself sufficient to induce entanglement even when aspects are scored in separate inference calls. To our knowledge, no existing study has tested this condition.

2.5.3. Discriminative Power

Rating compression. LLM judges often have weak discriminative power because they compress scores into a narrow upper range of the scale. As a result, they may rank two properties correctly while assigning scores that are too close together to reflect meaningful quality differences [15, 13]. A model can therefore perform well on ranking metrics while still producing scores that are too compressed to distinguish properties in a practically useful way. Post-hoc calibration is a common remedy.[55, 14].

Scale choice. The choice of the correct scale is also important. Lee et al. [56] find that LLM evaluators are not fully inter-scale consistent: numeric interval scales such as 1 to 5 and 1 to 10 align better with each other than Likert or binary formats, but changing the scoring scale can still lead to inconsistent judgements. Stureborg et al. [15] similarly find that increasing granularity does not necessarily improve evaluation quality: 1 to 100 scoring performs worse than 1 to 10 in average rank correlation with expert judgements provided on a 1 to 5 scale.

3. Implementation

The previous chapter established that we want to compare two different types of scores under a measurement theory framework: the numeric aspect ratings that users provide directly, and the LLM scores generated from their corresponding review text. The purpose of the generated scores is to potentially provide a more useful representation of the corresponding hotels.

This chapter specifies how the scoring pipeline is designed. A core consideration is that it should be possible to later identify the source of potential score improvements. Without careful design, it would not be clear whether improvements can be attributed to individual pipeline decisions, or whether they are simply a general property of LLM-based scoring. We therefore design the scoring system not as a technically neutral implementation step, but as an experiment that allows us to identify what details contribute to higher-quality scores.

3.1. Design Logic

The pipeline evaluates scores at the hotel level. For this, individual reviews are aggregated together, resulting in a list of averaged aspect scores and a large text segment that is used by the LLM to generate corresponding aspect ratings. The aggregation serves two purposes. First, it reduces the non-structural noise of individual reviews documented in Section 2.4.1. Second, it allows the model to weigh evidence across reviews and potentially come to a more accurate conclusion than it would be able to find from individual reviews.

The methodological concern that shapes the rest of the chapter is criterion entanglement: the risk that the LLM produces correlated scores across aspects not because the aspects are genuinely related, but because of how the scoring pipeline is designed.

Three mechanisms were identified as the most plausible sources of criterion entanglement.

Co-generation. LLMs generate text autoregressively: each new token is produced based on both the original prompt and the tokens already generated in the same response. When several aspect scores are produced in a single inference call, the model does not generate them independently. Earlier scores remain in the output context and can influence later scores.

Schema awareness. Structured-output schemas are injected into the model’s prompt, so that the model answers in a specific format [57, 58]. If the schema includes the names of all aspects, the model knows the full evaluation framework even when the prompt itself only requires one aspect to be scored. This may cause the model to form a holistic impression of the hotel before scoring.

Instruction framing. The wording of the scoring instruction, for instance asking the model to predict a user’s rating rather than asking it to evaluate the reviews directly, frames the task differently and may shift how the model integrates evidence across aspects [59].

The pipeline is structured as an ablation across these three features. Each successive condition removes one of them while holding the others fixed, so that a change in the correlation structure can be attributed to the specific design choice. Because a reduction in correlation could reflect added noise rather than better construct separation, Chapter 4 evaluates whether the changes are structured rather than arbitrary.

3.2. Scored Dimensions and Score Scale

Each Hostelworld review contains seven standard aspect ratings: security, location, staff, atmosphere, cleanliness, facilities, and value for money. These aspect ratings are the main human benchmark in this thesis. Hostelworld also displays an overall rating used to rank properties, which is calculated as the mean of the seven aspect scores. In the LLM pipeline, the model is asked to generate scores for the same seven aspects as well as for overall, so that the generated scores can be compared directly with the platform ratings.

Overall is treated differently from the seven aspects. It is included in the level and agreement analyses in Chapter 4, because it provides a useful summary comparison between human and LLM scores. However, it is later excluded from correlation analyses, since it is mechanically derived from the aspect ratings and would therefore inflate the observed correlation structure.

The pipeline also includes scoring for three additional dimensions: bathrooms, party, and social. These are not ratings that users provide on Hostelworld, and they are therefore not considered as part of the core analysis. Instead, they are diagnostic probes. Their purpose is to test whether the model distinguishes between concepts that the platform groups together under broader aspect labels. Bathrooms captures an area where cleanliness and facilities may overlap, while party and social separate two meanings often contained within Hostelworld’s atmosphere rating.

Atmosphere requires special clarification within the pipeline prompts. On Hostelworld, atmosphere is generally interpreted as party or social atmosphere rather than as interior design, comfort, or room ambience. The platform interface reinforces this interpretation by displaying the atmosphere category with a beer icon, which human reviewers see but the model does not. For this reason, the prompt includes the following clarification:

atmosphere: generally interpreted as party / social atmosphere.

No other aspect receives a comparable platform-specific note. The risk that this clarification disproportionately shapes the atmosphere score is addressed by the inclusion of the diagnostic probes for party, and social, which are analysed in Chapter 4.

All LLM scores are generated with two decimals on the $[0, 1]$ interval. Human scores are transformed to the same scale before comparison. Hostelworld users rate each aspect from 1 to 5, and the platform displays these ratings as scores from 2 to 10. Since the effective

lower bound of the displayed platform scale is therefore 2 rather than 0, human scores are normalised by mapping 2 to 0 and 10 to 1:

$$x_{\text{normalized}} = \frac{x - 2}{8}.$$

For each property, the human comparison score is the arithmetic mean of the ratings from the same reviews that were included in the model input. This ensures that the human and LLM scores are based on the same textual evidence base.

3.3. Experimental Conditions

The pipeline varies three design features: scoring aspects jointly or separately, revealing the full set of aspects in the output schema, and framing the instruction as either user-rating prediction or direct evaluation.

In **Joint**, the model scores all aspects of a hotel in a single call. This is the most efficient configuration, but also the most vulnerable to co-generation. For reproducibility it is noted that the aspect order is always fixed as security, location, staff, atmosphere, cleanliness, facilities, value_for_money, overall, bathrooms, party, and social. It is not tested whether a different order could result in changes of the intercorrelation pattern. This cannot be ruled out, as Chapter 4 later identifies security as the aspect with the strongest negative bias, meaning that it consistently receives lower ratings from Claude.

In **SchemaLeaks**, the model scores one aspect per call, but the structured-output schema still contains the names of all other aspects that are scored. This removes co-generation while preserving the model’s awareness of the broader evaluation framework. However, this transition also introduces a confound: when aspects are scored in separate calls, generation noise is introduced independently each time, which by itself may reduce correlation. Chapter 4 addresses this by testing whether the reduction is structured rather than arbitrary.

In **Predict**, the model scores one aspect per call with a generic schema that does not reveal the other aspects that are evaluated. The instruction framing is identical to what is used in SchemaLeaks, predicting what score a typical user would assign:

```
You are evaluating {{ property_type }} reviews. Based on the reviews above,
predict what numerical rating a typical user would assign for the aspect {{
target_aspect }}.
```

This removes both co-generation and schema awareness while retaining the framing used in the previous step.

In **Rate**, the model again scores one aspect per call with the generic schema, but is instructed to analyse the reviews and provide a numerical rating directly:

```
Analyze the {{ property_type }} reviews above and provide a numerical rating for
the aspect {{ target_aspect }}.
```

This changes the task from user simulation to direct evaluation.

The distinction between Predict and Rate is subtle but potentially consequential: Predict frames the model as a simulator of human behaviour, including human biases, while Rate positions it as an independent evaluator. This contrast also probes the prompt sensitivity issues described in the literature review.

The ordering of conditions is deliberate. Moving from Joint to SchemaLeaks removes only co-generation. Moving from SchemaLeaks to Predict removes only schema awareness. Moving from Predict to Rate changes only the framing. If inter-aspect correlations are reduced at each step, the removed feature is most likely responsible for it.

The expected ordering from most to least entangled variants is therefore Joint, SchemaLeaks, Predict, and Rate.

3.4. The Schema-Leak Mechanism

Structured output is a mechanism used to make the model return ratings in a fixed JSON format. This is necessary for automated parsing, but it also changes the scoring procedure. The output specification is not always a neutral formatting layer. Depending on the provider, it may either become part of the prompt context or constrain the form of the generated response. In this thesis, schema leakage refers to this possibility that aspect names encoded in the output specification affect the scores themselves.

The mechanism differs between the Claude and DeepSeek pipelines. Claude is accessed through AWS Bedrock. At the time the pipeline was implemented, the standard way to enforce structured output for Claude was to define a tool whose input schema matched the required JSON format and to force the model to call that tool [58]. Native JSON-schema structured output later became available for Claude on Amazon Bedrock, but the Claude results reported here use the earlier tool-use based implementation [60].

For Claude, the mechanism around schema leakage is well-documented. Tool use constructs a system prompt from the tool definitions, and those definitions include the JSON input schema [57, 58]. This means that the schema does not remain an external parser or validator: the aspect names and type constraints become part of the prompt context that Claude receives. If the schema names all evaluated aspects, those aspect names are therefore directly known to the model before it generates the rating. The schema can therefore function as an informational channel, not only as an output-formatting constraint.

DeepSeek is accessed through OpenRouter using `response_format` with `type: "json_schema"` and `strict: true`, which requires the response to match the provided schema [61]. Unlike Claude's forced-tool setup, OpenRouter's public documentation does not specify whether this is achieved through prompt insertion, constrained decoding, provider-side validation, or a combination of mechanisms. DeepSeek's own API documentation describes JSON output

mode, but not the same strict JSON-schema mechanism [62]. For DeepSeek, schema leakage should therefore be interpreted more cautiously. The schema is part of the structured-output request, but it is not clearly documented as prompt content in the same sense as Claude’s tool schema.

3.5. Review Sampling

Properties are included in the inference pipeline only if they have at least 50 reviews, ensuring a sufficient basis for text-based scoring. Chapter 6 evaluates the reliability of the model under lower sampled review budgets. Across the evaluated cities, 85% of properties exceed this threshold of 50 reviews; for the properties that fall below it, half have zero reviews. The filter therefore still keeps the scored hotels representative. Reviews containing fewer than three characters were excluded.

To mitigate the documented tendency of language models to reason less effectively over very long contexts [63, 64], the input is capped at 110,000 characters per property. Reviews are ordered from newest to oldest and included in full; a review is never truncated mid-text. If adding the next review would exceed the budget, it is omitted entirely. In practice, this results in roughly 200 to 400 reviews per property, and about 30% of properties in large cities have aggregated review lengths that match or slightly exceed this limit.

This design could introduce a small recency bias: Section 2.4.1 documents how later reviews are more positive. The LLM first sees the reviews ordered by newest date, and might implicitly be anchored by the impression formed from the most recent reviews.

3.6. Prompt Caching and Cost

Scoring in the Predict or Rate variants requires eleven calls per property over the same review text, increasing input token cost by roughly an order of magnitude. Anthropic offers a 90% discount on cached input tokens, but caching is prefix-based: only the initial part of each request is reused, and it must be identical across calls [65]. The review block is therefore placed first in each request, forming the shared prefix, while the short instruction requesting a rating for a specific aspect is placed at the end. This also constrains the schema design: the output schema, which forms part of the system prompt, must remain constant across calls to preserve the cache. This creates the limitation of using a generic schema that does not name any aspect for Predict and Rate, discussed in Section 3.4.

Typical prompt design requires placing the instruction prompt at the beginning, with the content following it. OpenAI’s prompting guidelines for gpt-4.1 specify that instructions should ideally be placed in both the beginning and the end of the prompt [66]. The pipeline reverses this conventional order and places the review text at the beginning for Predict. To maintain consistency across prompt variants, the same ordering is used for the Joint variant.

Placing instructions after the review text reverses the conventional prompt order. Whether this change affects scoring quality cannot be tested within the current design, since the normal order would break prefix caching and multiply costs accordingly.

3.7. Runtime Controls and Reproducibility

All generation settings not directly related to the pipeline design are held constant across conditions, so that observed differences can be attributed to the pipeline itself.

Experiments use `claude-sonnet-4-6` hosted on AWS Bedrock in the `us-east-1` region, and `deepseek-v3.2` on OpenRouter. Generation settings are fixed across all runs: `temperature = 0.0` and `top_p = 1.0`.

The settings were chosen to maximise reproducibility within the limits of the platform, however they do not guarantee bitwise-identical outputs across runs. Minor non-determinism may remain from floating-point arithmetic in distributed inference. Furthermore, OpenRouter is not used with a fixed provider, which may introduce additional variation in the DeepSeek scores.

Schema definitions are provided in the appendix.

3.8. Platform Selection and Data Source

This section explains why Hostelworld was chosen and describes the scraped corpus. While many platforms use aspect-based ratings, the choice of platform is an important methodological decision for the measurement-theoretic comparison, as platform decisions can amplify the issues that make numeric scores less useful.

Five properties of Hostelworld make it well suited for the measurement-theoretic analysis developed in this thesis.

3.8.1. Why Hostelworld

Incentivised halo effects. Hostelworld requires reviewers to provide a numeric score for each of the seven aspects before a text-based comment can be written. Reviewers who only want to leave a comment must therefore still rate aspects numerically, even when they have not formed a view on each category. For these reviewers, uniform scoring is the path of least resistance. A second interface decision further contributes to the high halo effects seen within aspect ratings: the city-level overview displays only the aggregate overall score, computed as the mean of the seven aspects. A reviewer who wants to reward or penalise a property and affect its rankings therefore has a structural incentive to assign a non-representative score to each aspect, regardless of whether the textual evidence supports a uniform rating. Both effects work in the same direction and together contribute to the strong halo in the human baseline. This is desirable for the present study: the more halo-contaminated the numeric baseline, the larger the gap a text-grounded instrument can plausibly close, and the more informative the comparison becomes.

Detailed metadata. Every review carries significant information about the reviewer profiles, allowing for potentially more detailed insights into how demographic factors affect either the

numeric or the text-derived scores. Concretely, each review has information about nationality, gender, age range, the reviewer’s review count, and the reviewer’s trip type.

Homogeneous userbase. Hostelworld provides a more homogeneous userbase compared to generalised hospitality platforms. On TripAdvisor or Booking, cities like Barcelona might disproportionally contain properties rated by leisure travellers, while properties in New York could more often be rated by business travellers. In contrast, hostels predominantly serve budget and backpacker travel, which results in more consistent standards and writing styles. The evaluation of the scraped data shows that the traveller profile is very consistent: 92% are solo travellers. Of this group, 54% are female and 46% male. Travellers mostly fall within two age-groups 18–24 (40%) or 25–30 (37%).

High review counts. Hostels typically have a high number of reviews, which allows for a rich textual signal without filtering the dataset to a non-representative subset of its entries. 53% of scraped properties have at least 50 reviews, 27% have at least 100, and 257 properties exceed 500. Comparable public review corpora are not always structured in a way that contain many cities with many properties at similar review depths, which complicates city-level analysis at scale.

Interpretability of city-specific patterns. Existing public review datasets such as the LARA Tripadvisor corpus are typically skewed toward a small number of major destinations, which makes city-level comparisons sensitive to which markets happen to be well represented. Hostelworld’s distribution is comparatively flat at the upper end. Among the cities with the most properties, thirty cities have exactly 20 hostels each, and the maximum, Budapest, has 25. Cross-city analysis is therefore not dominated by a few outsized markets.

3.8.2. Data Acquisition

Data was collected in August 2025 by scraping publicly accessible Hostelworld endpoints. Requests were rate-limited to reduce load on the platform. The site’s structure provides numeric sequential IDs for each individual city, enabling simple and comprehensive iteration over all properties.

The reference scrape covers 4,514 properties across 949 distinct cities. The LLM pipeline operates on a deliberately smaller subset, consisting of 342 properties across 18 cities for the Chapter 4 analysis and an even smaller subset in later robustness analyses. The reduction is intentional. Within the available compute budget, more informative results can be obtained by running varied pipeline conditions and different experiments on the same set of properties than by extending coverage.

4. Construct Validity of LLM Aspect Ratings

The previous chapter described a scoring pipeline designed to reduce artificial sources of inter-aspect correlation. The present chapter evaluates whether this design succeeds empirically, whether the resulting LLM scores preserve plausible agreement with human ratings, and whether they provide more differentiated aspect-level measurement.

Construct validity is analysed in two steps. The first is validating that LLM scores fall within a plausible range and broadly preserve agreement with human ratings. Sections 4.1 and 4.2 address this part. They also document the strong halo contamination previously identified on other platforms in the literature review that motivates the rest of the analysis. In a second step, Sections 4.2.4 - 4.4 demonstrate that the LLM scores are more construct-valid than human numeric ratings, by showing that they decorrelate the scores in a structured way.

4.1. Agreement with Human Ratings

Before analysing the structure of the LLM scores, it is necessary to establish that they behave like plausible ratings.

Figure 4.1 plots LLM-predicted overall scores against human overall averages, pooling all eighteen cities ($n = 342$ hotels). The rank-order association is strong ($r = 0.975$), but the raw scores are both compressed and shifted relative to humans: when humans rate a hotel numerically at 0.6, this corresponds to a text-derived score of around 0.4. In this figure, calibration is performed within-sample. The calibrated MSE of 0.001 is therefore optimistic. However, the figure demonstrates that the overall predicted ratings strongly match the actual overall ratings.

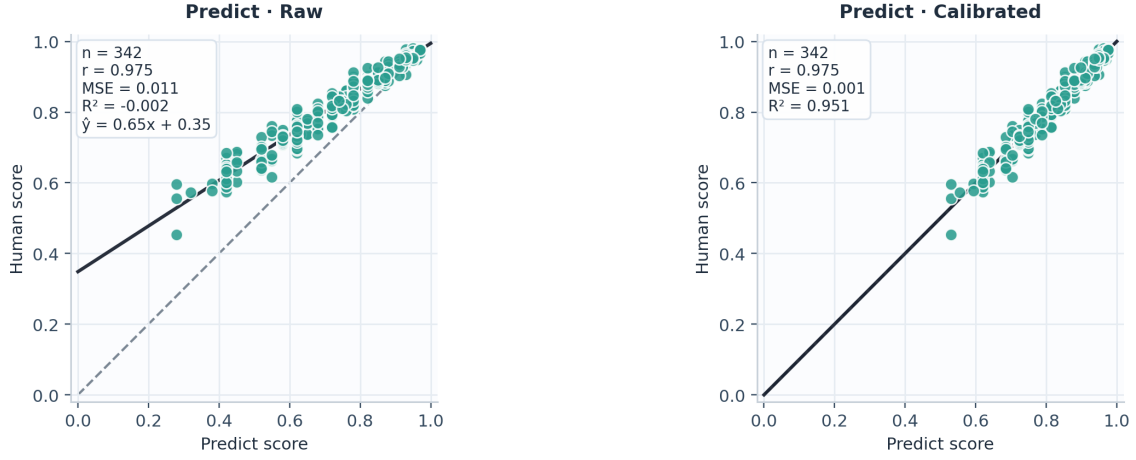


Figure 4.1.: Predict overall scores vs. human overall averages, raw (left) and after linear calibration (right). All eighteen cities pooled, $n = 342$ hotels.

The same pattern of predictive quality holds at the aspect level, although correlations are weaker and biases differ by aspect. Table 4.1 shows these values, averaged across cities. Cleanliness, staff, and the overall score achieve the strongest correlations ($r > 0.91$), while value for money is the weakest ($r = 0.712$). Biases range substantially, with security having the largest negative bias (-0.261) and location the smallest absolute bias (-0.035). This could partially reflect that the model rates some aspects consistently lower than it rates other aspects, but it could also reflect that users rate some aspects consistently high despite mixed sentiment in their review text. In either case, the results show that linear calibration is required if the scores are used in a prediction context.

Table 4.1.: Per-aspect agreement metrics (Predict condition), averaged across cities.

Aspect	Mean r	Mean Bias
Overall	0.960	-0.086
Staff	0.925	-0.072
Cleanliness	0.912	-0.181
Location	0.831	-0.035
Facilities	0.830	-0.177
Security	0.802	-0.261
Atmosphere	0.784	-0.199
Value for money	0.712	-0.146

These correlations should be interpreted cautiously. High values do not guarantee that the model is predicting each aspect separately. If the human aspect ratings are strongly intercorrelated, the model can achieve high agreement by detecting the overall sentiment of the review text and applying that impression across all aspects. In that case, the model may seem to predict cleanliness, location, and facilities as distinct dimensions, while in practice mostly tracking general hotel quality. The centred profile analyses from Section 4.4 test this.

4.2. Halo Effects in the Human Rating Baseline

Predicting whether a hotel is generally rated highly or poorly is not the main challenge. A single overall score can already capture that information. The reason to use multi-aspect ratings is that hotels can be good and bad in different ways. One property may have an excellent location but weak facilities; another may have better facilities but a less convenient location. Aspect-level scoring is useful only if it captures these kinds of differences. On Hostelworld, however, the seven aspect ratings often move together so strongly that they resemble one overall judgment rather than seven separate dimensions.

4.2.1. Magnitude of the Halo Effect

Across the eighteen cities, human aspect ratings are strongly intercorrelated. This means that ratings for supposedly separate dimensions tend to move together rather than vary independently. This subsection first provides an aggregate metric to quantify this pattern. It does not yet distinguish between aspect pairs that should plausibly be related and aspect pairs that should be more independent. That distinction is analysed later in Section 4.2.3, where the correlation structure is broken down by aspect clusters. Here, the narrower goal is to measure how strongly the seven human aspect ratings move together on average.

Average inter-item correlation. This relationship is summarised using the average inter-item correlation (AIIC). For each city, we first calculate the Pearson correlation between every pair of aspect ratings across hotels. Since there are seven aspects, this produces 21 pairwise correlations. The AIIC is the average of these correlations:

$$\text{AIIC}_c = \frac{2}{K(K-1)} \sum_{a < b} r_{c,ab},$$

where $K = 7$ is the number of aspects, and $r_{c,ab}$ is the correlation between aspect a and aspect b in city c . A higher AIIC means that the aspect ratings contain less independent information. In the extreme case, if all aspect ratings moved perfectly together, AIIC would be 1.

Across all cities, human aspect ratings are strongly intercorrelated. The average AIIC ranges from 0.652 to 0.910 (median 0.799), excluding Canggu (0.342), which has too few non-flat reviews for stable estimation. The first principal component explains between 75% and 92% of total variance in the remaining seventeen cities (median 85%).

4.2.2. Identical Aspect Scores and the No-Flat Benchmark

One direct source of the high inter-aspect correlation is the large number of reviews in which users assign exactly the same score to every aspect. This pattern, which we will refer to here as flat-lining, affects between 18% and 65% of reviews in the cities evaluated. In the full corpus, flat-lining affects about 40% of reviews. The pattern is more common among users who wrote fewer reviews and is often repeated by the same users. The probability of a flat review is 50% if the reviewer’s previous review was also flat, compared with 24% otherwise.

Flat reviews are also highly positive. Among reviews that assign the same score to every aspect, 97.6% rate with the highest possible score. Taken together, these patterns suggest that uniform scoring often reflects a positive, undifferentiating response style rather than a separate evaluation of the individual aspects.

To verify in subsequent chapters whether the decorrelation of LLM-derived scores follows a reasonable and structured pattern, we construct an alternative numeric rating benchmark that is formed by removing these reviews. This is labelled the No-Flat benchmark.

This benchmark does not represent a perfect ground truth, but it is required for a reasonable analysis of the halo effect. A uniform review can be a sincere judgment that all aspects were equally good or bad. However, for the purpose of estimating aspect-level structure, it contributes no information about relative differences between aspects. Removing flat-liners reduces AIIC in every city by only 0.08 to 0.32. Substantial halo persists: No-Flat AIIC still ranges from 0.439 to 0.831, with a median of 0.634. The benchmark therefore marks a direction of improvement rather than a destination, and should be interpreted accordingly.

4.2.3. Selective Correlation Across Aspect Pairs

Figure 4.2 visualises the halo effect using the correlation matrix for Florence. In the matched human matrix, nearly every off-diagonal cell is strongly positive; only location shows partial independence, correlating at roughly 0.25–0.46 with other aspects. Removing flat-liners preserves the heavy positive clustering among the six subjective aspects but pushes location to near-zero or slightly negative correlations, a qualitatively different structure. The correlation matrix of the LLM-derived scores in the Predict variant goes further: the cluster of physical characteristics (cleanliness, facilities, and security), stays highly correlated. The subjective cluster (atmosphere, staff, value) has distinctly lower correlations, and location is the most decoupled aspect.

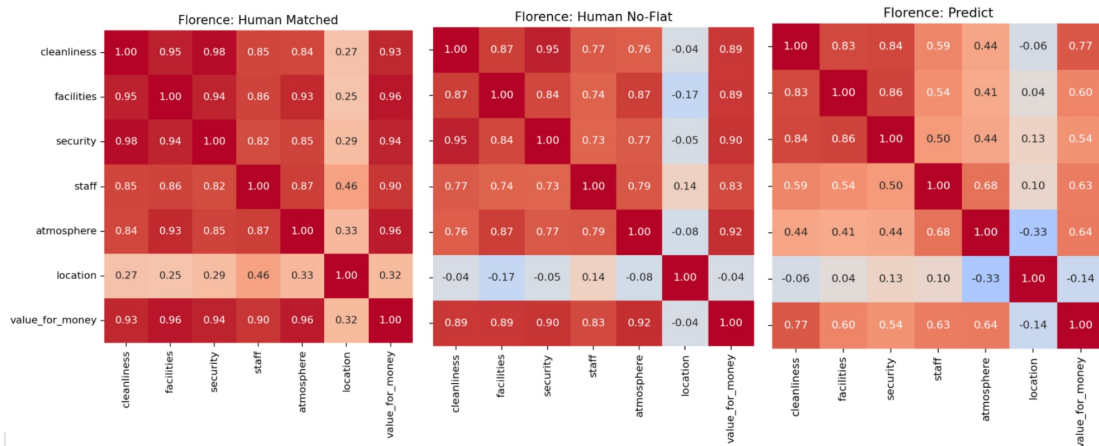


Figure 4.2.: Aspect correlation heatmaps for Florence. Left: human matched sample. Middle: human No-Flat. Right: LLM Predict.

The high average correlation does not mean that all aspect pairs move similarly. In all measurement instances, some aspects are more tightly connected than others: physical-cluster pairs (cleanliness, facilities, security) are the most connected, subjective pairs (involving atmosphere, value for money, and staff) slightly less so, and location pairs are the most independent. Figure 4.3 makes this explicit.

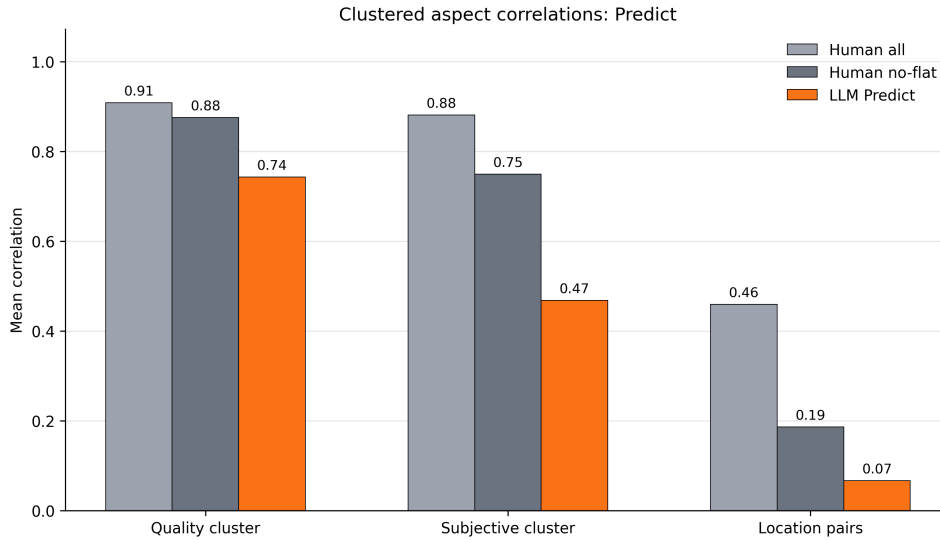


Figure 4.3.: Selective decorrelation: mean correlation by cluster (Physical, Subjective, Location) for Human all, Human No-Flat, and LLM Predict.

It also shows that the LLM’s lower correlations are not arbitrary. They follow the same pattern that appears when uniform human reviews are removed. The physical cluster changes only a little from the full human sample to the No-Flat benchmark, and it also remains the most connected cluster in the LLM scores. Subjective pairs separate more clearly in the No-Flat benchmark, and the LLM decorrelates them more strongly than the physical cluster. Location shows the strongest break from the full human baseline, and correspondingly shows the lowest intercorrelation in the LLM-generated scores. In other words, the model does not create a completely new structure. It strengthens the same pattern of differentiation that is already visible among the more discriminating human ratings.

4.2.4. Pipeline Design Affects Halo

All LLM variants produce substantially lower inter-aspect correlation than either human baseline (Table 4.2). The reduction follows the ablation design: Every step that was designed to provide a more independent evaluation of aspects reduces criterion entanglement.

Table 4.2.: Mean AIIC across cities by source.

Source	Mean AIIC
Human	0.767
No-Flat	0.610
Joint	0.415
SchemaLeaks	0.383
Predict	0.361
Rate	0.337

**Figure 4.4.:** AIIC by scoring source and city, sorted by Predict position. City means are shown at the bottom. Dot colours correspond to scoring sources: Rate, Predict, SchemaLeaks, Joint, No-Flat, and Human.

Figure 4.4 shows the central findings of the ablation experiment and demonstrates the high contribution of schema awareness to criterion entanglement. The figure normalises Predict to 0 and Joint to 1, visualising the relative AIIC contribution of SchemaLeaks (0.4) within these two variants.

When ordering the pipeline variants by AIIC, the ordering Human > No-Flat > Joint > SchemaLeaks > Predict > Rate holds in every city except Barcelona, where the last two are reversed. For that city, the run was repeated to verify this was not introduced through an unintended pipeline change.

This means that removing co-generation accounts for roughly 60% of the observed Joint-to-Predict AIIC reduction, while removing schema awareness accounts for the remaining 40%. The schema-awareness effect is therefore an important component: It contributes almost equally to the changes in criterion-entanglement as the co-generation effect that was previously identified in the literature.

Some limitations apply to the interpretability of the results. First, the percentages observed are descriptive for the current dataset and might not carry over to other platforms. Second, the aspect order in the Joint variant is fixed, with the security score generated first. This

matters because security, the first generated aspect, has the largest negative bias among all generated aspect scores. For a clearer measurement of AIIC in the Joint variant, more aspect order combinations should be tested and averaged.

4.3. The Reduction Is Structured, Not Noisy

The cluster analysis in Section 4.2.3 shows that the LLM’s decorrelation is selective: location pairs are most affected, subjective pairs show moderate reductions in correlation, and physical-cluster pairs remain comparatively coherent. This supports the interpretation that the LLM is not simply adding random noise to each aspect. However, the cluster analysis is still a coarse and potentially biased summary. It groups the 21 pairwise aspect correlations into three reasonable, but retrospectively defined, categories.

This section formalises the idea that the LLM decorrelates in a structured way. It introduces a metric that does not depend on predefined categories, and that shows the similarity to No-Flat human correlation patterns in a single number.

4.3.1. Aggregate Trajectory Alignment

Section 4.2.3 introduced a visually intuitive correlation matrix for the seven aspect pairs. After excluding symmetric pairs and self-correlations, the informative part consists of 21 unique aspect correlations. These values are represented as a 21-dimensional vector that characterises the correlation structure of all hotels in a city.

Let

$$\rho_c^H, \rho_c^{NF}, \rho_{c,m}^L \in \mathbb{R}^{21}$$

denote the 21-correlation vectors for Human All, Human No-Flat, and LLM condition m in city c . The human de-haloing shift and the corresponding LLM shift are:

$$\begin{aligned}\Delta_c^{NF} &= \rho_c^{NF} - \rho_c^H \\ \Delta_{c,m}^L &= \rho_{c,m}^L - \rho_c^H.\end{aligned}$$

If the LLM decorrelates aspect scores in the same way as the No-Flat correction, these two shift vectors should point in a similar direction. This is measured using cosine similarity:

$$\cos(\Delta_c^{NF}, \Delta_{c,m}^L) = \frac{\Delta_c^{NF} \cdot \Delta_{c,m}^L}{\|\Delta_c^{NF}\| \|\Delta_{c,m}^L\|}.$$

A value close to +1 means that the LLM changes the same pairwise correlations in the same direction as the No-Flat correction. A value near 0 means that the two movements are unrelated, while a negative value would indicate movement in opposite directions.

Because cosine similarity measures only direction, the relative size of the LLM shift along the human de-haloing trajectory is also reported:

$$P_{c,m} = \frac{\Delta_{c,m}^L \cdot \Delta_c^{NF}}{\Delta_c^{NF} \cdot \Delta_c^{NF}}.$$

This projection ratio measures the component of the LLM shift that lies in the Human All to Human No-Flat direction. A value of $P_{c,m} = 1$ means that the LLM moves as far as the No-Flat correction along this trajectory. Values above 1 indicate that the LLM moves further in the same de-haloing direction, while values between 0 and 1 indicate a smaller movement in that direction. Negative values would indicate movement opposite to the No-Flat correction.

This metric therefore formalises the cluster result without relying on the retrospective grouping of aspect pairs. It asks whether the full correlation structure moves in the same direction as the empirical human de-haloing benchmark.

Table 4.3.: Aggregate trajectory alignment: cosine similarity between the LLM shift and human de-biasing vectors, with relative magnitude for the Predict variant.

City	Joint	SchLk	Predict	Rate	Magnitude
Amsterdam	0.41	0.47	0.46	0.46	1.30
Barcelona	0.90	0.89	0.87	0.86	1.25
Berlin	0.91	0.93	0.93	0.93	2.38
Lisbon	0.88	0.92	0.94	0.92	2.05
New York	0.87	0.86	0.87	0.88	4.01
Florence	0.69	0.76	0.76	0.79	1.26
Singapore	0.84	0.88	0.88	0.87	3.82
7-city mean	0.79	0.82	0.82	0.82	2.30

All cosines are positive and most exceed 0.80. Amsterdam (0.41–0.47) is the consistent outlier: it is the only city in which removing flat-liners increases certain location correlations, producing an unusual de-biasing vector that no LLM condition reproduces. The pipeline conditions yield similar directional cosines. What distinguishes them is how far they move (Section 4.2.4).

The magnitude column shows that the LLM typically moves well beyond the No-Flat correction. The overshoot is largest where flat-liners constitute a small share of reviews (New York, Singapore), implying that the No-Flat benchmark captures only a modest portion of the structural halo in those cities.

This result supports the chapter’s central interpretation: the LLM weakens the same correlations that human de-haloing weakens. But the analysis operates at the level of entire correlation matrices. It compares the structure of relationships across all hotels in a city. It does not yet tell us whether the LLM captures the correct pattern of strengths and weaknesses for any individual hotel. That is the subject of the next section.

4.4. Within-Hotel Profile Recovery

A stricter test asks whether the LLM captures the pattern of relative strengths and weaknesses within individual hotels. Two hotels may both score 8/10 overall, but one might be praised for its location and criticised for its facilities while another shows the reverse. This internal

profile, meaning which aspects stand out relative to the hotel’s own mean, is what makes multi-aspect scoring informative beyond a single overall rating.

4.4.1. Centred Profile Agreement

To isolate the internal profile, each hotel’s aspect scores are centred on that hotel’s own mean. Let $\mathbf{x}_i$ be the vector of aspect scores for hotel i , and let $\bar{x}_i$ be its mean across aspects. The centred profile is:

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{x}_i \mathbf{1}.$$

This transformation removes the overall quality level and leaves only relative emphasis: which aspects are relatively strong or weak for that specific hotel. This makes the profile less sensitive to level differences across scoring instruments. For example, if the LLM is systematically harsher than users, that bias affects the absolute score level but is reduced once both profiles are centred on their own means.

Agreement between the centred LLM profile and the centred Human No-Flat profile is measured using cosine similarity and pair-ordering accuracy. Cosine similarity captures whether the two centred profiles have the same overall shape. Pair-ordering accuracy measures whether the LLM ranks pairs of aspects in the same relative order as the de-haloed human benchmark. Together, these metrics test whether the LLM recovers the internal profile of each hotel, beyond its overall rating level.

Table 4.4.: Centred within-hotel profile agreement (LLM vs. Human No-Flat), by city. Closer% = percentage of hotels for which the LLM centred profile is closer to No-Flat than to Human All, using L1 distance.

City	Joint cos	Joint PairOrd	Predict cos	Predict PairOrd	Closer %
Amsterdam	0.80	77.7	0.77	75.3	90.9
Barcelona	0.77	77.8	0.75	76.4	91.3
Berlin	0.79	78.1	0.74	75.9	86.4
Lisbon	0.72	75.2	0.72	73.1	95.0
New York	0.81	80.1	0.82	77.0	100.0
Florence	0.80	79.3	0.77	75.5	92.9
Singapore	0.88	86.5	0.88	85.7	100.0

Centred cosine similarities are consistently high, ranging from 0.72 to 0.88. This indicates that the LLM recovers meaningful within-hotel differences. In 86% to 100% of hotels, the centred LLM profile is closer to the Human No-Flat benchmark than to the raw human profile. The LLM differs from the raw human baseline in the direction of the de-haloed benchmark.

4.5. Summary

LLM scores have high rank correlation with human numeric scores at the overall level ($r = 0.975$) and moderate correlation for aspects, ranging from 0.71 (value for money) to 0.93 (staff). They are subject to a strong negative bias, requiring linear calibration if predictive capabilities

are required. This served as a basis for the model scores to be sufficiently plausible as measurement instrument of the true quality of a property. The chapter then documented that the numeric platform scores suffer from substantial halo, with 40% of Hostelworld reviewers assigning identical scores across all aspects, and AIIC averaging 0.77 across cities. Every LLM configuration reduces halo, and the reduction follows the ablation design. Three arguments are presented for why the reduction of correlations represents a plausible direction. First, the LLM keeps correlation in pairs that should correlate while reducing correlation in pairs that should plausibly move independently. Second, the LLM's correlation structure strongly matches the correlation structure of a subset of reviewers that rate more differentiatingly. Third, the model not only ranks hotels against each other based on their aspect scores, but also recovers the internal relative strengths and weaknesses of individual hotels. Together, these findings demonstrate that the LLM scores provide superior construct validity in cases where numeric ratings are strongly impacted by halo effects.

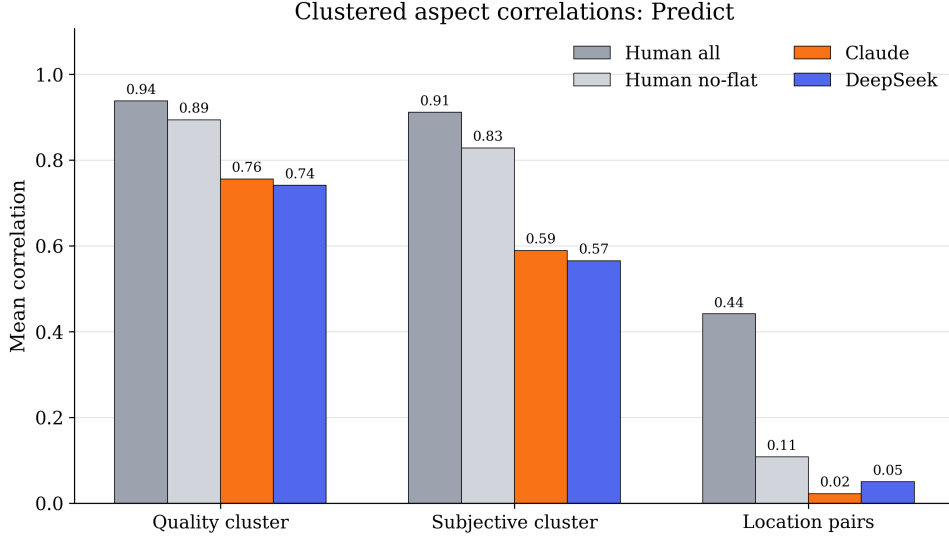
5. Cross-Model Robustness and Operational Stability

Chapter 4 established the construct-validity findings using a single model, Claude Sonnet 4.6 [67]. This chapter tests whether those findings generalise to DeepSeek V3.2 on a matched 162-hotel subset across nine cities [68]. The analysis separates three issues. First, it asks whether DeepSeek reproduces the broad hotel-level structure found with Claude. Second, it asks whether DeepSeek is stable enough to support the finer ablation logic used to compare pipeline variants. Third, it examines whether the model behaves reliably as an operational component of the scoring pipeline. The answer is mixed: DeepSeek replicates the aggregate hotel-level pattern after invalid null emissions are removed, but its rerun noise and unexpected output failures make it less suitable for fine-grained academic experiments without additional safeguards.

5.1. Construct Validity Replication

DeepSeek reproduces the main findings of Chapter 4 after filtering invalid score outputs, which are analysed separately in Section 5.3. Scores are strongly decorrelated, and the structure of what aspects correlate together matches the pattern identified with Claude. Correlations in the quality cluster (cleanliness, facilities, security) remain comparatively high. Subjective correlations fall more strongly. Location pairs are decorrelated the most.

Figure 5.1 compares Claude and DeepSeek in the Predict condition across 162 matched hotels from all nine cities, corresponding to 1,134 hotel-aspect cells. Table 5.1 reports the hotel-level metrics of Chapter 4 in the Joint condition.

**Figure 5.1.:** Clustered aspect correlations in the Predict condition.**Table 5.1.:** Joint hotel-level replication across matched hotels.

Model	Mean MAE	Pearson Correlation			Cosine Similarity	
		Mean	Min	Max	Numeric Centred Profile	De-Halo Trajectory
Claude	0.1414	0.8926	0.8385	0.9601	0.7865	0.8410
DeepSeek	0.0909	0.8538	0.7700	0.9282	0.8566	0.8337

The DeepSeek metrics in Table 5.1 are computed after filtering the null emissions described in Section 5.3. The table therefore evaluates DeepSeek’s usable outputs after post-validation, not the stability of the raw output stream. This distinction matters because the same model can reproduce the hotel-level structure once invalid rows are removed while still being less operationally stable than Claude.

Mean aspect Pearson is computed by correlating each model’s hotel-level aspect scores with the matched human aspect means and then averaging these seven aspect-specific correlations. In both models, cleanliness has the highest correlation with human ratings, while security has the lowest.

DeepSeek has lower mean absolute error partially because its scores are less negatively biased than Claude’s. That bias was illustrated in Figure 4.1: Claude consistently rates with lower scores than humans and requires linear recalibration; for DeepSeek this behaviour is less extreme.

Claude preserves cross-hotel rankings more consistently aspect by aspect, but DeepSeek recovers within-hotel profile shapes better, as shown by the higher centred cosine. Trajectory similarity is almost identical (0.84 versus 0.83): both models shift inter-aspect correlations

in the same direction as the human de-haloing benchmark, by the same amount. This is the numerical representation of what is demonstrated in Figure 5.1.

These aggregate metrics therefore indicate that DeepSeek replicates the main construct-validity claim: LLM-derived scores provide more differentiated and less halo-contaminated aspect-level measurement than human numeric ratings.

5.2. Rerun Noise and Ablation Interpretability

Chapter 4 also identified schema awareness as a substantial and previously underexplored contributor to criterion entanglement. Replicating this ablation result on another model requires sufficiently low rerun noise. This means that scoring the same reviews under identical conditions should produce stable scores. If ordinary rerun variation is as large as the differences among Joint, Predict, and SchemaLeaks, then the ablation cannot distinguish pipeline design variation effects from model instability.

For cost reasons, rerun stability is measured only in the cheapest condition (Joint) on a subset of N hotels across four cities: Barcelona, Florence, Porto, and Prague. Each hotel is scored three times under identical conditions, and the standard deviation across reruns is taken as the rerun-noise floor for that model.

Figure 5.2 separates three quantities: rerun noise, cross-condition variation, and between-hotel dispersion. Noise SD reflects how much the scores of a single aspect change under three identical Joint reruns. Variant SD is the spread of scores across the Joint rerun mean, Predict, and SchemaLeaks variants for that same hotel-aspect observation. Hotel SD is between-hotel dispersion using the Joint mean within each aspect.

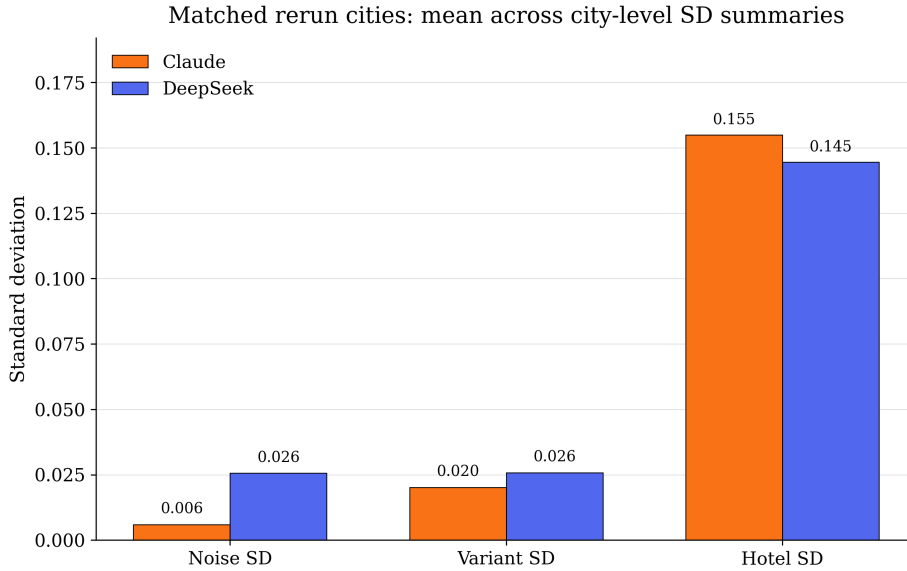


Figure 5.2.: Noise, variant, and between-hotel standard deviation, pooled across Barcelona, Florence, Porto, and Prague.

Claude has the reliability structure needed for the Chapter 4 ablation. Its hotel SD is roughly 39 times its rerun noise, and its variant spread is about 5.6 times its rerun noise. Hotels differ from one another by much more than ordinary rerun instability, and prompt variants differ enough to be interpreted.

DeepSeek supports hotel-level aggregation but not the ablation. Its hotel dispersion is about 6.0 times rerun noise, which is enough for aggregate hotel comparisons. But its variant spread, approximately 0.026, is indistinguishable from its rerun noise, also approximately 0.026, meaning that any apparent difference between Joint, Predict, and SchemaLeaks could be explained by ordinary stochastic variation alone. In this design, differences between Joint, Predict, and SchemaLeaks are not separable from ordinary DeepSeek instability.

Rerun noise is measured only for Joint, and more reruns would lower the effective noise floor. But under the rerun budget used here, the Chapter 4 ablation cannot be replicated on DeepSeek.

5.3. Null Emissions as Operational Failures

DeepSeek sometimes returns results in which all requested scores are 0.0. Since 0.0 is within the allowed $[0, 1]$ scale, these rows pass schema validation, even though they do not reflect a reasonable assessment of the review text. If not filtered, they would be treated as normal predictions. Claude does not show the same failure mode in the observed runs.

Null emissions appear in Joint and Predict but not in SchemaLeaks, and they are unevenly distributed across cities. Porto, Prague, and Florence account for most observed cases; several other cities show none.

Table 5.2.: Null-row incidence. Failures divided by total calls, with percentages.

Group	Joint	Predict
Porto	6 / 57, 10.53%	17 / 418, 4.07%
Prague	5 / 63, 7.94%	22 / 462, 4.76%
Florence	3 / 42, 7.14%	28 / 308, 9.09%
Barcelona	2 / 69, 2.90%	0 / 250, 0.00%
Amsterdam	1 / 66, 1.52%	0 / 242, 0.00%
Other DeepSeek cities	0 / 64, 0.00%	0 / 725, 0.00%
Claude, all cities	0 / 363, 0.00%	0 / 1,815, 0.00%

The failures are not concentrated on particular hotels. In Prague, for example, the three Joint reruns share no failing hotel. Barcelona and Amsterdam were sampled from the initially unaffected DeepSeek cities to test whether null emissions were confined to Porto, Prague, and Florence. Neither city showed null rows in the first run, and neither showed null rows in Predict. However, both produced null rows in one of the two Joint reruns. The issue is therefore not limited to the original problem cities, nor is it explained by a one-off pipeline change; it appears to be probabilistic. The city-level pattern suggests that review characteristics likely matter, but the mechanism is unclear.

The practical implication is straightforward: all scores equal to 0.0 must be removed before analysis. But the issue also highlights that despite broadly comparable metrics, DeepSeek is a more fragile model that can fail in unexpected ways.

5.4. Reduced Instruction-Following

Null rows are not the only reliability issue. DeepSeek also has weaker instruction following in SchemaLeaks, the condition where isolation matters most.

In this variation, the prompt asks for one aspect per call, but the schema exposes the full aspect list. This creates two possible failures. The model may omit the requested aspect and return another one, or it may return multiple aspects despite the single-aspect prompt. The first is a direct compliance failure. The second weakens the intended isolation of the condition.

Table 5.3.: Post-validation outcomes in SchemaLeaks.

Model	First pass	Second	Third or later	Fallback only	Unresolved
Claude	99.94%	0.06%	0.00%	0.00%	0.00%
DeepSeek	92.07%	5.12%	0.11%	2.70%	0.00%

Claude almost always satisfies the SchemaLeaks constraint on the first pass. DeepSeek does not. Only 92.07% of DeepSeek SchemaLeaks calls pass immediately. Among the variation’s failed calls, 62.50% fail because the requested aspect is missing, 36.81% fail because multiple aspects are co-generated, and 0.69% fall into other categories.

Retry and fallback eventually resolve all calls, but they do not fully restore the experiment. The fallback path does not reimpose the single-aspect constraint. For DeepSeek, 2.70% of SchemaLeaks calls are accepted only through fallback, which marks the subset where the stricter isolation constraint is no longer guaranteed.

This matters because SchemaLeaks is designed to isolate the effect of schema awareness: the model sees the full aspect list but is asked to score only one. When the model returns multiple aspects in a single response, the sequential anchoring that SchemaLeaks is meant to exclude is reintroduced, and the condition no longer cleanly isolates schema awareness from co-generation.

Because DeepSeek’s rerun noise already exceeds its variant spread (Section 5.2), this compliance failure has limited additional impact: SchemaLeaks cannot be cleanly interpreted on DeepSeek regardless of fallback behaviour. The finding is nonetheless relevant for any future work that considers DeepSeek, as it highlights differences in instruction following.

5.5. Effective Inference Cost

At listed OpenRouter prices, Claude Sonnet 4.6 has substantially higher costs than DeepSeek V3.2, at 11.9x input and 39.7x output prices [69, 68]. The relevant comparison, however,

is effective cost per hotel, which depends on scoring design and provider-specific caching behaviour.

Figure 5.3 shows that caching effects differ substantially across conditions. Claude Predict benefits most from caching (+78.9%): its eleven-call-per-hotel structure shares an identical review-block prefix across calls, and a 90% discount on cached-input reads applies to that shared prefix [70].

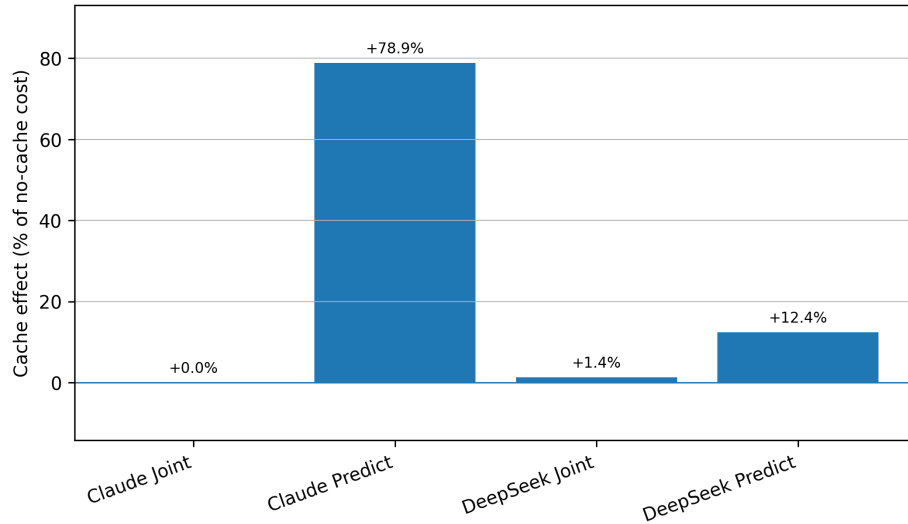


Figure 5.3.: Caching effects by model and condition.

Claude Joint receives no cache benefit, as each hotel is scored in a single call. Furthermore, the system prompt with the rating instructions falls below the 2,048-token minimum cacheable prompt length for Claude Sonnet 4.6 [71].

DeepSeek shows modest gains under both conditions (+1.4% Joint, +12.4% Predict). It does not require a minimum prefix length for caching to work, but the discount on cached-input reads is only about 50% as measured by actual savings, and caching works much less reliably. DeepSeek’s documentation states that caching is best effort and does not guarantee a 100% hit rate [72].

SchemaLeaks is not shown separately, as its cost profile mirrors Predict.

Costs can be reduced further with batch processing, in which requests are processed asynchronously rather than in real time and receive a 50% discount on inference [70, 73, 74]. For Claude Joint and the DeepSeek variants, this is a viable solution. On Amazon Bedrock, this analysis treats Batch Inference and prompt caching as separate cost-reduction paths. AWS documents prompt caching for supported runtime features such as Converse, InvokeModel, cross-region inference, and Prompt management, whereas Batch Inference jobs are submitted through the separate `CreateModelInvocationJob` workflow. AWS does not document prompt-caching support for Batch Inference jobs [75, 73, 76].

Figure 5.4 shows per-hotel costs for real-time inference. Prices are shown in cents. Claude Predict (13.6¢) costs roughly twice Claude Joint (6.2¢) despite eleven calls per hotel, with

caching accounting for the difference. DeepSeek Joint (0.5¢) is by far the cheapest condition; Predict (4.6¢) is nearly nine times more expensive, reflecting the limited caching reliability and lower caching discounts.

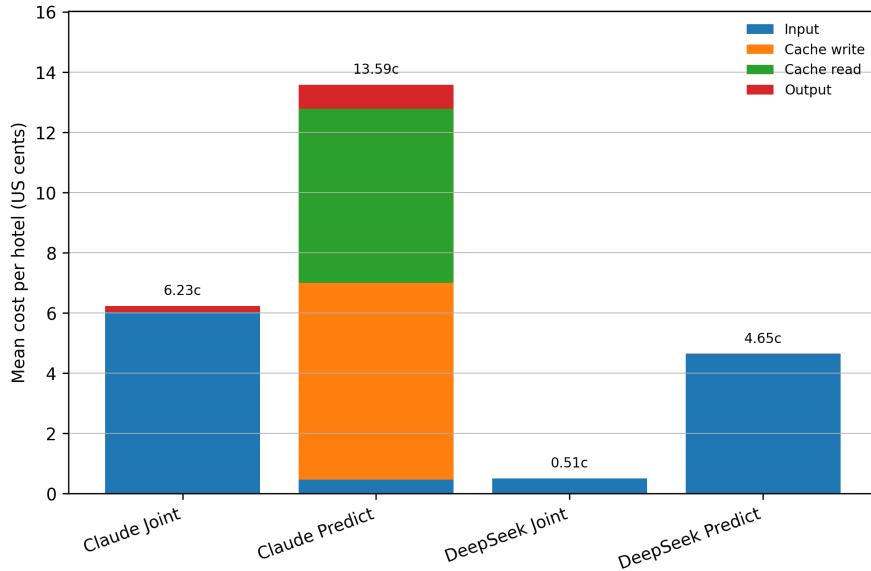


Figure 5.4.: Per-hotel costs for real-time inference. Prices are shown in cents.

Applying the batch discount would reduce all non-caching variants by approximately half, bringing DeepSeek Joint to around 0.26 cents per hotel [70, 74].

For most analytical purposes, DeepSeek Joint with batch processing is therefore the most practical configuration. At that cost, multiple reruns become affordable, and averaging across them directly addresses the output instability documented in the preceding section.

5.6. Summary

The results point to a limited but useful role for DeepSeek.

Claude potentially serves as a more useful model for academic research related to prompt design. The Chapter 4 ablation depends on small differences between prompt conditions. Claude’s variant spread is substantially larger than its rerun noise, which makes those differences interpretable. DeepSeek’s variant spread is approximately equal to its rerun noise at the rerun budget used here, so differences between Joint, SchemaLeaks, and Predict cannot be interpreted with comparable confidence.

For hotel-level tasks, the conclusion is different. DeepSeek can be used as a lower-cost alternative when the goal is to rank hotels, recover within-hotel profiles, or produce halo-reduced aggregate scores. In the Joint hotel-level comparison across all nine cities, DeepSeek recovers the broad cross-hotel pattern, has lower mean absolute error, shows stronger within-hotel profile agreement, and produces a de-haloing trajectory very close to Claude’s.

This use of DeepSeek requires safeguards. Null results must be filtered. JSON schemas are enforced less reliably and the correct format of outputs should be validated manually. Scores should be averaged across reruns when used for hotel-level aggregation.

The recommendation is therefore split by use case. Use Claude when the analysis depends on small prompt-condition differences or reproducible single-run estimates. Use DeepSeek for hotel-level aggregation when filtering, validation, and rerun averaging are in place. Its lower nominal cost is real, but the resulting scores require more validation and averaging than Claude's.

6. Reliability Under Reduced Review Evidence

6.1. From Rerun Stability to Sampling Reliability

Chapter 2 defined reliability as the stability of a measurement instrument under comparable conditions. For hotel-aspect scoring, this stability has more than one source. Chapter 5 examined rerun reliability: whether the LLM produces consistent scores when the same reviews are scored repeatedly under identical conditions. The present chapter examines a different source of instability: sampling reliability, which concerns whether the score reflects a stable hotel-level signal or the idiosyncrasies of the particular reviews that happened to be sampled.

Chapter 4 established the LLM’s construct-validity advantage using up to 110,000 characters of review text per hotel, corresponding to roughly 400 reviews on average. This chapter asks whether LLM-based scoring remains stable when the evidence base is much smaller. For each hotel, smaller random review samples are scored and compared with the score from the 110,000-character reference window. The experiment measures how quickly each scoring instrument converges toward its own large-sample reference as the amount of review evidence increases. This makes the comparison a test of sampling reliability.

The two instruments use sampled reviews in different ways. The numeric mean receives a complete aspect rating from every sampled review and gives each rating equal weight. This makes it sensitive to atypical reviews, especially when the sample is small.

The LLM’s behaviour is less predictable. It may be less affected by isolated complaints because it can interpret each review in relation to the surrounding evidence. This could improve stability at small input sizes. However, the LLM only receives evidence for aspects that reviewers mention in text. Numeric ratings provide a value for every aspect by design; review text provides uneven coverage. Cleanliness and location are often discussed, while aspects such as security may appear rarely. The LLM’s sampling reliability therefore depends not only on how many reviews are sampled, but also on whether those reviews contain enough aspect-specific evidence.

6.2. Review Subsampling Design

The experiment covers 165 hotels across nine cities: Amsterdam, Barcelona, Berlin, Florence, Lisbon, New York, Porto, Prague, and Singapore.

For each hotel, review samples were drawn at four input sizes: 1,250, 2,000, 5,000, and 10,000 characters. These correspond to approximately 5, 8, 21, and 43 reviews per draw. Each draw was repeated under seven random seeds. Reliability was only tested for the Joint variant

of Claude.

Let R_h denote the full set of reviews available for hotel h , and let $R_h^{N,s}$ denote the subset sampled at input size N under seed s :

$$R_h^{N,s} \subset R_h.$$

6.3. Reference Scores and Reliability Metrics

Both scoring instruments are evaluated on the same sampled reviews. The numeric mean uses the aspect ratings attached to those reviews, while the LLM uses the corresponding review text. For the numeric score, the per-aspect ratings attached to the sampled reviews are averaged. For the LLM score, the text of those same reviews is scored under the Joint condition of Chapter 3. For scoring instrument I , the small-sample score is denoted

$$\hat{\mu}_{h,a,N,s}^I$$

representing the score for hotel h , aspect a , input size N , and seed s .

Each small-sample score is compared against a large-sample reference produced by the same scoring instrument:

$$\mu_{h,a}^{I,*}.$$

The reference is not treated as ground truth, but as the instrument's high-evidence estimate. It is computed on the recent-first review window capped at 110,000 characters per hotel, as defined in Section 3.5. For the numeric mean, $\mu_{h,a}^{\text{num},*}$ is the mean of the aspect- a ratings within that window. For the LLM, $\mu_{h,a}^{\text{LLM},*}$ is the model's Joint score on the text in that window. Both references use the same set of reviews; only the scoring rule differs.

For each aspect and input size, reliability is measured by how closely the scores from small review samples match the scores from the large review sample. A high correlation means that the score reaches a stable value even with little text.

The calculation is done separately for each seed. For each scoring instrument, aspect, input size, and seed, small-sample scores are correlated with the corresponding large-sample references across hotels:

$$r_{a,N,s}^I = \text{corr}_h \left(\hat{\mu}_{h,a,N,s}^I, \mu_{h,a}^{I,*} \right). \quad (6.1)$$

This quantity captures the extent to which one sampled review set per hotel preserves the hotel ranking established by the large-sample reference.

The reported value for each aspect and input size averages the seven seed-specific correlations:

$$r_{a,N}^I = \frac{1}{7} \sum_s r_{a,N,s}^I. \quad (6.2)$$

Section 6.4 reports the average across the seven substantive Hostelworld aspects, excluding

overall because it is mechanically aggregated from the others:

$$\bar{r}^I(N) = \frac{1}{7} \sum_{a \in \mathcal{A}} r_{a,N}^I, \quad (6.3)$$

$$\mathcal{A} = \{\text{cleanliness, location, staff, atmosphere, facilities, security, value for money}\}.$$

6.4. Convergence to the Large-Sample Reference

Figure 6.1 plots reliability as a function of input size. Both curves increase with additional text, and the numeric mean is more reliable at every tested input size. At $N = 1,250$, approximately 5 reviews, $\bar{r}^{\text{num}} = 0.713$, compared with $\bar{r}^{\text{LLM}} = 0.640$. At $N = 10,000$, approximately 43 reviews, the corresponding values are 0.956 and 0.896. The gap remains similar across the full tested range.

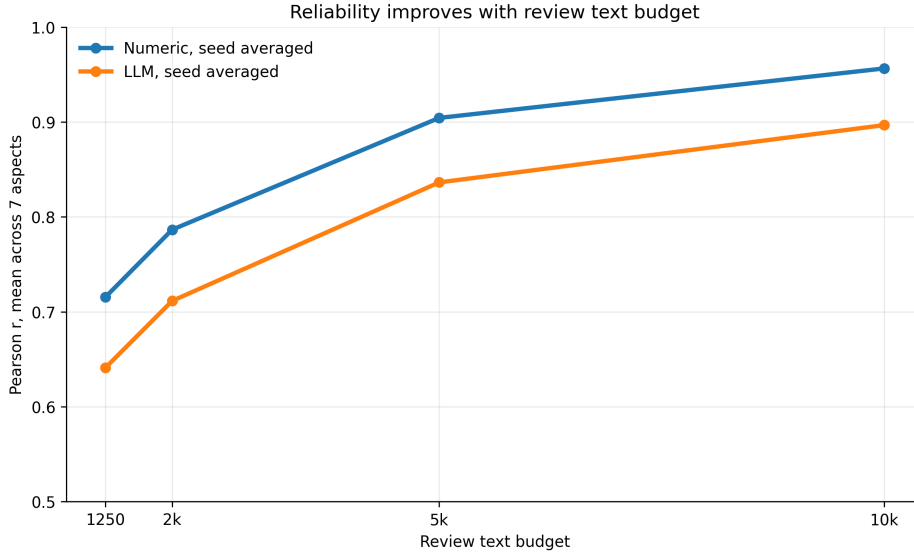


Figure 6.1.: Reliability by input size. Values average the seven substantive aspects across seven random seeds. Nine cities, 165 hotels.

For the numeric mean, this pattern is partly expected by construction. The small-sample score is a mean of sampled ratings, and the large-sample reference is a mean of a larger set of the same ratings. As the number of sampled reviews increases, the smaller mean should approach the larger mean by the law of large numbers.

For the LLM, convergence relies on accumulating contextual evidence rather than mathematical averaging. More reviews should give the model a broader evidence base, so some movement toward the large-sample reference is expected. But the model does not average review-level aspect scores. It produces one hotel-level judgment from a review block, and could in principle use the additional text in a less mean-like way, for example by changing

how it weighs rare incidents or conflicting evidence. The fact that the LLM curve broadly tracks the numeric convergence pattern therefore suggests that the model is functioning, at least in part, by averaging the sentiment it observes in the reviews.

6.5. Aspect-Level Reliability and Evidence Sparsity

The aggregate curve hides important aspect-level differences. Figure 6.2 plots $r_a^{\text{num}}(N)$ and $r_a^{\text{LLM}}(N)$ separately for each aspect.

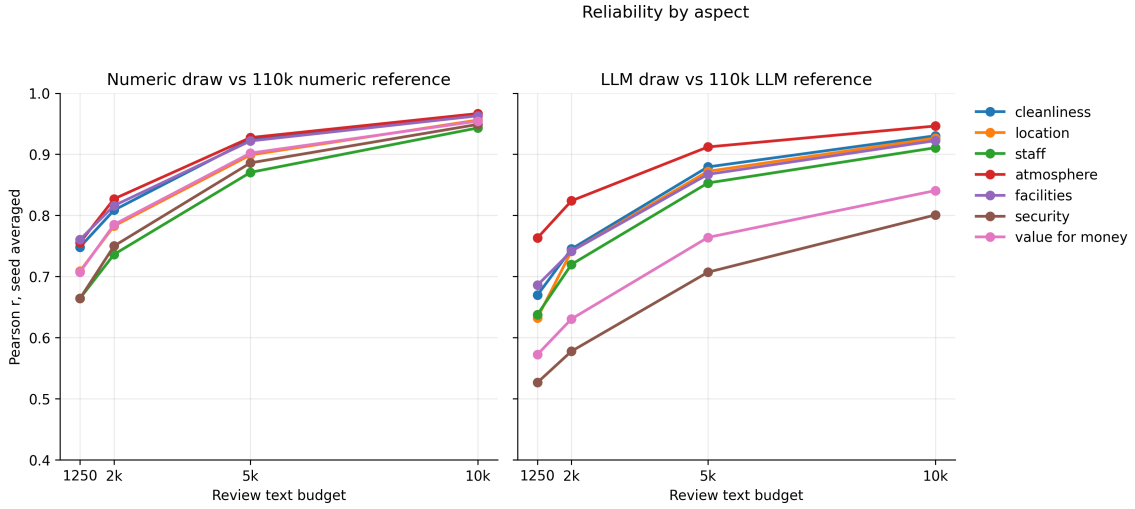


Figure 6.2.: Per-aspect reliability by numeric mean and LLM score. Left panel: numeric mean. Right panel: LLM.

For the numeric mean, reliability is highly consistent across the seven aspects at every input size. At $N = 10,000$, aspect-level correlations range from 0.943 for staff to 0.966 for atmosphere, a total spread of 0.023.

The LLM results show a more differentiated structure. Five aspects, cleanliness, location, staff, atmosphere, and facilities, form a relatively stable upper group, while security and value for money remain systematically lower. At $N = 10,000$, the upper group ranges from 0.91 to 0.95. By contrast, security reaches $r_{\text{security}}^{\text{LLM}} = 0.800$, and value for money reaches $r_{\text{value}}^{\text{LLM}} = 0.840$. At the smallest input size, $N = 1,250$, the divergence is larger: security and value for money reach 0.524 and 0.570, while the remaining aspects range from 0.63 to 0.76. Excluding these two aspects would reduce the aggregate numeric to LLM reliability gap in Figure 6.1 by approximately one half.

The same two aspects also produced the weakest results in Chapter 4. In that chapter, where the benchmark was the human numeric mean, security and value for money had the lowest cross-hotel correlations, at $r = 0.802$ and $r = 0.712$. In the present chapter, the benchmark is instead the LLM’s own large-sample reference score. The recurrence of the same two weak aspects under a different benchmark indicates that the issue is not merely a

disagreement between LLM scores and human ratings. Rather, these aspects appear to be less stable targets for LLM-based scoring because the relevant evidence is less consistently available in review text.

6.6. Summary

The likely reason is the form in which evidence appears. Location and cleanliness are commonly mentioned in Hostelworld reviews, so even small samples often contain relevant textual evidence for these aspects. Security is often discussed only when a concrete incident occurs, while value for money is frequently expressed indirectly through comments about price, expectations, trade-offs, or comparison points. This demonstrates that for aspects that are infrequently discussed, there is a natural ceiling to the usefulness of LLM-based aspect scoring regardless of model capabilities.

The Chapter 4 result therefore depends on sufficient review volume and sufficient textual coverage of the aspect being scored. Where both conditions hold, as for cleanliness, location, staff, atmosphere, and facilities, the LLM is the better instrument in terms of construct validity.

7. Informativeness of Scores

Claude and DeepSeek often assign the same numeric score to different hotels rated on the same aspect. For DeepSeek Predict, two hotels drawn at random within the same city receive the same score on a given aspect with probability 11.8%; for Claude Joint, the corresponding figure is 4.8%. For a model instructed to assign scores with two-decimal precision in $[0, 1]$, these tie rates are notable.

They also complicate the interpretation of the models' wider standard deviations. Chapter 4 demonstrated that the model uses more of the full rating scale compared to humans. One might read this as the LLM scoring hotels on a finer scale than human raters do. That conclusion does not follow directly: a wider standard deviation measures how spread out the scores are, not how many distinct values the model actually uses to produce that spread.

This section therefore examines the effective resolution of the scoring instrument and its implications for interpreting small numerical differences between hotels.

7.1. Digit Anchoring in LLM Score Outputs

Figure 7.1 shows the distribution of the hundredths digit for each model-variant combination. DeepSeek concentrates almost entirely on digits 0 and 5: across its Joint and Predict variants, they are represented in 70–90% of all scores. Claude concentrates on the three digits 2, 5, and 8, which together account for 78–86%. The preferred digits are therefore model-specific, but in both cases the model operates on a much smaller effective range.

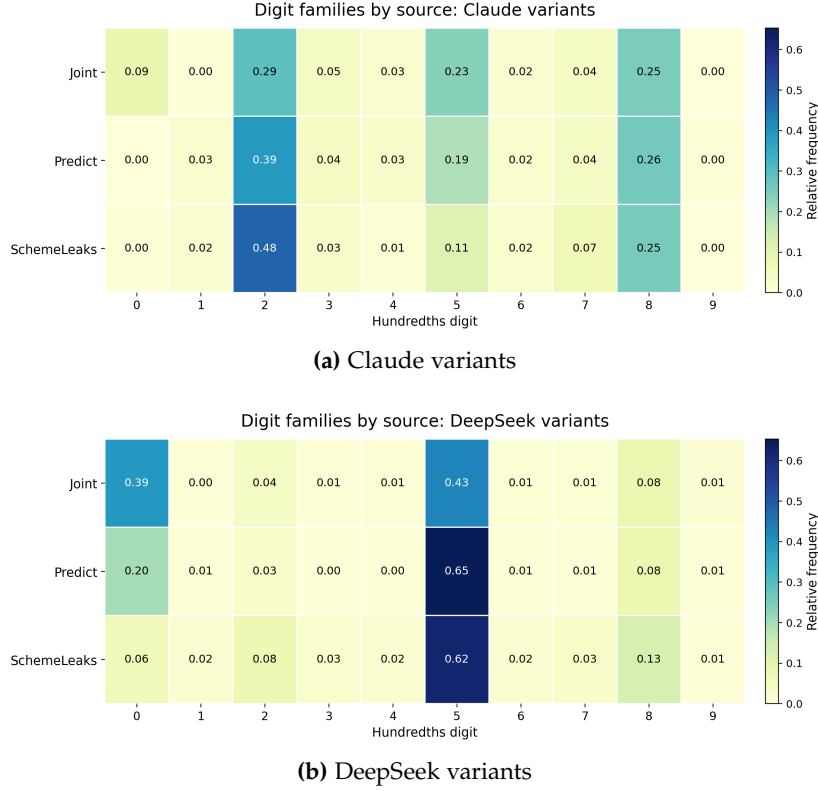


Figure 7.1.: Distribution of the hundredths digit for each model-variant combination.

The SchemaLeaks variant demonstrates that digit preferences are not simply rounding behaviour. Prompt variations or changes in the schema can affect the distribution of preferred digits.

The varying digit preferences are consistent with recent evidence observing numerical bias in LLMs, amplified by post-training [77], and with evidence that prompt context affects this digit bias [78].

7.2. Comparing Variants Formally

Measuring which variant is the most balanced depends on the metric chosen. We consider two interpretations.

7.2.1. Qualitative Analysis

Visually, the distributions in Figure 7.1 suggest that Joint is the most balanced and SchemaLeaks the most concentrated. But this visual comparison reflects only how evenly mass is distributed within each model’s preferred grid. It does not consider the redistribution of digits outside that preferred grid, which is precisely what distinguishes how the SchemaLeaks condition affects DeepSeek compared to Claude.

7.2.2. Last-Digit Entropy

This difference is considered when we compare the models more formally using the Shannon entropy of the last-digit distribution:

$$H_{\text{last}} = - \sum_{d=0}^9 p_d \log_2 p_d. \quad (7.1)$$

Lower values represent higher concentration on individual digits, and the quantity is maximised by a uniform distribution across all ten digits, at $\log_2 10 \approx 3.32$ bits. By this metric, DeepSeek-SchemaLeaks (2.00 bits) is the most balanced of DeepSeek’s three variants, above both DeepSeek-Joint (1.83) and DeepSeek-Predict (1.45). The mean of human numeric scores is close to the theoretical maximum (3.31). The visual and entropy comparison therefore answer different questions: one reflects how evenly the preferred grid is used, the other how much of the ten-digit space is used at all.

7.2.3. Full Value Entropy

Either comparison, however, considers only the last digit. To evaluate the precision of the scores as a measurement instrument, the relevant quantity is the entropy of the full two-digit value,

$$H_{\text{value}} = - \sum_{v \in \mathcal{V}} p_v \log_2 p_v, \quad \mathcal{V} = \{0.00, 0.01, \dots, 1.00\}, \quad (7.2)$$

with theoretical maximum $\log_2 101 \approx 6.66$ bits. In the preceding analysis, we focused only on the last digit because the tenths digit has mostly consistent entropy for both models: 2.71–2.95 bits across all six conditions, close to its own maximum of 3.32 bits. For comparison, the human entropy of the tens digit is worse at 2.2 bits. The differences between models and variants mostly reflect their last-digit precision. The differences between humans and LLMs mostly reflect their tens-digit precision.

Figure 7.2 compares the models based on their full-value entropy, shown on the right. To allow for more intuitive interpretation, the primary metric is the number of distinct scores needed to cover 95% of all scores that were generated in each case.

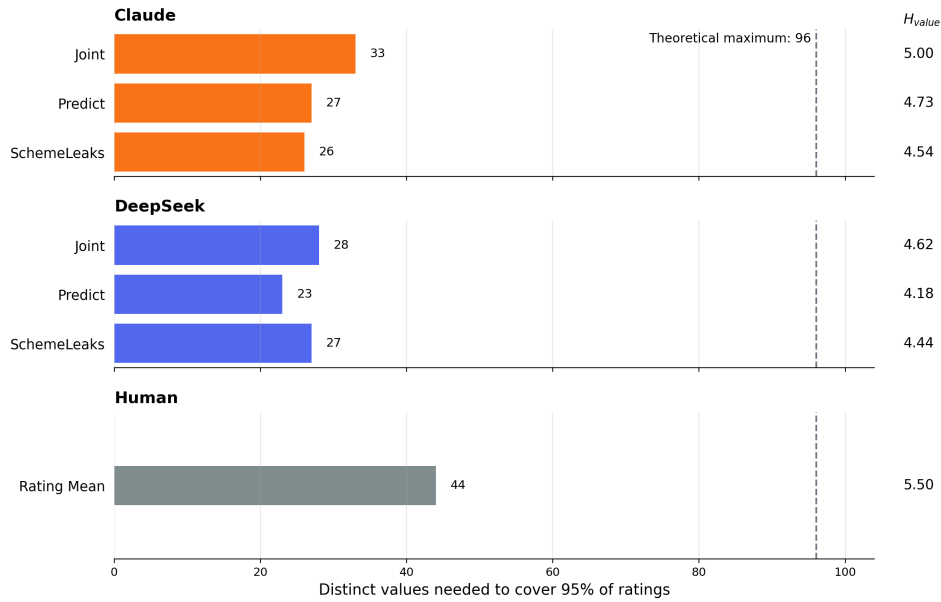


Figure 7.2.: Full-value entropy and number of distinct scores needed to cover 95% of generated scores.

Despite almost-perfect last-digit entropy, human scores are far below the theoretical optimal informativeness, due to their tendency to cluster ratings at the top end of the scale. Yet, they outperform the best LLM variant by about 30%.

7.3. Summary

The tie rates introduced at the start of the section are a direct consequence of these effective-alphabet sizes. Across the six model-variant combinations, the rank ordering of tie rates matches the rank ordering of H_{value} exactly across all six conditions. Ties do not represent low textual signal from the properties; they are a direct consequence of the model’s digit preference.

Two implications follow. First, despite the wider standard deviation of LLM scores, both tie rate and entropy demonstrate that averaged human numeric scores are the more fine-grained measurement instrument. Second, within-aspect numerical gaps smaller than roughly 0.05 for DeepSeek or 0.03 for Claude fall below the instrument’s effective resolution. Such gaps should be treated as indistinguishable rather than as small but meaningful differences in the underlying construct.

8. Conclusion

LLM-derived aspect scores are not simply better than numeric ratings. They are better for aspect differentiation and construct validity where textual evidence is sufficient, but not for reliability under sparse evidence and not for fine numerical resolution. The advantage is therefore context-dependent.

Hostelworld’s numeric baseline is highly halo-contaminated. On Hostelworld, around 40% of reviewers assign the same score to every aspect, the seven aspects correlate at an average of 0.77 across cities, and the first principal component explains 75% to 92% of total variance in seventeen of eighteen cities. The aspect scores therefore behave more like one general impression than seven distinct evaluations.

LLM scores are a more construct-valid alternative. LLM scores correlate moderately to highly with the numeric aspect ratings (between 0.71 and 0.93 across aspects), highlighting the potential as an alternative measurement instrument. Three independent tests show that they are also more construct-valid: First, the reduction in inter-aspect correlation is structured rather than uniform: physical qualities (cleanliness, facilities, security) remain comparatively connected, subjective qualities separate more strongly, and location decouples most clearly. The latter aligns with prior literature positioning location as an aspect that should not correlate strongly with others. Second, independent of these author-chosen clusters, the LLM’s correlation structure follows the same pattern observed among reviewers who use the rating scale more discriminatingly. Third, the LLM recovers the internal pattern of relative strengths and weaknesses within individual hotels.

Pipeline design determines how construct-valid the scores are. The pipeline ablation isolated three mechanisms that artificially inflate inter-aspect correlations: scoring aspects jointly, schema awareness, and instruction framing. The expected ordering Joint > SchemaLeaks > Predict > Rate held in almost all cities. Of the AIIC reduction between Joint and Predict, roughly 60% was attributable to scoring aspects individually and 40% to removing the model’s awareness of the broader evaluation framework. Prior work documents joint scoring as a source of criterion entanglement, but to our knowledge, the finding that revealing the broader evaluation framework in the output specification biases scores even when each aspect is scored in a separate call has not been tested elsewhere. The output schema is therefore part of the measurement instrument, not a neutral formatting layer.

Partial replication on a second model. DeepSeek V3.2 replicated the main construct-validity findings. However, its rerun-to-rerun standard deviation in Joint was approximately equal to the standard deviation across pipeline variants, so differences between Joint, SchemaLeaks, and Predict cannot be cleanly separated from stochastic variation. DeepSeek also shows operational failures not present for Claude: null emissions in which every requested score equals 0.0, and reduced instruction-following in SchemaLeaks (7.93% of calls failed at first pass). With filtering and validation in place, DeepSeek is a viable lower-cost alternative for hotel-level aggregation, but the granular pipeline ablation requires Claude.

Limitations of LLM-based scoring. Under reduced review evidence, the numeric mean converged toward its own large-sample reference more reliably than the LLM at every tested input size, with the gap concentrated in security and value for money: security tends to be discussed only when a concrete incident occurs, and value for money is expressed indirectly. The wider standard deviation of LLM scores also does not translate into finer-grained measurement. Both models anchored on a small set of preferred hundredths digits (Claude on 2, 5, and 8; DeepSeek on 0 and 5), and averaged human numeric scores have roughly 30% more entropy than the best LLM variant. Within-aspect differences below approximately 0.03 for Claude and 0.05 for DeepSeek therefore fall below the instrument’s effective resolution and should not be interpreted as meaningful.

8.1. Limitations

The thesis tests two models. The construct-validity claim generalises, but the contribution of schema awareness is established only on Claude, because DeepSeek’s rerun noise at the budget used here is too high to support the ablation. Whether the 60/40 decomposition holds for other models, on other platforms, or even in a non-hospitality context, remains open. The Joint condition uses a fixed aspect order in which security is generated first. Because security has the largest negative bias of any aspect in the Claude scores, the Joint AIIC may be partially shaped by this ordering, and a systematic variation of aspect order would yield a more robust estimate of the joint-scoring effect. The sampling-reliability analysis is limited to the Claude Joint condition. The rerun reliability analysis is also limited to the Joint condition. Other variants could exhibit larger noise. The choice of platform results in a lower generalisability of the results: Section 3.8.1 argues that Hostelworld’s homogeneous userbase, mandatory aspect ratings, and incentive structure make it especially well suited for this comparison, and the relative advantage of LLMs might disappear under different conditions.

8.2. Implications

For booking platforms. User-provided aspect scores systematically understate genuine differences between properties. LLM-derived scores could serve as a supplementary measurement layer for properties with sufficient review volume, surfacing relative strengths and weaknesses that halo obscures in the numeric ratings.

For LLM-as-judge research. Studies that use LLMs as multi-criterion evaluators should report the output schema alongside the prompt; treating the schema as a parsing detail makes the experiment possibly underspecified. Second, when isolation between criteria is important, schemas should be designed to not reveal all criteria.

8.3. Future Work

From the current pipeline design it cannot be concluded whether the schema awareness effect is mostly caused by the schema, or by awareness of the evaluation criteria within the system prompt. A future study should isolate this by keeping the schema constant and revealing information about the evaluation criteria only as part of the prompt. This would clarify whether the effect is driven by the presence of an evaluative framework, by the specific aspect names, or by the structure of the schema as such. Hybrid approaches that combine numeric ratings with text-derived scores could be tested as solution to offer higher construct-validity, even at low review counts.

The pipeline did not test whether a different rating scale could increase entropy among the LLM scores. Existing research already compares the effects of prompted scoring ranges on agreement with expert labels, but these labels are typically on a coarser scale than the numeric aspect ratings presented here. Hospitality offers a domain where this research could be extended to identify how the granularity of LLM-based scores can be enhanced.

A. Appendix

A.1. Search Queries

Query 1 looks to identify all LLM papers analysing aspect-ratings to identify papers that are very similar, but not in the hospitality domain. Query 2 focuses on all LLM papers in hospitality. For Google Scholar, only the first 100 results were included in each query.

```
TITLE-ABS-KEY("large language model" OR LLM OR GPT OR ChatGPT)
AND TITLE-ABS-KEY("aspect rating" OR "aspect score" OR "multi-aspect"
  OR "multi aspect" OR "aspect-level" OR "aspect level" OR "aspect-based"
  OR "aspect based" OR "multi-criteria" OR "attribute rating"
  OR "attribute score" OR "sub-rating" OR "sub rating")
AND TITLE-ABS-KEY(review OR reviews OR rating OR ratings)
```

```
TITLE-ABS-KEY("large language model" OR LLM OR GPT OR ChatGPT)
AND TITLE-ABS-KEY(hotel OR hospitality OR tourism OR hostel
  OR accommodation OR TripAdvisor OR Booking.com)
AND TITLE-ABS-KEY(review OR reviews)
```

A.2. Schema

Non-leaking schema

```
{
  "type": "object",
  "properties": {
    "results": {
      "type": "array",
      "description": "A list of aspect-rating pairs.",
      "items": {
        "type": "object",
        "properties": {
          "aspect": { "type": "string" },
          "rating": {
```

```
        "type": "number",
        "minimum": 0.0,
        "maximum": 1.0
      }
    },
    "required": ["aspect", "rating"],
    "additionalProperties": false
  }
},
"required": ["results"],
"additionalProperties": false
}
```

Leaking schema

```
{
  "type": "object",
  "properties": {
    "results": {
      "type": "object",
      "properties": {
        "security": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "location": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "staff": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "atmosphere": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "cleanliness": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "facilities": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "value_for_money": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "overall": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "bathrooms": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "party": { "type": "number", "minimum": 0.0, "maximum": 1.0 },
        "social": { "type": "number", "minimum": 0.0, "maximum": 1.0 }
      },
      "required": [],
      "additionalProperties": false
    }
  },
  "required": ["results"],
  "additionalProperties": false
}
```

List of Figures

4.1. Overall score calibration	20
4.2. Florence correlation heatmaps	22
4.3. Selective decorrelation	23
4.4. AIIC by scoring source and city	24
5.1. Clustered aspect correlations by model	30
5.2. Noise, variant, and hotel-level standard deviation	31
5.3. Caching effects by model and condition	34
5.4. Per-hotel inference costs	35
6.1. Reliability by input size	39
6.2. Per-aspect reliability	40
7.1. Last-digit distributions by model and variant	43
7.2. Full-value entropy and effective score alphabet	45

List of Tables

- 2.1. Summary of aspect-prediction studies 5
- 4.1. Per-aspect agreement metrics 20
- 4.2. Mean AIIC across cities by source 24
- 4.3. Aggregate trajectory alignment 26
- 4.4. Centred within-hotel profile agreement 27
- 5.1. Joint hotel-level replication 30
- 5.2. Null-row incidence 32
- 5.3. Post-validation outcomes in SchemaLeaks 33

Bibliography

- [1] H. Wang, Y. Lu, and C. Zhai. "Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach." In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '10: The 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington DC USA: ACM, July 25, 2010, pp. 783–792. ISBN: 978-1-4503-0055-1. DOI: 10.1145/1835804.1835903.
- [2] B. Lu, M. Ott, C. Cardie, and B. K. Tsou. "Multi-Aspect Sentiment Analysis with Topic Models." In: *2011 IEEE 11th International Conference on Data Mining Workshops*. 2011 IEEE International Conference on Data Mining Workshops (ICDMW). Vancouver, BC, Canada: IEEE, Dec. 2011, pp. 81–88. ISBN: 978-1-4673-0005-6 978-0-7695-4409-0. DOI: 10.1109/ICDMW.2011.125.
- [3] H. Wang and M. Ester. "A Sentiment-aligned Topic Model for Product Aspect Rating Prediction." In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1192–1202. DOI: 10.3115/v1/D14-1126.
- [4] W. Xue, T. Li, and N. Rishe. "Aspect Identification and Ratings Inference for Hotel Reviews." In: *World Wide Web 20.1* (Jan. 2017), pp. 23–37. ISSN: 1386-145X, 1573-1413. DOI: 10.1007/s11280-016-0398-9.
- [5] Y. Wang, Y. Huang, and M. Wang. "Aspect-Based Rating Prediction on Reviews Using Sentiment Strength Analysis." In: *Advances in Artificial Intelligence: From Theory to Practice*. Ed. by S. Benferhat, K. Tabia, and M. Ali. Vol. 10351. Cham: Springer International Publishing, 2017, pp. 439–447. ISBN: 978-3-319-60044-4 978-3-319-60045-1. DOI: 10.1007/978-3-319-60045-1_45.
- [6] A. Laddha and A. Mukherjee. "Aspect Opinion Expression and Rating Prediction via LDA–CRF Hybrid." In: *Natural Language Engineering 24.4* (July 2018), pp. 611–639. ISSN: 1351-3249, 1469-8110. DOI: 10.1017/S135132491800013X.
- [7] E. Hasan and C. Ding. "Aspect-Aware Multi-Criteria Recommendation Model with Aspect Representation Learning." In: *2023 IEEE International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. 2023 IEEE International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT). Venice, Italy: IEEE, Oct. 26, 2023, pp. 268–272. ISBN: 979-8-3503-0918-8. DOI: 10.1109/WI-IAT59888.2023.00043.

- [8] K. Schouten and F. Frasincar. “Survey on Aspect-Level Sentiment Analysis.” In: *IEEE Transactions on Knowledge and Data Engineering* 28.3 (Mar. 1, 2016), pp. 813–830. issn: 1041-4347. doi: 10.1109/TKDE.2015.2485209.
- [9] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu, S. Wang, K. Zhang, Y. Wang, W. Gao, L. Ni, and J. Guo. *A Survey on LLM-as-a-Judge*. Oct. 19, 2025. doi: 10.48550/arXiv.2411.15594. arXiv: 2411.15594 [cs]. URL: <http://arxiv.org/abs/2411.15594> (visited on 05/02/2026). Pre-published.
- [10] D. Li, B. Jiang, L. Huang, A. Beigi, C. Zhao, Z. Tan, A. Bhattacharjee, Y. Jiang, C. Chen, T. Wu, K. Shu, L. Cheng, and H. Liu. “From Generation to Judgment: Opportunities and Challenges of LLM-as-a-judge.” In: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. Suzhou, China: Association for Computational Linguistics, 2025, pp. 2757–2791. doi: 10.18653/v1/2025.emnlp-main.138.
- [11] K. Chehbouni, M. Haddou, J. C. K. Cheung, and G. Farnadi. *Neither Valid nor Reliable? Investigating the Use of LLMs as Judges*. Version 2. 2025. doi: 10.48550/ARXIV.2508.18076. URL: <https://arxiv.org/abs/2508.18076> (visited on 05/01/2026). Pre-published.
- [12] A. Bavaresco, R. Bernardi, L. Bertolazzi, D. Elliott, R. Fernández, A. Gatt, E. Ghaleb, M. Giulianelli, M. Hanna, A. Koller, A. Martins, P. Mondorf, V. Neplenbroek, S. Pezzelle, B. Plank, D. Schlangen, A. Suglia, A. K. Surikuchi, E. Takmaz, and A. Testoni. “LLMs Instead of Human Judges? A Large Scale Empirical Study across 20 NLP Evaluation Tasks.” In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Vienna, Austria: Association for Computational Linguistics, 2025, pp. 238–255. doi: 10.18653/v1/2025.acl-short.20.
- [13] L. Gierbach, C.-P. Su, T. Saanum, R. Socher, E. Schulz, and Z. Akata. *Reference-Free Rating of LLM Responses via Latent Information*. Sept. 29, 2025. doi: 10.48550/arXiv.2509.24678. arXiv: 2509.24678 [cs]. URL: <http://arxiv.org/abs/2509.24678> (visited on 05/02/2026). Pre-published.
- [14] A. Sahoo, J. K. Karnuthala, T. P. Budhwani, P. Agarwal, S. Vaidyanathan, A. Siu, F. Dernoncourt, J. Healey, N. Lipka, R. Rossi, U. Bhattacharya, and B. Kveton. *Quantitative LLM Judges*. Oct. 23, 2025. doi: 10.48550/arXiv.2506.02945. arXiv: 2506.02945 [cs]. URL: <http://arxiv.org/abs/2506.02945> (visited on 05/02/2026). Pre-published.
- [15] R. Stureborg, D. Alikaniotis, and Y. Suhara. *Large Language Models Are Inconsistent and Biased Evaluators*. May 2, 2024. doi: 10.48550/arXiv.2405.01724. arXiv: 2405.01724 [cs]. URL: <http://arxiv.org/abs/2405.01724> (visited on 05/02/2026). Pre-published.
- [16] K. I. Roumeliotis, N. D. Tselikas, and D. K. Nasiopoulos. “Leveraging Large Language Models in Tourism: A Comparative Study of the Latest GPT Omni Models and BERT NLP for Customer Review Classification and Sentiment Analysis.” In: *Information* 15.12 (Dec. 10, 2024), p. 792. issn: 2078-2489. doi: 10.3390/info15120792.

- [17] P. Nnanna, O. Amujo, C. P. Ezenkwu, and E. Ibeke. "Leveraging LLMs for User Rating Prediction from Textual Reviews: A Hospitality Data Annotation Case Study." In: *Information* 16.12 (Dec. 2, 2025), p. 1059. ISSN: 2078-2489. DOI: 10.3390/info16121059.
- [18] A. J. W. de Vink, N. Amat-Lefort, and L. Han. *ReviewGraph: A Knowledge Graph Embedding Based Framework for Review Rating Prediction with Sentiment Features*. Nov. 15, 2025. DOI: 10.48550/arXiv.2508.13953. arXiv: 2508.13953 [cs]. URL: <http://arxiv.org/abs/2508.13953> (visited on 05/02/2026). Pre-published.
- [19] S. Zhao and G. Jiang. "Enhancing Recommender Systems with LLM-extracted Explicit Features." In: *Knowledge and Information Systems* 68.1 (Dec. 2026), p. 24. ISSN: 0219-1377, 0219-3116. DOI: 10.1007/s10115-025-02665-2.
- [20] Y. Yang, Y.-Y. Guo, and G. Ren. "Decomposing Preferences with a Large Language Model for Accurate and Interpretable Rating Prediction." In: *Information Sciences* 748 (Aug. 2026), p. 123525. ISSN: 00200255. DOI: 10.1016/j.ins.2026.123525.
- [21] S. Wu, J. Yuan, X. Zhang, Z. Liu, and P. Zhang. "Temporal-Aware and Aspect Sentiment Driven Recommender Model for Rating Prediction." In: *Proceedings of the 2025 2nd International Conference on Digital Society and Artificial Intelligence*. DSAI 2025: 2nd International Conference on Digital Society and Artificial Intelligence. Wuhan China: ACM, May 23, 2025, pp. 485–491. ISBN: 979-8-4007-1433-7. DOI: 10.1145/3748825.3748901.
- [22] T. N. Soe and K. Punyachalard. "Meaningful Hotel Recommendation Based on Hotel Review Text." MA thesis. Sweden: Dalarna University, June 9, 2025.
- [23] V. Patil, S. V. Bacha, R. Yamani, Y. Sun, and M. Kejriwal. *Beyond the Star Rating: A Scalable Framework for Aspect-Based Sentiment Analysis Using LLMs and Text Classification*. Feb. 24, 2026. DOI: 10.48550/arXiv.2602.21082. arXiv: 2602.21082 [cs]. URL: <http://arxiv.org/abs/2602.21082> (visited on 05/02/2026). Pre-published.
- [24] I. W. McMurry. "Quantifying Social Sentiment in Hostels Using A Domain-Specific Transformer Pipeline." In: *The Proceedings for the 15th Workshop on Computational Approaches to Subjectivity, Sentiment Social Media Analysis (WASSA 2026)*. The Proceedings for the 15th Workshop on Computational Approaches to Subjectivity, Sentiment Social Media Analysis (WASSA 2026). Rabat, Morocco: Association for Computational Linguistics, 2026, pp. 24–36. DOI: 10.18653/v1/2026.wassa-1.3.
- [25] S. Morini-Marrero, J. M. Ramos-Henriquez, and A. Bilgihan. "Analyzing the Concordance and Consistency of AI and Human Ratings in Hospitality Reviews." In: *Journal of Hospitality and Tourism Technology* 16.5 (Nov. 4, 2025), pp. 937–965. ISSN: 1757-9880, 1757-9899. DOI: 10.1108/JHTT-04-2024-0251.
- [26] J. Liu. "Rubric-Conditioned Large Language Model Labeling: Agreement, Uncertainty, and Label Consistency in Subjective Text Annotation." In: *Computers in Human Behavior* 181 (Aug. 2026), p. 108988. ISSN: 07475632. DOI: 10.1016/j.chb.2026.108988.
- [27] J. Liu. *Learning to Fuse Aspect-Level LLM Annotations for Low-Quality Ordinal Sentiment Supervision*. 2026. DOI: 10.2139/ssrn.6244519. URL: <https://www.ssrn.com/abstract=6244519> (visited on 05/02/2026). Pre-published.

- [28] Y. Wang, J. Huang, L. Du, Y. Guo, Y. Liu, and R. Wang. "Evaluating Large Language Models as Raters in Large-Scale Writing Assessments: A Psychometric Framework for Reliability and Validity." In: *Computers and Education: Artificial Intelligence* 9 (Dec. 2025), p. 100481. issn: 2666920X. doi: 10.1016/j.caeai.2025.100481.
- [29] Z. Xiao, S. Zhang, V. Lai, and Q. V. Liao. "Evaluating Evaluation Metrics: A Framework for Analyzing NLG Evaluation Metrics Using Measurement Theory." In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Singapore: Association for Computational Linguistics, 2023, pp. 10967–10982. doi: 10.18653/v1/2023.emnlp-main.676.
- [30] D. T. Campbell and D. W. Fiske. "Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix." In: *Psychological Bulletin* 56.2 (1959), pp. 81–105. issn: 1939-1455, 0033-2909. doi: 10.1037/h0046016.
- [31] S. Messick. "Validity of Psychological Assessment: Validation of Inferences from Persons' Responses and Performances as Scientific Inquiry into Score Meaning." In: *American Psychologist* 50.9 (Sept. 1995), pp. 741–749. issn: 1935-990X, 0003-066X. doi: 10.1037/0003-066X.50.9.741.
- [32] G. A. Ferguson. "On the Theory of Test Discrimination." In: *Psychometrika* 14.1 (Mar. 1949), pp. 61–68. issn: 0033-3123, 1860-0980. doi: 10.1007/BF02290141.
- [33] M. Hankins. "Questionnaire Discrimination: (Re)-Introducing Coefficient δ ." In: *BMC Medical Research Methodology* 7.1 (Dec. 2007), p. 19. issn: 1471-2288. doi: 10.1186/1471-2288-7-19.
- [34] M. Hankins. "How Discriminating Are Discriminative Instruments?" In: *Health and Quality of Life Outcomes* 6.1 (2008), p. 36. issn: 1477-7525. doi: 10.1186/1477-7525-6-36.
- [35] T. Radojevic, N. Stanisic, and N. Stanic. "Inside the Rating Scores: A Multilevel Analysis of the Factors Influencing Customer Satisfaction in the Hotel Industry." In: *Cornell Hospitality Quarterly* 58.2 (May 2017), pp. 134–164. issn: 1938-9655, 1938-9663. doi: 10.1177/1938965516686114.
- [36] S. Cicognani, P. Figini, and M. Magnani. "Social Influence Bias in Ratings: A Field Experiment in the Hospitality Sector." In: *Tourism Economics* 28.8 (Dec. 2022), pp. 2197–2218. issn: 1354-8166, 2044-0375. doi: 10.1177/13548166211034645.
- [37] J. L. Nicolau, J. P. Mellinas, and E. Martín-Fuentes. "The Halo Effect: A Longitudinal Approach." In: *Annals of Tourism Research* 83 (July 2020), p. 102938. issn: 01607383. doi: 10.1016/j.annals.2020.102938.
- [38] X. Liu, C. Li, J. L. Nicolau, and M. Han. "The Value of Rating Diversity within Multidimensional Rating System: Evidence from Hotel Booking Platform." In: *International Journal of Hospitality Management* 110 (Apr. 2023), p. 103434. issn: 02784319. doi: 10.1016/j.ijhm.2023.103434.

- [39] J. Wirtz. "Improving the Measurement of Customer Satisfaction: A Test of Three Methods to Reduce Halo." In: *Managing Service Quality: An International Journal* 11.2 (Apr. 1, 2001), pp. 99–112. ISSN: 0960-4529. DOI: 10.1108/09604520110387239.
- [40] J. Wirtz. "Halo in Customer Satisfaction Measures: The Role of Purpose of Rating, Number of Attributes and Customer Involvement." In: *International Journal of Service Industry Management* 14.1 (Mar. 1, 2003), pp. 96–119. ISSN: 0956-4233. DOI: 10.1108/09564230310466001.
- [41] G. Zervas, D. Proserpio, and J. W. Byers. "A First Look at Online Reputation on Airbnb, Where Every Stay Is above Average." In: *Marketing Letters* 32.1 (Mar. 2021), pp. 1–16. ISSN: 0923-0645, 1573-059X. DOI: 10.1007/s11002-020-09546-4.
- [42] M. M. Mariani and M. Borghi. "Effects of the Booking.Com Rating System: Bringing Hotel Class into the Picture." In: *Tourism Management* 66 (June 2018), pp. 47–52. ISSN: 02615177. DOI: 10.1016/j.tourman.2017.11.006.
- [43] N. Hu, J. Zhang, and P. A. Pavlou. "Overcoming the J-shaped Distribution of Product Reviews." In: *Communications of the ACM* 52.10 (Oct. 2009), pp. 144–147. ISSN: 0001-0782, 1557-7317. DOI: 10.1145/1562764.1562800.
- [44] J. P. Mellinas, S.-M. Martínez María-Dolores, and J. J. Bernal García. "Booking.Com: The Unexpected Scoring System." In: *Tourism Management* 49 (Aug. 2015), pp. 72–74. ISSN: 02615177. DOI: 10.1016/j.tourman.2014.08.019.
- [45] V. Leoni and D. Boto-García. "'Apparent' and Actual Hotel Scores under Booking.Com New Reviewing System." In: *International Journal of Hospitality Management* 111 (May 2023), p. 103493. ISSN: 02784319. DOI: 10.1016/j.ijhm.2023.103493.
- [46] M. Sclar, Y. Choi, Y. Tsvetkov, and A. Suhr. *Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I Learned to Start Worrying about Prompt Formatting*. July 1, 2024. DOI: 10.48550/arXiv.2310.11324. arXiv: 2310.11324 [cs]. URL: <http://arxiv.org/abs/2310.11324> (visited on 05/02/2026). Pre-published.
- [47] J. Zhuo, S. Zhang, X. Fang, H. Duan, D. Lin, and K. Chen. "ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs." In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. Findings of the Association for Computational Linguistics: EMNLP 2024. Miami, Florida, USA: Association for Computational Linguistics, 2024, pp. 1950–1976. DOI: 10.18653/v1/2024.findings-emnlp.108.
- [48] Y. Hwang, D. Lee, T. Kang, M. Lee, and K. Jung. *When Wording Steers the Evaluation: Framing Bias in LLM Judges*. Jan. 20, 2026. DOI: 10.48550/arXiv.2601.13537. arXiv: 2601.13537 [cs]. URL: <http://arxiv.org/abs/2601.13537> (visited on 05/02/2026). Pre-published.
- [49] Q. Zhang, J. Zhou, Y. Wang, F. Lyu, Y. Ming, C. Xu, Q. Sun, K. Zheng, P. Kang, X. Liu, and C. Ma. *RubricBench: Aligning Model-Generated Rubrics with Human Standards*. Mar. 3, 2026. DOI: 10.48550/arXiv.2603.01562. arXiv: 2603.01562 [cs]. URL: <http://arxiv.org/abs/2603.01562> (visited on 05/02/2026). Pre-published.

- [50] Y. Feng, S. Wang, Z. Cheng, Y. Wan, and D. Chen. *Are We on the Right Way to Assessing LLM-as-a-Judge?* Dec. 17, 2025. doi: 10.48550/arXiv.2512.16041. arXiv: 2512.16041 [cs]. URL: <http://arxiv.org/abs/2512.16041> (visited on 05/02/2026). Pre-published.
- [51] H. Wei, S. He, T. Xia, F. Liu, A. Wong, J. Lin, and M. Han. *Systematic Evaluation of LLM-as-a-Judge in LLM Alignment Tasks: Explainable Metrics and Diverse Prompt Templates.* Mar. 30, 2025. doi: 10.48550/arXiv.2408.13006. arXiv: 2408.13006 [cs]. URL: <http://arxiv.org/abs/2408.13006> (visited on 05/02/2026). Pre-published.
- [52] R. Haldar and J. Hockenmaier. “Rating Roulette: Self-Inconsistency in LLM-As-A-Judge Frameworks.” In: *Findings of the Association for Computational Linguistics: EMNLP 2025*. Findings of the Association for Computational Linguistics: EMNLP 2025. Suzhou, China: Association for Computational Linguistics, 2025, pp. 24986–25004. doi: 10.18653/v1/2025.findings-emnlp.1361.
- [53] N. C. Schaeffer and S. Presser. “The Science of Asking Questions.” In: *Annual Review of Sociology* 29.1 (Aug. 2003), pp. 65–88. issn: 0360-0572, 1545-2115. doi: 10.1146/annurev.soc.29.110702.110112.
- [54] W. Li, X. Wang, S. Yuan, R. Xu, J. Chen, Q. Dong, Y. Xiao, and D. Yang. “Curse of Knowledge: Your Guidance and Provided Knowledge Are Biasing LLM Judges in Complex Evaluation.” In: *Findings of the Association for Computational Linguistics: EMNLP 2025*. Findings of the Association for Computational Linguistics: EMNLP 2025. Suzhou, China: Association for Computational Linguistics, 2025, pp. 14900–14924. doi: 10.18653/v1/2025.findings-emnlp.805.
- [55] H. Hashemi, J. Eisner, C. Rosset, B. Van Durme, and C. Kedzie. “LLM-Rubric: A Multidimensional, Calibrated Approach to Automated Evaluation of Natural Language Texts.” In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Bangkok, Thailand: Association for Computational Linguistics, 2024, pp. 13806–13834. doi: 10.18653/v1/2024.acl-long.745.
- [56] N. Lee, J. Hong, and J. Thorne. *Evaluating the Consistency of LLM Evaluators*. Nov. 30, 2024. doi: 10.48550/arXiv.2412.00543. arXiv: 2412.00543 [cs]. URL: <http://arxiv.org/abs/2412.00543> (visited on 05/02/2026). Pre-published.
- [57] Anthropic. *Tool Use with Claude*.
- [58] Amazon Web Services. *Tool Use — Amazon Bedrock User Guide*.
- [59] M. Sclar, Y. Choi, Y. Tsvetkov, and A. Suhr. “Quantifying Language Models’ Sensitivity to Spurious Features in Prompt Design or: How I Learned to Start Worrying about Prompt Formatting.” In: *The Twelfth International Conference on Learning Representations (ICLR)*. 2024. arXiv: 2310.11324 [cs.CL].
- [60] Amazon Web Services. *Structured Outputs Now Available in Amazon Bedrock*. Feb. 2026.
- [61] OpenRouter. *Structured Outputs — OpenRouter Documentation*.

- [62] DeepSeek. *JSON Output — DeepSeek API Docs*.
- [63] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang. “Lost in the Middle: How Language Models Use Long Contexts.” In: *Transactions of the Association for Computational Linguistics* 12 (2024), pp. 157–173. doi: 10.1162/tacl_a_00638.
- [64] M. Levy, A. Jacoby, and Y. Goldberg. “Same Task, More Tokens: The Impact of Input Length on the Reasoning Performance of Large Language Models.” In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Bangkok, Thailand: Association for Computational Linguistics, 2024, pp. 15339–15353. doi: 10.18653/v1/2024.acl-long.818.
- [65] Amazon Web Services. *Effectively Use Prompt Caching on Amazon Bedrock*.
- [66] OpenAI. *GPT-4.1 Prompting Guide — OpenAI Cookbook*.
- [67] Anthropic. *Claude Sonnet 4.6*. Feb. 17, 2026. URL: <https://www.anthropic.com/claude/sonnet> (visited on 05/04/2026).
- [68] OpenRouter. *DeepSeek: DeepSeek V3.2 - API Pricing and Providers*. 2026. URL: <https://openrouter.ai/deepseek/deepseek-v3.2> (visited on 05/04/2026).
- [69] OpenRouter. *Anthropic: Claude Sonnet 4.6 - API Pricing and Providers*. 2026. URL: <https://openrouter.ai/anthropic/claude-sonnet-4.6> (visited on 05/04/2026).
- [70] Anthropic. *Pricing - Claude API Docs*. 2026. URL: <https://platform.claude.com/docs/en/about-claude/pricing> (visited on 05/04/2026).
- [71] Anthropic. *Prompt Caching - Claude API Docs*. 2026. URL: <https://platform.claude.com/docs/en/build-with-claude/prompt-caching> (visited on 05/04/2026).
- [72] DeepSeek. *Context Caching*. 2026. URL: <https://api-docs.deepseek.com/guides/kv-cache> (visited on 05/04/2026).
- [73] Amazon Web Services. *Process Multiple Prompts with Batch Inference - Amazon Bedrock User Guide*. 2026. URL: <https://docs.aws.amazon.com/bedrock/latest/userguide/batch-inference.html> (visited on 05/04/2026).
- [74] Amazon Web Services. *Amazon Bedrock Pricing*. 2026. URL: <https://aws.amazon.com/bedrock/pricing/> (visited on 05/04/2026).
- [75] Amazon Web Services. *Prompt Caching for Faster Model Inference - Amazon Bedrock User Guide*. 2026. URL: <https://docs.aws.amazon.com/bedrock/latest/userguide/prompt-caching.html> (visited on 05/04/2026).
- [76] Amazon Web Services. *Create a Batch Inference Job - Amazon Bedrock User Guide*. 2026. URL: <https://docs.aws.amazon.com/bedrock/latest/userguide/batch-inference-create.html> (visited on 05/04/2026).
- [77] A. Sato, H. Kim, Z. Chen, M. Mita, and M. Komachi. *Exploring the Effects of Alignment on Numerical Bias in Large Language Models*. Version 2. 2026. doi: 10.48550/ARXIV.2601.16444. URL: <https://arxiv.org/abs/2601.16444> (visited on 05/03/2026). Pre-published.

- [78] Y. Dai. *Rescaling Confidence: What Scale Design Reveals About LLM Metacognition*. Version 1. 2026. DOI: 10.48550/ARXIV.2603.09309. URL: <https://arxiv.org/abs/2603.09309> (visited on 05/03/2026). Pre-published.