IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

AI-Powered Location Review Analyzer: A Multi-Stage LLM Framework with Authenticity Guardrails for Place Recommendation

Ms. V. M. Dilpak, Ritesh Korade, Deep Pawar, Chinmay Kalokhe
Dept. of Information Technology
AISSMS Institute of Information Technology, Pune, India

Abstract: *Online maps and review sites significantly impact food choices, shopping, and travel destinations, although raw star ratings conceal aspect-level information and may be susceptible to faked, plagiarized, or coordinated reviews. Manual review of many individual reviews is inefficient, while most sentiment analysis pipelines ignore the reliability of the input reviews. This work introduces the Location Review AI Analyzer, an end-to-end solution that converts a location name or a link from Google Maps to a data-driven visit recommendation. The framework scrapes the latest reviews for a location from Google Maps via Apify, analyzes aspects, sentiment, emotions, themes, and time trends using a Llama-3.3-70B language model on Groq, conducts an audit of the reviews using a guardrail that integrates seven deterministic heuristics and a language model forensic analysis, and then synthesizes a recommendation based on the findings. The results are presented to the user via a Streamlit application and JSON file export. We evaluate the framework through a live case study (11 reviews of a restaurant with 7,383 reviews on Google Maps) and a controlled simulation of the guardrail layer (500 trials per scenario). The heuristic layer did not flag any reviews in natural control scenarios but successfully detected the injection of manipulation at a level of 99.2%–100%, depending on the share of the manipulations, in all tested scenarios. We also show inconsistencies in the results of aggregates computed by the language model, which further supports the use of deterministic computations. Limitations of the experiment include small sample sizes, lack of calibrated thresholds, and the lack of ground truth labels.*

Keywords: review analysis, aspect-based sentiment analysis, large language models, fake review detection, guardrails, recommendation systems, Google Maps

I. INTRODUCTION

Nowadays online reviews have become one of the main sources of information used to select a restaurant, a hotel, or any other tourist destination. Many research efforts in opinion mining are aimed at extracting sentiments and product features [1], [2]. However, in reality, a customer faced with a facility having thousands of reviews will see nothing but a star rating on average and several latest comments. Star ratings fail to reveal the specific aspects (food quality, service, cleanliness, etc.) contributing to the overall satisfaction rate and can be either positively or negatively biased using fake or manipulated reviews [3], [4].

Today, the progress in large language models makes possible the automated analysis of a bunch of reviews and producing structured judgements without task-specific training of the model [5], [6]. Still, there is another issue associated with employing the LLM for the analysis of reviews – the model will be happy to summarize anything, even some fake or copied reviews. Therefore, it is important to introduce a data verification stage prior to the summarization.

In this paper, we present the Location Review AI Analyzer pipeline having four stages: scraping, sentiment analysis, review audit, and recommendation. The design idea is to separate judgements and measurements. The deterministic

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

Python code calculates rating distribution and 7 manipulation checks based on rules, while the language model interprets language, aspects, and inconsistencies. The main contributions are:

An end-to-end, free-tier pipeline from place name or Google Maps URL to a recommendation, with live place search and URL expansion.

A guardrail module that merges seven deterministic manipulation heuristics with an LLM forensic analysis and reports a trust score, fake-review probability, and an adjusted rating.

A controlled synthetic evaluation of the heuristic layer, including attack-intensity sweeps and false-positive checks, and a live case study that documents inconsistencies in LLM-generated aggregates.

II. MOTIVATION

There are three things that inspired me to pursue this project. First, the star rating system is quite coarse-grained. Two establishments with the same average of 4.1 stars can actually be drastically different from each other when it comes to their service quality, price, and cleanliness, which can only be discerned from the review text. Second, review manipulation is common enough to have commercial significance [3]; thus, a recommendation without authentication is already manipulated. Third, current solutions focus on solving either one of the issues; either the text is scraped, analyzed for sentiments, or checked for spam, but it's up to the user to tie them together.

The aim of the project is to bring all of this into one pipeline that is cheap to run (uses free API tiers), transparent (all intermediate scores can be seen in the dashboard and JSON report), and straightforward about uncertainty (trust score and data reliability label come along with every result).

III. RELATED WORK

Sentiment analysis of reviews started with lexicon and feature-based approaches that computed the sentiment towards product features [1], [2]. ABSA formalized the problem of aspect identification and polarity detection in the form of a shared task on SemEval-2014 [7]; a recent survey traces the development from rule-based approaches to neural networks [8]. Pre-trained transformer models like BERT [9] boosted ABSA performance but require supervised training on domain-specific data.

Recent works address the question if LLMs can replace such supervised models. The empirical research revealed that LLMs demonstrate good performance on the sentiment analysis in zero-shot and few-shot scenarios but fail on more structured tasks [10]. This result makes it reasonable to use LLM for flexible extraction of aspects while checking the numerical results produced by the model deterministically, as we do here.

Deception in the reviews is studied since Jindal and Liu showed that there is a significant amount of duplicate or near-duplicate reviews [4]. Ott et al. created the gold-standard collection of fake hotel reviews and learned a classifier for deception detection based on this data [11]. Mukherjee et al. studied Yelp's filtering mechanism and discovered the role of behavioral signals such as reviewing patterns [12]. Recently, language models were used to generate plausible fake reviews [13]. Such progress requires more sophisticated approach to fraud detection and suggests considering both linguistic and behavioral signals.

Agentic and multi-stage systems divide the task into subtasks performed by LLM [14]. Prompting methods that force LLM to produce structured and reasoned output help improving the performance on more complex tasks [15]. Our pipeline is designed according to such principles with separated sentiment, guardrail, and recommendation stages, where the output of the previous stage serves as input for the next one. Table I summarizes the relevant papers and the gap our framework tries to fill.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

TABLE I: Comparison of Related Work

Ref.	Contribution	Gap relevant to this work
[1] Pang & Lee	Survey of opinion mining and sentiment analysis.	Pre-LLM; does not address review authenticity or end-user recommendation.
[2] Hu & Liu	Feature-level summaries of customer reviews.	No integrity check on the reviews themselves.
[4] Jindal & Liu	Large-scale study of opinion spam; duplicate-based detection.	E-commerce focus; not combined with sentiment or recommendation.
[11] Ott et al.	Gold-standard deceptive hotel reviews; text classifiers.	Needs labelled data; single domain; not a deployed pipeline.
[12] Mukherjee et al.	Behavioural analysis of Yelp's review filter.	Offline study; no sentiment or recommendation layer.
[7], [8] ABSA	Aspect extraction and polarity benchmarks and survey.	Fixed aspect sets; need annotated data or fine-tuning.
[10] Zhang et al.	Evaluation of LLMs on sentiment tasks.	Benchmark study; no end-to-end review-to-decision system.
[13] Salminen et al.	Generation and detection of fake reviews with language models.	Detection only; not integrated with place recommendation.

IV. PROBLEM STATEMENT AND METHODOLOGY

A. Problem Statement

For a place p denoted either by name or by URL, and a collection of n scrapped reviews R = {r1, …, rn} with star ratings, text content, authors, and dates, the system should generate (i) sentiment analysis of p on aspects basis, (ii) trustworthiness of R evaluation, and (iii) recommendation label with visit score and justifications. The input data can be noisy: there can be multiple interpretations for names, few reviews can exist, and reviews can be artificial.

B. Score Definitions

Various measures are calculated using the deterministic approach based on the original data. Star rating of the scraped sample is simply calculated instead of requesting that value from the LLM. For the guardrail heuristics, let wi be the set of words in review i. The similarity between two reviews is the Jaccard index:

$$J(i, j) = |wi \cap wj| / |wi \cup wj| \quad (1)$$

Author diversity and date diversity are given by:

$$Da = |\text{unique authors}| / n \quad (2)$$

$$Dt = |\text{unique dates}| / |\text{dated reviews}| \quad (3)$$

If the composite score is not provided by the LLM, then we fallback to the weighted combination of sentiment, rating, and trust, with all of them being scaled to the 0–10 range:

$$C = 0.40\ S + 0.40\ R + 0.20\ T \quad (4)$$

where S equals ten times the sentiment score, R equals twice the mean scraped star rating, and T equals ten times the trust score. The sentiment score is in the [−1, +1] range, while the trust score is in the [0, 1] range.

V. PROPOSED FRAMEWORK

This pipeline consists of four stages, preceded by the resolution of input and followed by a presentation layer (Figure 1). Each stage processes the structured output of its predecessor. The call to the language model is made using a helper

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

function which calls the model once for JSON, removes markdown fences, parses the outermost JSON object through pattern-matching, and retries once on parsing failure; if there is no success at all, it returns an empty output.

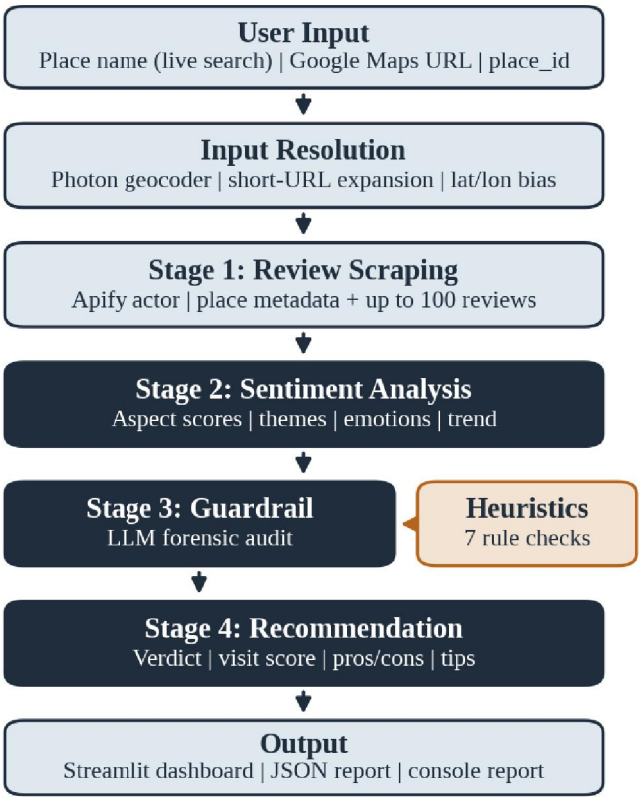

Fig. 1. Architecture of the Location Review AI Analyzer. Dark boxes are LLM-based stages; the rule-based heuristics feed the guardrail stage.

A. Input Resolution

The user can search by name or enter a link from Google Maps. In case of name search, live suggestions for the city box and place box appear as the user types using the Photon geocoder based on OpenStreetMap [16]. Place suggestions will be biased in favor of the selected city as the city will be embedded in the query and its coordinates provided. Shortened links in URL mode are resolved by following HTTP redirects. The third possible input, prefix of the Google place_id, is sent to the scraper directly. If it is a free-text query, the geocoding process is performed and the obtained coordinates are passed to the scraper together with the specified zoom level.

B. Review Acquisition

Reviews are acquired with the help of Apify Google Maps scraper actor [17] with the newest reviews requested in English in descending order limited by a user-specified limit (from 10 to 100 with a default value of 30). Up to three places are selected as candidates for the free-text query and the one which has the largest number of Google reviews is kept assuming that it is the intended landmark. Metadata about the place (name, address, category, aggregate rating, number of reviews, opening/closure times) is kept. Reviews without text are discarded while review text is cut off after 800 characters.

C. Sentiment Analysis Module

The module requests up to 30 reviews from Llama-3.3-70B (Groq [18]) using a prompt that includes an exact JSON schema. In order to capture the temporal dynamics in the data, the sample is segmented into a recent third and the rest

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

of it before applying the prompt. The schema expects: per-review sentiment, score, emotion, intensity and key phrase; seven aspect scores (from food quality to crowd/waiting time) in {0,10} and null if not mentioned in the review; recurring themes together with their frequency and a sample phrase; sentiment in the recent period vs. the rest of it and the trend label; crowd profile; emotion distribution; keyword lists; and overall sentiment in [−1, +1]. The star rating distribution is computed programmatically and returned along with the results. If the overall score is missing, it defaults to the average of all individual scores.

D. Guardrail Module

The guardrail starts off with a sequence of seven deterministic heuristics (Table II). The tests apply to rating skew, filler reviews, text copy-pasting, repeating phrases, low author diversity, small sample size, and clustered dates of publication. Pairwise checks and n-grams are done with a maximum of 20 reviews.

TABLE II: Deterministic Heuristic Checks in the Guardrail

ID	Check	Flag condition
H1	Rating skew	5-star share > 0.75, or 1-star share > 0.40, or all ratings identical.
H2	Short filler	5-star reviews shorter than 12 characters exceed 30% of the sample.
H3	Near-duplicates	$J > 0.55$ for at least 3 review pairs (first 20 reviews).
H4	Repeated phrases	A three-word sequence appears in at least 4 of the first 20 reviews.
H5	Author diversity	$D_a < 0.7$ when $n > 8$.
H6	Sample size	$n < 6$.
H7	Date clustering	$D_t < 0.25$ with at least 5 dated reviews.

After the flags and statistics, as well as the sentiment context and up to 25 reviews (400 characters each) are processed, the information is sent to the 70B model for forensics. In return, we get the trust score, credibility score, likelihood of being a fake review, level of bias and its direction, linguistic analysis and review behavior analysis, contradiction between the reviewers, credibility of each aspect, verified genuine positives and concerns with review IDs, and a corrected true rating. Patterns identified heuristically and by the model are combined into one list without duplication.

E. Recommendation Module

The next step gets the context (sentiment, themes, trust, verified positives and concerns, and suspicious patterns) and produces one of four labels (highly recommended, recommended, visit with caution, not recommended), a confidence score, a visit score, pros and cons weighted appropriately, visitor types suitable for the place, reasons to stay away from it, practical advice, when to visit, and data reliability label. If the model does not produce a score breakdown, it is calculated using Eq. (4).

F. Presentation and Export

Finally, Streamlit [19] interface presents a verdict banner, six metric cards (visit score, AI confidence score, trust score, risk of fake reviews, trend, and true rating) and five tabs: Executive Verdict, Deep Sentiment, Guardrail and Authenticity, Visitor Guide, and Raw Data for Export. All charts are generated using Plotly [20]. The same data is outputted as console output and saved in JSON format for future use. One-second pause is inserted between LLM calls to comply with free-tier rate limits.

VI. EXPERIMENTAL SETUP AND IMPLEMENTATION

All the experiments are conducted in Python. The configuration is listed in Table III. As all used services have free tiers, the evaluation does not require any payment. API keys are loaded from environment files and not saved in the report.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

TABLE III: Implementation Configuration

Component	Setting	Value
Scraper	Actor	Apify Google Maps scraper [17]
Scraper	Reviews	10–100 (default 30), newest first, English
Geocoding	Service	Photon / OpenStreetMap [16]
LLM	Model	llama-3.3-70b-versatile via Groq [18]
LLM	Decoding	Temperature 0.2; 2,048 max tokens; JSON only
LLM input	Sentiment	Up to 30 reviews
LLM input	Guardrail	Up to 25 reviews, 400 characters each
Interface	Frontend	Streamlit [19], Plotly [20]
Output	Artifacts	Dashboard, console report, JSON file

Two evaluations were performed. First, there was a live case study in which the pipeline processed one restaurant (Sweet Home Fun N Food, Ahilyanagar, Maharashtra), and the resulting JSON report was examined, including the consistency test of the LLM's aggregates versus its own per-review labels. Second, a test of the deterministic heuristic layer was performed. We produced synthetic datasets containing 20-40 reviews with star ratings according to the J-shaped distribution (40%, 30%, 13%, 8%, and 9% for 5 to 1 stars respectively), text according to an approximate 100-word review vocabulary, unique authors, and random dates. From these datasets, eight attack scenarios were produced, including rating boost (90% of reviews have 5 stars), review bombing (60% have 1 star), equal ratings, short filler 5-star reviews, copy-paste text (60% of reviews are generated from one noisy template), sock-puppet authors (the authors pool contains approximately a quarter of n), temporal clustering (there are two posting dates), and tiny samples (2-5 reviews). Each scenario and a natural control were tested in 500 runs each using the heuristic function provided by the project, extracted without modifications from its source code. Sweep of the intensity of the attacks used 400 runs per point.

VII. RESULTS AND DISCUSSION

A. Live Case Study

The scraper kept 11 reviews with text among 7,383 Google reviews for this place, so that the sample includes 0.15% of the reviews. Table IV summarizes the result. The sample average rating (3.82) is smaller than Google aggregate (4.1), and the adjusted rating of the guardrail (3.5) is even smaller. No flag was raised by the heuristic, all 11 authors were unique, and no near-duplicates were detected. The LLM considered the sample as moderately credible (trust 0.70, probability of fake reviews 0.20) with low bias and positive direction.

TABLE IV: Case Study Output for One Restaurant

Quantity	Value
Google aggregate rating / total reviews	4.1 / 7,383
Reviews analysed; mean star rating	11; 3.82
5-star / 1-star share	36% / 9%
Unique authors; duplicate pairs	11 of 11; 0
Heuristic flags raised	0
Overall sentiment; LLM score	Positive; 0.65
Mean of per-review LLM scores	0.455
Temporal trend (older → recent)	Declining (0.70 → 0.20)

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

Quantity	Value
Trust / credibility / fake probability	0.70 / 0.80 / 0.20
Adjusted true rating	3.5 / 5
Verdict; confidence	Recommended; 0.80
Visit score; composite (as returned)	0.75; 0.68

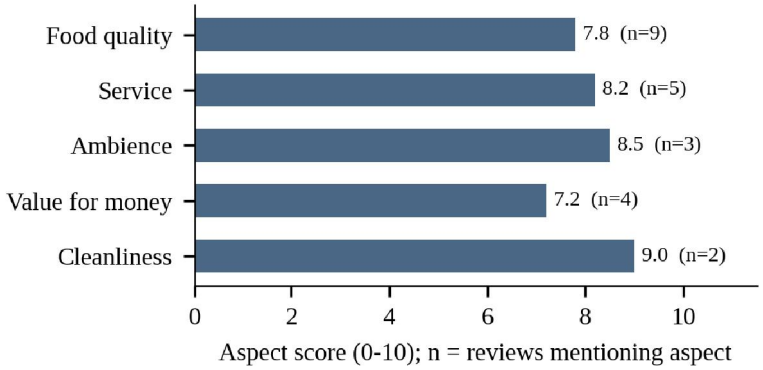

Fig. 2. Aspect scores returned by the sentiment stage. Accessibility and crowd or waiting time were not mentioned and are omitted.

As shown in Fig. 2, cleanliness and ambience received the highest scores (9.0 and 8.5 respectively), but these are based on just two or three reviews, while value for money received the lowest score (7.2). Food quality, being the aspect most often referred to (nine out of eleven reviews), got a score of 7.8. The distribution in the sample is mostly four- and five-star reviews (Fig. 3a), while the declining trend label can be based only on three recent and eight past reviews.

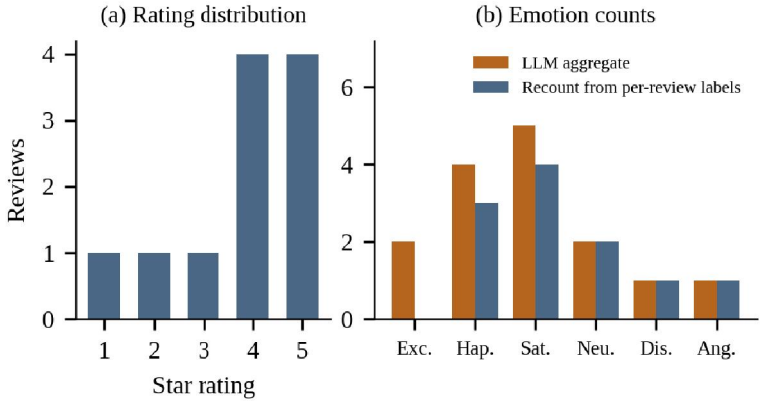

Fig. 3. (a) Star ratings of the 11 analysed reviews. (b) Emotion counts as reported by the LLM versus a recount from its own per-review labels.

Internal inconsistencies were also found in the LLM outputs (Fig. 3b). The total count of emotions is 15 for 11 reviews while counting the labels separately results in 11. The total sentiment value is 0.65, which differs from the average value of the sentiments produced by the model (0.455). Lastly, the prompt asks to give a visit score from 0 to 10, while the model gives values of 0.75 and 0.68, which are scaled between 0 and 1. These mistakes do not affect the qualitative decision, however, the inconsistency shows that aggregate numbers provided by LLMs must be recomputed in code, just as it must happen with the stars count.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

B. Heuristic Guardrail Evaluation

Table V shows the results of the controlled experiment. For all the natural control sets, the heuristic guardrail did not produce any alerts in any of the 500 trials. For each manipulation test case, the appropriate alert was triggered in at least 99.2% of the trials at the mentioned intensity level. Since all the manipulation scenarios were designed to cross the threshold of the rule, it means that the behavior of the rules was consistent.

TABLE V: Heuristic Layer on Synthetic Review Sets (500 Trials Each)

Scenario	Target flag (%)	Any flag (%)	Mean flags
Natural (control)	–	0.0	0.00
Rating boosting	100.0	100.0	1.17
Review bombing	99.2	99.2	0.99
Uniform ratings	100.0	100.0	2.00
Short filler 5-star	100.0	100.0	2.52
Copy-paste text	100.0	100.0	2.00
Sock-puppet authors	100.0	100.0	1.00
Temporal clustering	100.0	100.0	1.00
Tiny sample (n < 6)	100.0	100.0	1.27

The sweeps shown in Fig. 4 give us more insight into sensitivity. Rating boosting detection increases from 9% at the 64% share of five-star reviews to 53% at 76% and goes above 97% at 88%, which matches the 0.75 threshold stated in H1. Review bombing is detected at 37% share of one-stars, where its detection increases to 94% at 55%, close to the 0.40 threshold. Copy-paste detection begins at 58% at only 10% share of templated reviews and reaches 100% at 20%. In the natural data, the false positive rate was 0.5% at n = 8, 0.25% at n = 12, and 0% for n ≥ 20 (in 400 tries each time). The false positive rate is optimistic in my opinion since the synthetic vocabulary did not replicate stock phrases like “good food.”

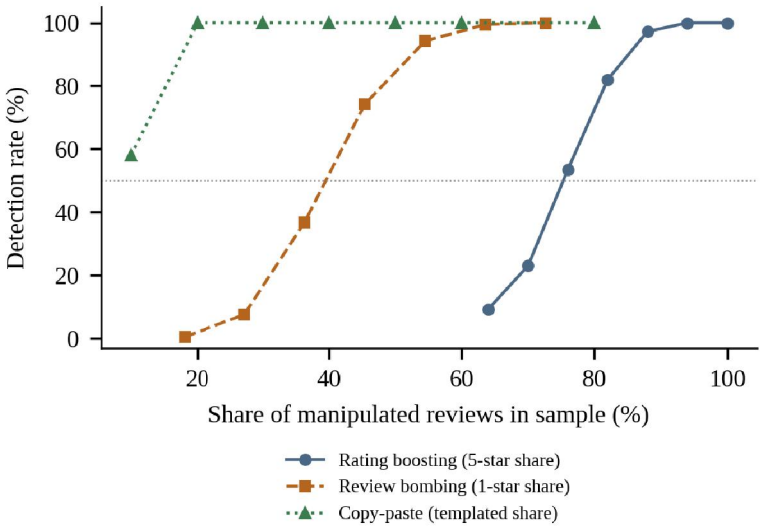

Fig. 4. Detection rate of three heuristic checks as the manipulated share of reviews increases (400 trials per point). For boosting and bombing, shares are the expected 5-star and 1-star shares of the generated sample.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

C. Discussion

This finding justifies two design decisions. Firstly, deterministic checks are cost-effective, repeatable, and interpretable, since every flagged violation includes the name of the rule and the numerical value. Secondly, LLM aggregates must be validated. This is the only way to identify the inconsistency in Section VII-A: per-review labels were available next to the aggregate. Thus, both pieces of information should be published; the dashboard provides raw reviews and JSON reports for this reason.

The findings do not prove the overall correctness of the processing pipeline. There is one live use-case, no manually-labeled ground-truth for sentiments and authenticity, no baseline against a rating-based or single-prompt version, and no measurement of end-to-end latency and cost. The values of trust score and fake-review probability are model-based estimates rather than proven facts.

VIII. LIMITATIONS AND FUTURE SCOPE

There are some limitations of the framework. The sample size is small and biased toward the most recent reviews. The case study used 11 reviews from the place with thousands of reviews. The threshold for heuristic flags is set manually and has not been calibrated on labelled data. The input to the LLM-based model is scraped and untrusted, so it might try the prompt injection. Currently, there is no filtering against this possibility. English-only reviews are requested and thus Hindi, Marathi and code-mixed reviews are not included. The LLM-based assessment of trust cannot be verified with external evidence; furthermore, its numeric results are potentially inconsistent. The free-tier rate limits and model deprecations may decrease the availability of the system. Finally, reviews are collected by third party scraper and thus raise the questions about terms of service and privacy (reviewer names are in the JSON report).

Future Work:

Further work involves: Recomputing the aggregation (count of emotions, sentiment, scores) deterministically from per-review labelling and validating against the results on the schema with known scale; calibration of thresholds on labelled spam corpora; embedding-based duplicate detection and reviewer history features; supporting multiple languages; filtering out or isoling review text to prevent prompt injection; comparison with human labellers and other baseline methods; and measurements of latency and cost on many machines.

IX. CONCLUSION

In this paper, we introduced the Location Review AI Analyzer, a toolchain which extracts reviews from Google Maps, computes aspect-based sentiment using the large language model, verifies the validity of reviews using both rules-based and large language models and produces a final recommendation visible in the interactive dashboard. A controlled experiment showed that the heuristic guardrail produced no alarms on natural synthetic data and detected injected manipulation when its share exceeded the rule thresholds. A case study showcased how the end-to-end output looks like and demonstrated inconsistencies in large language model aggregates, which suggests the need for deterministic recomputation of aggregates. This work is preliminary: it establishes that the pipeline does what it supposed to but not that it outperforms the alternatives. Evaluation using labeled data on larger data sets is the next step.

ACKNOWLEDGMENT

The authors acknowledge the guidance and assistance of the faculty and project guide.

REFERENCES

[1] B. Pang and L. Lee, “Opinion mining and sentiment analysis,” *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.

[2] M. Hu and B. Liu, “Mining and summarizing customer reviews,” in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

[3] M. Luca and G. Zervas, "Fake it till you make it: Reputation, competition, and Yelp review fraud," *Manage. Sci.*, vol. 62, no. 12, pp. 3412–3427, 2016.
[4] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proc. Int. Conf. Web Search Data Mining (WSDM)*, 2008, pp. 219–230.
[5] T. B. Brown et al., "Language models are few-shot learners," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 1877–1901.
[6] A. Dubey et al., "The Llama 3 herd of models," arXiv:2407.21783, 2024.
[7] M. Pontiki et al., "SemEval-2014 Task 4: Aspect based sentiment analysis," in *Proc. 8th Int. Workshop Semantic Evaluation (SemEval)*, 2014, pp. 27–35.
[8] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, "A survey on aspect-based sentiment analysis: Tasks, methods, and challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 11, pp. 11019–11038, 2023.
[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.
[10] W. Zhang, Y. Deng, B. Liu, S. J. Pan, and L. Bing, "Sentiment analysis in the era of large language models: A reality check," arXiv:2305.15005, 2023.
[11] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2011, pp. 309–319.
[12] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "What Yelp fake review filter might be doing?," in *Proc. 7th Int. AAAI Conf. Weblogs Social Media (ICWSM)*, 2013, pp. 409–418.
[13] J. Salminen, C. Kandpal, A. M. Kamel, S. Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," *J. Retailing Consum. Services*, vol. 64, Art. no. 102771, 2022.
[14] L. Wang et al., "A survey on large language model based autonomous agents," *Front. Comput. Sci.*, vol. 18, Art. no. 186345, 2024.
[15] J. Wei et al., "Chain-of-thought prompting elicits reasoning in large language models," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, 2022, pp. 24824–24837.
[16] Komoot, "Photon: An open-source geocoder for OpenStreetMap data." [Online]. Available: https://photon.komoot.io
[17] Apify, "Google Maps Scraper (compass/crawler-google-places)." [Online]. Available: https://apify.com/compass/crawler-google-places
[18] Groq, "GroqCloud developer documentation." [Online]. Available: https://console.groq.com/docs
[19] Streamlit, "Streamlit documentation." [Online]. Available: https://docs.streamlit.io
[20] Plotly, "Plotly Python graphing library." [Online]. Available: https://plotly.com/python/