Autonomous Fact-Checking: Integrating Local Inference and Adversarial NLP Attacks

Rômulo Henrique Nascimento Duarte[1], **Humberto Nunes de Lira**[1], **Vivianny Khatly Medeiros Pereira**[1], **Isaac Sebastian Lima de Araújo**[1], **Allan Gabriel da Cunha Vasconcelos**[1], **Yuri de Almeida Malheiros Barbosa**[1]

[1]Universidade Federal da Paraíba (UFPB)
João Pessoa – PB – Brazil

romulohenri000@gmail.com, nuneshumberto120@gmail.com, vkhatly@gmail.com
sisaac933@gmail.com, allan.vasconcelos@academico.ufpb.br, yuri@ci.ufpb.br

Abstract. *Digital misinformation threatens democratic discourse, especially in low-resource languages like Brazilian Portuguese. We present a fact-checking pipeline comparing structured claim decomposition (Claimify) against full-text extraction across four retrieval-augmented classifiers:* `gemini-2.5-flash-lite` *and three local open-weight models (Llama-3.1-8B, Qwen2.5-7B, Gemma-2-9B). Robustness is assessed under character-level and synonym perturbations. On a pilot sample, the API-based model shows a preliminary macro-F1 advantage for decomposition, which the three local models fail to replicate consistently. We also observe a systematic bias toward the false label across models.*

1. Introduction

Digital misinformation threatens democratic institutions and public discourse at an unprecedented scale, a challenge compounded in low-resource linguistic contexts such as Brazilian Portuguese by a scarcity of specialized tools and annotated resources [Guo et al. 2022, Silva et al. 2025]. Automated fact-checking has emerged as a promising direction to scale verification efforts beyond human-centric operations, typically framing the task as a classification problem in which a model assigns a veracity label to a claim, often supported by retrieved evidence [Thorne et al. 2018, Guo et al. 2022]; while English benefits from mature benchmarks [Wang 2017, Schlichtkrull et al. 2023], Brazilian Portuguese resources remain largely limited to binary, statement-only datasets [Silva et al. 2020, Garcia et al. 2022], with claim-level annotation efforts such as ClaimPT [Campos et al. 2026] only beginning to emerge for Portuguese. Beyond this data scarcity, two technical challenges limit existing systems' reliability: the quality of claim extraction directly affects downstream verification, since noisy or under-specified claims mislead classifiers regardless of their capacity [Metropolitansky et al. 2025, Guo et al. 2022]; and although adversarial robustness has received growing attention in NLP broadly [Morris et al. 2020, Jin et al. 2020], fact-checking systems are still seldom evaluated under adversarial conditions in low-resource languages, leaving their robustness to deliberate textual perturbations largely unknown in these settings. This paper investigates the following research question: **does structured claim decomposition improve the robustness and accuracy of LLM-based fact verification for Brazilian Portuguese, under both clean and adversarially perturbed inputs?** Our hypothesis is that decomposing documents into atomic, self-contained claims produces higher-quality classifier inputs than unstructured extraction. To investigate this, we build a deduplicated corpus from the *Aos Fatos* fact-checking portal, construct a balanced binary benchmark whose

false class is generated through NLI-validated logical negation, and evaluate a retrieval-augmented classification pipeline under controlled surface-level perturbations. Our objectives are: (i) to compare structured claim decomposition (*Claimify*) against a full-text extraction baseline on an identical set of documents; (ii) to quantify the robustness of both configurations under character-level and synonym-substitution attacks at two intensity levels; and (iii) to characterize systematic error patterns of the resulting classifiers.

The main contributions of this work are: (i) a **controlled comparison of structured claim decomposition against full-text extraction**, evaluated on an identical set of documents under a shared retrieval-augmented setup, isolating the extraction strategy as the sole varying factor; (ii) a **cross-model evaluation spanning an API-based model and three local open-weight models**, showing that the preliminary decomposition advantage observed for the API-based model does not replicate consistently across smaller, locally deployed models, an open question we discuss in Section 4; and (iii) an **adversarial robustness protocol for Brazilian Portuguese fact-checking**, combining controlled surface-level perturbations with NLI-validated semantic preservation gates and a claim-level logical negation procedure used to construct a balanced verification benchmark.

2. Related Work

Automated fact-checking has been widely framed as a natural language inference or classification task, in which a model assigns a veracity label to a given claim [Thorne et al. 2018, Guo et al. 2022]. Early systems relied on structured knowledge bases; more recent approaches leverage pre-trained and, increasingly, large language models prompted in-context. LIAR [Wang 2017] established a six-class benchmark for statement-level verification, and AVERITEC [Schlichtkrull et al. 2023] extended the paradigm to evidence-grounded, multi-hop verification. For Brazilian Portuguese, existing work has largely been confined to binary classification on datasets derived from news portals and social media [Silva et al. 2020, Garcia et al. 2022]. A recent step towards richer Portuguese resources is ClaimPT [Campos et al. 2026], a dataset of European Portuguese news articles annotated for factual claims at the span level (1,308 articles and 6,875 annotations); it targets claim *detection*, whereas our work focuses on downstream *verification* and its robustness, and addresses the Brazilian rather than European variant, narrowing the gap between evidence-grounded verification benchmarks in English and what is available for low-resource settings [Zuccon et al. 2022]. Poorly specified claims propagate noise into downstream classifiers regardless of their capacity [Guo et al. 2022]; we integrate *Claimify* [Metropolitansky et al. 2025], an LLM-based method that decomposes documents into atomic, self-contained claims while explicitly handling referential and structural ambiguity, and compare it against a baseline extracting claims from the full document text without decomposition or disambiguation. Retrieval-Augmented Generation (RAG) addresses a fundamental limitation of purely parametric models: their inability to access up-to-date or domain-specific evidence [Lewis et al. 2020]. Early RAG-based fact-checking systems retrieved evidence from structured knowledge bases or Wikipedia [Thorne et al. 2018], while more recent approaches index domain-specific corpora [Pan et al. 2023]. All four classification arms in our pipeline are retrieval-augmented over the same locally hosted index, ensuring that differences reflect the claim extraction strategy rather than asymmetric access to evidence; our results, however, indicate that re-

trieval access alone does not make the extraction strategies behave uniformly across model families (Section 4). Foundational work exposed the fragility of NLP models to character-level perturbations and synonym substitutions [Morris et al. 2020, Jin et al. 2020], and a growing body of research now examines attacks tailored to automated fact-checking specifically [Liu et al. 2025]; while this line of work evaluates robustness mainly in English, we apply LLM-generated surface-level perturbations, at two controlled intensity levels and validated by NLI-based semantic gates, to Portuguese claims.

3. Methodology

This section describes the four main components of the proposed fact-checking pipeline: dataset construction, claim extraction, veracity classification, and adversarial robustness evaluation.

3.1. Dataset Construction

The primary corpus, hereafter DATASET-AOSFATOS, was built through a multi-stage web scraping pipeline targeting the *Aos Fatos* fact-checking agency, covering 20 thematic pages of the "public policies" (*"políticas públicas"*) channel, using rotating headers and a mandatory delay between requests to respect the portal's infrastructure. For each article, we prioritize extraction of embedded `ClaimReview` JSON-LD metadata over plain HTML parsing and isolate the main content container to extract the full body text, discarding articles lacking verifiable or programmatically mappable content. An integrity audit revealed 53 duplicate records (repeated `id_documento` with identical content) among 1,047 collected rows; since duplicates would be twice as likely to be selected under stratified sampling, biasing every downstream stage, all were removed prior to sampling, yielding a final pool of **994 unique documents** (crawler implementation details are in our repository). The final corpus was serialized into a `.jsonl` format with document IDs, source URLs, titles, and full report text, the input to all claim extraction configurations in Section 3.2.

Verification benchmark and label construction. Unlike our earlier formulation, veracity labels are not taken from the portal's `reviewRating` verdicts, which apply to the article as a whole, not individual claims. Instead, we construct a balanced binary benchmark at the claim level: canonical claims constitute the *true* class, and their *false* counterparts are generated through controlled logical negation, validated by bidirectional NLI contradiction gates (Section 3.4), yielding one *false* instance per *true* claim and a 50/50 balance verified programmatically in every cell.

Sampling. From the pool of 994 unique documents, we sample **50 documents** using a fixed seed (42) with stratification by document length; documents used as few-shot demonstrations are kept outside the analysis set.

3.2. Claim Extraction

Raw fact-checking reports contain multiple sentences, background context, and narrative elements. Transforming this text into well-formed, verifiable claims is the first stage of

the pipeline, and the central experimental variable of this work. In both configurations described below, claims are extracted from the **full body text** of each report; headlines are never used as claim proxies. This holds for every stage of the pipeline, including classification (Section 3.3).

We evaluate two extraction conditions. **Baseline extraction**: an LLM is prompted to extract the factual claims present in the full document text in a single pass, without any decomposition or disambiguation stages, a straightforward, low-engineering approach. **Claimify**: we apply *Claimify* [Metropolitansky et al. 2025], an LLM-based method that decomposes the document into atomic, self-contained claims through dedicated selection, disambiguation, and decomposition stages, explicitly handling referential and structural ambiguity; following the method's staged design, we execute these stages in batched form per document (three LLM calls per document per condition), a documented adaptation of the original per-sentence formulation.

Both conditions use the same underlying LLM (Section 3.3) and receive identical inputs, so observed differences are attributable to the extraction strategy alone. The two strategies produce structurally different outputs: the baseline extracts on average 13.2 claims per document (660 total), while Claimify's decomposition yields 52.4 (2,621 total), reflecting its finer-grained segmentation. Extracted claims, hereafter *canonical claims*, constitute the *true* class and the source material for the perturbation and negation procedures (Section 3.4); document-level comparability between arms is enforced by the intersection protocol defined there.

3.3. Veracity Classification

We evaluate veracity classification through In-Context Learning (ICL) strategies over a retrieval-augmented pipeline. Every classification arm, regardless of extraction strategy, receives retrieval-augmented context from the same locally hosted index, ensuring symmetric access to evidence across all experimental conditions.

3.3.1. Classification Model

All pipeline stages requiring an LLM (claim extraction, perturbation, negation, and classification) use `gemini-2.5-flash-lite` at temperature 0, accessed through its floating alias since no dated snapshot was available at experiment time; we document this as an explicit reproducibility deviation, anchoring reproducibility to each run's execution date rather than an immutable identifier (prompts, hyperparameters, and execution metadata are available in our repository). The classifier receives the claim, retrieved evidence passages (Section 3.3.3), and a directive specifying the two allowed labels and output format, producing a brief reasoning followed by a final verdict line from which the label is parsed; as established in Section 3.2, claims under classification originate from the full body text of the source reports in both arms.

3.3.2. Local Models

To characterize how extraction strategy interacts with smaller, open-weight models, we evaluate three locally deployed instruction-tuned models, **Llama-3.1-8B-Instruct**, **Qwen2.5-7B-Instruct**, and **Gemma-2-9B-it**, served via the Hugging Face `transformers` pipeline with 4-bit quantization (BitsAndBytes) on a single GPU, ge-

nerating up to 250 new tokens per response at temperature 0. Local models reuse the same retrieval context (Section 3.3.3), prompt contracts, and parsing policy (Section 3.3.4) as the API-based model, with each surviving claim yielding one classification call per surface condition plus one for its negated counterpart (Section 3.4.4). Due to the cost of local generation, they are evaluated on half the pilot's surviving claims (44 of 89 baseline; 227 of 455 Claimify, same seed), yielding 3,252 raw calls per model; since this subsample is smaller than, and independently drawn from, the full pilot used for the API-based model, we report local results separately rather than in a direct comparison table (Section 7).

3.3.3. Retrieval-Augmented Generation

Fact-checking reports from DATASET-AOSFATOS are segmented into overlapping passages using a sliding window of 3 sentences with step 1 (via NLTK's Portuguese Punkt tokenizer) and indexed in a locally hosted, persistent ChromaDB instance; the index thus contains the same *Aos Fatos* articles from which the canonical claims are extracted, a design choice we control for explicitly. For each claim, passages originating from its own source document are excluded from retrieval at query time, preventing the classifier from trivially retrieving the report that generated the claim, an exclusion verified by automated audit for every run. Passages and queries are encoded with `intfloat/multilingual-e5-large`, pinned to a specific revision, with normalized embeddings; the retriever fetches a buffered candidate set, discards source-document passages, deduplicates near-identical windows from the same document (windows sharing two or more sentence indices), and returns the top-5 remaining passages, injected into the prompt as supporting context.

3.3.4. Prompting and Output Parsing

All configurations are evaluated under two prompting regimes: **zero-shot**, in which the model receives the claim, retrieved context, and a task description with no labeled examples; and **few-shot**, in which four labeled demonstrations (claim, short reasoning, final label) are prepended, drawn from documents outside the analysis set to prevent contamination, a constraint verified programmatically.

The classifier outputs one of the two target labels; any unparsable response, including explicit declines to commit citing insufficient evidence, is counted as a classification *error* under a policy fixed before the experiments, reflecting the deployment view that a pipeline failing to produce a verdict has failed at its task. We report this failure rate (`parse_fail`) separately in all result tables.

3.4. Adversarial Robustness Evaluation

To assess the pipeline's sensitivity to deliberate textual perturbations, we apply surface-level adversarial attacks at two intensity levels and evaluate classification performance across all resulting conditions. Unlike our earlier formulation, logical negation is not treated as an attack: it is the mechanism that generates the *false* class of the benchmark, described below. All perturbations and negations are generated by the same LLM as the rest of the pipeline (Section 3.3.1), at temperature 0.

3.4.1. Surface-Level Attack Strategies

Perturbations are applied to the full document body before claim extraction, so that each extraction arm processes clean and attacked versions of the same documents. Two strategies are used, each at two intensity levels: **character-level perturbations (typos)**, realistic typographic errors (swaps, insertions, deletions) controlled by an edit budget of one per 16 content words (*light*) or one per 8 (*strong*); and **synonym substitution**, replacing content words with semantically similar alternatives at a rate of 15% (*light*) or 30% (*strong*). Together with the unperturbed *identity* condition, which serves as the control, this yields five surface conditions per document per arm.

3.4.2. False-Class Generation via Logical Negation

The *false* class is generated by logical negation applied to each canonical claim *individually*: each claim is rewritten so that its factual core is inverted while remaining fluent, and the resulting claim receives the label *false* by construction (labels under negation are *inverted*, never preserved). Because negation operates claim-by-claim, each original claim maps one-to-one to its negated counterpart, requiring no post-hoc matching. Every negated claim must pass a bidirectional contradiction gate, a multilingual NLI model (`MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7`) classifying the original claim as contradicting the negated version *and* vice versa; claims failing this gate are excluded, ensuring every *false* instance is a genuine semantic inversion rather than a superficial rewording. This per-claim procedure achieves gate approval rates of 90.0% (baseline) and 91.8% (Claimify), well above the 12–19% approval of an earlier document-level attempt, where the model frequently left individual assertions unchanged, motivating the per-claim formulation.

3.4.3. Alignment, Validation Gates, and Intersection Protocol

For the surface conditions, claims extracted from attacked documents must be matched to their clean counterparts before any comparison is meaningful. We align each attacked-document claim to canonical claims via embedding cosine similarity (same `multilingual-e5-large` model as the retrieval index), with a threshold of $\tau = 0.90$ fixed by inspection of the pilot score distribution (± 0.05 sensitivity showed near-constant survival rates, so the threshold is not critical); aligned pairs must also pass an NLI-based **semantic preservation gate**, verifying the perturbation did not alter the claim's meaning beyond the intended surface-level noise. The final evaluation set is the **intersection** of claims surviving alignment and gating across all conditions, guaranteeing every configuration (extraction arm $\times$ prompting regime $\times$ surface condition) is evaluated on the same content, eliminating survivorship differences as a confound. We denote by σ the fraction of sampled documents in this intersection ($\sigma = 0.58$, 29 of 50 in the pilot); within each arm, every cell is exactly balanced between *true* and *false* instances (50/50), verified programmatically.

3.4.4. Evaluation Protocol and Metrics

For each evaluation cell we report accuracy, **macro F1-score**, per-class F1 (*true*/*false*), and `parse_fail` (Section 3.3.4). Since every cell is balanced 50/50 by construction, macro and weighted F1 coincide, so we report macro F1 uniformly as the primary metric.

Robustness to a given attack is the degradation relative to identity, $\Delta\text{F1} = \text{F1}_{\text{id}} - \text{F1}_{\text{attacked}}$, where smaller ΔF1 indicates greater robustness.

4. Results and Discussion

All results below come from the pilot run: $D = 50$ documents sampled as described in Section 3.1. Given the sample size, we present these findings as preliminary evidence rather than definitive conclusions, and we dimension every claim accordingly.

4.1. Extraction and Survival Statistics

As established in Section 3.2, the two arms yield 660 and 2,621 canonical claims, respectively. After alignment, validation gates, and the intersection protocol (Section 3.4.3), 89 baseline and 455 Claimify claims survive, drawn from the 29 documents in the global intersection ($\sigma = 0.58$); Claimify's finer-grained decomposition thus yields both more claims per document and a higher survival volume, though survival *rates* per condition vary by attack type, as shown next.

4.2. Alignment and Validation Gates

Table 1. Alignment and gate-pass rates, baseline vs. Claimify (pilot, $D = 50$; $n = 660$/2,621 claims per condition). Aligned/gate pass/cos as defined in Section 3.4.3.

	Baseline			Claimify		
Condition	Aligned	Gate pass	Cos	Aligned	Gate pass	Cos
negation	1.000	0.900	0.987	1.000	0.918	0.978
syn strong	0.874	0.361	0.983	0.890	0.481	0.980
syn light	0.898	0.388	0.983	0.822	0.436	0.975
typo strong	0.855	0.533	0.990	0.863	0.590	0.987
typo light	0.830	0.477	0.989	0.886	0.589	0.990

Table 1 reports, per arm and condition, the fraction of claims aligned at $\tau = 0.90$ (Aligned), passing the semantic gates (Gate pass), and mean cosine similarity of aligned pairs (Cos). Negation passes at high rates in both arms (0.900 and 0.918), confirming the reliability of the per-claim procedure (Section 3.4.2). Surface conditions pass at substantially lower rates (0.36–0.59), as expected: the gates are deliberately conservative, discarding perturbed claims whose meaning drifted beyond surface noise. Typo conditions survive at higher rates than synonym conditions in both arms, consistent with synonym substitution being the more invasive perturbation.

4.3. Classification Results

Table 3 reports classification results per arm and prompting regime, averaged over the five surface conditions; the full per-cell results for all twenty evaluation cells are available in our repository. Three patterns emerge. **Claimify shows a preliminary advantage.** Averaged over surface conditions, the Claimify arm outperforms the baseline by 2.5 macro-F1 points under few-shot (0.667 vs. 0.642) and 3.0 points under zero-shot (0.624 vs. 0.594). Given the modest number of surviving baseline claims (89 claims, 178 instances per cell),

we interpret this as preliminary evidence favoring structured decomposition, in line with the open debate on whether claim decomposition helps or hinders downstream verification [Hu et al. 2025], not as a conclusive result.

Few-shot consistently outperforms zero-shot in both arms and in every surface condition, with gains of 4.3–4.8 macro-F1 points. This contrasts with our earlier formulation, in which few-shot prompting sometimes degraded performance on noisy inputs, and suggests that with retrieval-augmented context and controlled claim quality, in-context demonstrations provide a stable benefit.

Robustness to surface attacks is high in both arms. Across all twenty cells, degradation relative to identity is small throughout ($|\Delta\text{F1}| \leq 0.028$), indicating the surviving, gate-validated perturbations rarely flip classifier decisions. We emphasize the selection effect: these are precisely the perturbations that *preserved* claim semantics, so this measures robustness to meaning-preserving noise, not arbitrary corruption.

Table 2. Per-cell classification results (pilot, $D = 50$), by arm, surface condition, and prompting regime (fs/zs). id = identity (unperturbed control). Each cell is balanced 50/50, with $n = 178$ instances per cell in the baseline arm and $n = 910$ in the Claimify arm. F1t/F1f = F1 for *true*/*false*; pf = parse_fail.

		Macro F1		F1t		F1f		pf	
Arm	**Surface**	fs	zs	fs	zs	fs	zs	fs	zs
baseline	id	0.651	0.596	0.582	0.484	0.721	0.708	0.039	0.011
baseline	syn strong	0.641	0.596	0.571	0.484	0.711	0.708	0.028	0.011
baseline	syn light	0.668	0.602	0.611	0.496	0.725	0.708	0.028	0.006
baseline	typo strong	0.615	0.587	0.529	0.472	0.701	0.702	0.034	0.006
baseline	typo light	0.636	0.587	0.561	0.472	0.711	0.702	0.034	0.006
claimify	id	0.667	0.625	0.605	0.545	0.730	0.706	0.027	0.042
claimify	syn strong	0.662	0.624	0.598	0.540	0.726	0.708	0.026	0.047
claimify	syn light	0.671	0.626	0.613	0.545	0.729	0.707	0.022	0.043
claimify	typo strong	0.669	0.619	0.607	0.536	0.731	0.702	0.029	0.040
claimify	typo light	0.667	0.625	0.605	0.545	0.729	0.706	0.026	0.042

Table 3. Average classification results per arm and prompting regime (pilot, $D = 50$), averaged over the five surface conditions.

Arm	Prompt	Acc	Macro F1	F1_{true}	F1_{false}	parse_fail
baseline	fs	0.646	0.642	0.571	0.714	0.033
baseline	zs	0.623	0.594	0.482	0.706	0.008
claimify	fs	0.669	0.667	0.606	0.729	0.026
claimify	zs	0.628	0.624	0.542	0.706	0.043

4.4. Systematic Error Patterns

Two error patterns hold across both arms and all conditions.

Systematic bias toward *false*. F1 on the *false* class (0.70–0.73) is consistently higher than on *true* (0.47–0.61) in every cell; since cells are balanced by construction, this reflects classifier behavior, not class imbalance. A plausible, untested factor is the retrieved context itself: fact-checking reports frequently contain explicit debunking language, which may prime the classifier toward *false* verdicts regardless of the claim.

Non-committal responses concentrate in the Claimify arm. The `parse_fail` rate is consistently higher for Claimify under zero-shot (4.0–4.7%) than baseline (0.6–1.1%); inspection of raw responses confirms these are explicit declines to verdict, not parsing artifacts, more frequent for the atomic, decontextualized claims Claimify produces, whose evidentiary requirements are harder to satisfy within the top-5 passages. Under our pre-registered policy (Section 3.3.4) these count as errors, so the Claimify advantage above is achieved *despite* this penalty, a tension with decomposition granularity aligning with [Hu et al. 2025] and, in our view, the most consequential open question for pipelines of this kind.

4.5. Local Models: Cross-Model Comparison

Table 4. Local models: results by arm, prompting regime, and metric, averaged over five surface conditions (44 baseline / 227 Claimify claims; 88/454 instances per cell). F1t/F1f = F1 for *true*/ *false*; pf = parse_fail.

		Macro F1		F1t		F1f		pf	
Model	**Arm**	fs	zs	fs	zs	fs	zs	fs	zs
Gemma-2-9B	baseline	0.636	0.655	0.545	0.553	0.727	0.757	0.186	0.123
Gemma-2-9B	Claimify	0.560	0.587	0.546	0.555	0.573	0.620	0.340	0.277
Llama-3.1-8B	baseline	0.422	0.531	0.296	0.400	0.549	0.662	0.434	0.177
Llama-3.1-8B	Claimify	0.430	0.539	0.321	0.430	0.538	0.648	0.467	0.152
Qwen2.5-7B	baseline	0.586	0.504	0.489	0.314	0.684	0.694	0.080	0.005
Qwen2.5-7B	Claimify	0.546	0.516	0.440	0.338	0.651	0.694	0.075	0.007

Table 4 reports classification results for the three local models, averaged over the five surface conditions, on the reduced subsample described in Section 3.3.2 (44 baseline / 227 Claimify claims; 88/454 balanced instances per cell). Three patterns emerge, two extending the API-based model's findings and one specific to local deployment.

No consistent effect of structured decomposition emerges. Gemma-2-9B favors the baseline in both regimes (-7.6 and -6.8 points), Llama-3.1-8B shows negligible differences ($\leq$0.8 points), and Qwen2.5-7B favors baseline under few-shot but reverses under zero-shot, unlike the API-based model's consistent advantage (Section 4.3).

Non-committal responses vary far more across local models than for the API-based model. `parse_fail` ranges from under 1% (Qwen2.5-7B, zero-shot) to 46.7% (Llama-3.1-8B, Claimify, few-shot), an order of magnitude wider than `gemini-2.5-flash-lite`'s 2.7–5.7%. Llama-3.1-8B and Gemma-2-9B are especially affected under few-shot (43–47% and 19–34%), while Qwen2.5-7B stays close to the API-based model's range. Since local generation is capped at 250 tokens (Section 3.3.2), we cannot fully disentangle genuine uncertainty from response truncation, a confound discussed further in Section 7.

The bias toward *false* is present in all three local models, more pronounced in two. Llama-3.1-8B and Qwen2.5-7B show large true/false F1 gaps under zero-shot (Llama: 0.40 vs. 0.66; Qwen: 0.31–0.34 vs. 0.69), wider than for the API-based model, while Gemma-2-9B shows a smaller gap under baseline but comparable under Claimify. This asymmetry, robust across all four models, is discussed in Section 4.4.

5. Conclusion

This paper presented a fact-checking pipeline for Brazilian Portuguese that addresses two underexplored questions: whether structured claim decomposition improves classification quality, and how robust the resulting classifiers are, both to adversarial perturbation and across model families. By constructing a deduplicated corpus from *Aos Fatos*, building a balanced verification benchmark through NLI-validated logical negation, and evaluating four retrieval-augmented models under a shared protocol, we obtained a more nuanced picture than either extreme would suggest. For the API-based model, structured decomposition via *Claimify* shows a consistent, if preliminary, macro-F1 advantage over full-text extraction (Section 4.3), an advantage that does **not** replicate consistently across the three local, open-weight models evaluated under the same protocol (Section 4.5). Rather than supporting a uniform claim about decomposition, these results suggest that its benefit is conditional on model capacity or training, an open question we did not anticipate at the outset and that we believe merits targeted follow-up work. Two further patterns emerged across all four models: a systematic bias toward the *false* label, more pronounced in the local models, and non-committal responses (`parse_fail`) that varied far more widely among local models than for the API-based model, a spread plausibly confounded by our fixed local generation budget rather than genuine differences in model uncertainty. Taken together, this work argues against treating claim decomposition, or open-weight local deployment, as unconditionally beneficial for retrieval-augmented fact verification; their effects appear to interact with model-specific factors a single pilot cannot disentangle, an invitation for larger-scale, model-diverse evaluations rather than a negative result.

6. Future Work

The most immediate next step is a **controlled, equal-scale cross-model comparison** of all four models on the same survivor set and token budget, disentangling whether local models' failure to replicate the Claimify advantage is genuine or an artifact of the reduced subsample and generation budget used here (Section 7). We also plan to apply bootstrap confidence intervals to the pilot results, quantifying how much of the 2.5–3.0 point Claimify advantage is attributable to sampling variability; isolate the contribution of retrieval (RAG on/off, varying k) from that of decomposition; and investigate the systematic bias toward *false*, for instance by filtering verdict-bearing sentences from retrieved context. Further directions include expanding DATASET-AOSFATOS to other agencies, a human-reliability audit of the canonical claim set, and evaluating additional open-weight model families.

Reproducibility Statement

All data, code, and configuration needed to reproduce this work, corpus construction scripts, full prompt texts, the configuration file (seed, thresholds, retrieval parameters), and evaluation code, are publicly available in our repository, along with basic run instructions. As documented in Section 3.3.1, the classification LLM is the only component not pinnable to an immutable identifier, so its execution date is recorded instead. All artifacts are archived at `https://doi.org/10.17605/OSF.IO/YZ736`.

7. Limitations

Sample size and local model comparability. Results come from a pilot of 50 documents, with the baseline arm retaining only 89 surviving claims. The reported Claimify advantage for the API-based model (2.5–3.0 macro-F1 points) is within a range where sampling variability cannot be ruled out; we frame it as preliminary throughout. The three local models are evaluated on an independently sampled, smaller subset (half the pilot's survivors), so their results cannot be directly compared to `gemini-2.5-flash-lite`'s in a single table. Equal-scale, cross-model evaluation with confidence intervals is the immediate next step.

Confounded parse failure rates for local models. Local generation was capped at 250 tokens for GPU memory and runtime reasons. `parse_fail` rates range from under 1% to over 46%, far wider than the API-based model's range, and markedly higher under few-shot for two of the three models. We cannot distinguish genuine model uncertainty from response truncation before the verdict line, limiting how much of the local models' failure to replicate the Claimify effect owes to extraction strategy versus this generation budget. **Scope.** All four models share the same retrieval setup ($k = 5$) and embedding model; we did not run ablations isolating the contribution of retrieval or architecture. The API-based model is accessed through a floating alias, with reproducibility anchored by execution date rather than an immutable identifier. DATASET-AOSFATOS is scraped from a single agency and bounded to its public-policy channel, which risks institutional and topical bias; canonical claims are extracted from reports that may quote the very misinformation they debunk, a failure mode our gates do not screen for and that we did not audit with human review.

Ethical Considerations

The dataset was constructed by scraping publicly available content from the *Aos Fatos* fact-checking portal, with request delays to minimize server impact and no collection of personally identifiable information. Automated fact-checking systems, including the pipeline proposed here, should not be used as sole arbiters of truth: classification errors are inevitable, and human oversight remains essential for consequential decisions. The adversarial perturbation and negation procedures generate synthetic false claims strictly for controlled evaluation, disclosed with a defensive intent to expose pipeline vulnerabilities; we acknowledge such techniques could nonetheless be misused to evade fact-checking or fabricate misleading content at scale. Finally, the language models used may reflect biases in their pre-training data, and outputs should be interpreted with appropriate caution in politically sensitive contexts.

Use of Artificial Intelligence

This work used generative AI tools as writing and engineering assistants: **Claude Fable 5** and **Claude Sonnet 4.6** (Anthropic) supported drafting and revising manuscript sections and the implementation of the experimental pipeline, under the authors' direction. All design decisions, data, analyses, and conclusions remain the authors' sole responsibility; they reviewed and validated all AI-assisted content.

Referências

Campos, R., Sequeira, R., Nerea, S., Cantante, I., Folques, D., Cunha, L. F., Canavilhas, J., Branco, A., Jorge, A., Nunes, S., Guimarães, N., and Silvano, P. (2026). ClaimPT: A portuguese dataset of annotated claims in news articles. In *Advances in Information Retrieval (ECIR 2026)*, volume 16486 of *Lecture Notes in Computer Science*, pages 544–560. Springer.

Garcia, D., Silva, A., and Santana, C. (2022). Fake news detection in brazilian portuguese: Datasets and current challenges. In *Proceedings of the Brazilian Symposium on Information Systems*, pages 115–122.

Guo, Z., Schlichtkrull, M., and Vlachos, A. (2022). From ready-to-wear to ready-to-match: A survey on automated fact-checking. *ACM Computing Surveys*, 55(6):1–34.

Hu, Q., Long, Q., and Wang, W. (2025). Decomposition dilemmas: Does claim decomposition boost or burden fact-checking performance? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6313–6336.

Jin, D., Jin, Z., Zhou, J. T., and Szolovits, P. (2020). Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8018–8025.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Lewis, V., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, pages 9459–9474.

Liu, F., Abuadbba, A., Moore, K., Nepal, S., Paris, C., Wu, J., Yang, J., and Sheng, Q. Z. (2025). Adversarial attacks against automated fact-checking: A survey. *arXiv preprint arXiv:2509.08463*.

Metropolitansky, I., Goldstein, A., and Shahaf, D. (2025). Claimify: Decomposing documents into atomic verifiable claims using large language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 412–426.

Morris, J., Lifland, E., Yoo, J. Y., Grigsby, J., Jin, D., and Qi, Y. (2020). TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

Pan, J., Wang, X., and Gao, Y. (2023). A survey on retrieval-augmented text generation for large language models. *arXiv preprint arXiv:2304.09400*.

Schlichtkrull, M., Guo, Z., and Vlachos, A. (2023). AVeriTeC: A dataset for real-world fact checking with evidence. *arXiv preprint arXiv:2305.13117*.

Silva, J., Santos, R., Almeida, T., and Pardo, T. (2020). Fake.Br corpus: A corpus for fake news detection in brazilian portuguese. In *Proceedings of the International Conference on Computational Processing of Portuguese*, pages 416–425.

Silva, R. M., Amamou, H., Ferraz, L. B. S., da Silva, F. K. A., and Avila, A. R. (2025). Fake news detection in portuguese under large language model-generated content. *Journal of the Brazilian Computer Society*, 31(1):1149–1166.

Thorne, J., Vlachos, A., Christodoulopoulos, C., and Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the Human Language Technologies: NAACL*, pages 809–824.

Wang, W. Y. (2017). Liar, liar pants on fire: A new benchmark dataset for fake news detection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 422–426.

Zuccon, G., Koopman, B., and Palotti, J. (2022). The landscape of fact-checking resources for low-resource languages. *Journal of Artificial Intelligence Research*, 74:1435–1461.