"Do You Mean…?": Measuring and Mitigating Intent Drift in LLM-Based Query Correction

Anonymous Author(s)

Abstract

Query correction and rewriting systems are ubiquitous in search engines, chatbots, and virtual assistants. When these systems alter the user's original intent, downstream performance degrades and users lose trust. Yet no systematic evaluation exists for measuring how often large language models (LLMs) change user intent during query correction. We address this gap by evaluating two state-of-the-art LLMs (GPT-4.1 and CLAUDE SONNET 4.5) across three correction strategies of increasing aggressiveness on 400 queries from the BANKING77 and CLINC150 intent classification benchmarks. We find that conservative correction ("fix errors") preserves intent 98.5% of the time, while more aggressive strategies ("rewrite clearly," "improve") cause intent shifts in 9–15% of cases—a 6–10$\times$ increase in violations. Claude makes more aggressive edits than GPT-4.1, leading to higher violation rates under open-ended instructions. We validate these findings using an LLM-as-judge protocol and show that bidirectional natural language inference entailment is the strongest automated predictor of intent change ($r = -0.408$, $p < 0.0001$). Finally, we propose CONFCORRECT, a confidence-aware correction strategy that eliminates intent violations entirely while requesting clarification for only 9.3% of ambiguous queries. Our results provide actionable guidelines for building correction systems that know when to fix, when to ask, and when to leave well enough alone.

1 Introduction

When a user types "How long with my cash withdrawal stay pending for?" into a banking chatbot, a helpful system might correct the typo and return the right answer. But when the same system rewrites "Can you help with a transfer to an account" as "Can you assist me with transferring funds to another account?," it may silently shift the user's intent from *beneficiary_not_allowed* to *transfer_into_account*—returning an answer to a question the user never asked. This tension between helpfulness and intent fidelity is at the heart of every query correction system deployed today.

The problem is widespread. Query correction powers search engines, virtual assistants, chatbots, and autocomplete systems used by billions of people daily. Modern systems increasingly rely on large language models (LLMs) to rewrite user queries for clarity, correct spelling errors, or "improve" phrasing before passing the query to downstream components [Arora et al., 2024]. While these rewrites often help, they can also alter the user's original intent in subtle ways that are difficult to detect and costly to recover from. Despite extensive work on spelling correction [Jayanthi et al., 2020, Zhang et al., 2020a], clarification question generation [Hu et al., 2020, Wang et al., 2023], and text rewriting [Li et al., 2025], no systematic evaluation exists for how often LLMs change user intent during query correction.

Our contribution. We conduct the first systematic study of intent preservation in LLM-based query correction. We evaluate two state-of-the-art LLMs (GPT-4.1 and CLAUDE SONNET 4.5) across three prompting strategies of increasing aggressiveness on 400 queries drawn from established intent classification benchmarks (BANKING77 [Casanueva et al., 2020] and CLINC150 [Larson et al.,

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

2019]). Our main finding is striking: the choice of prompt instruction—not the choice of model—is the dominant factor in determining intent preservation. Conservative correction ("fix errors") preserves intent 98.5% of the time, while aggressive rewriting ("improve") causes intent shifts in up to 15% of cases.

We validate our automated metrics against an LLM-as-judge protocol, finding that bidirectional natural language inference (NLI) entailment is the strongest predictor of intent change ($r = -0.408$, $p < 0.0001$), outperforming semantic similarity and intent classifier agreement. We then propose CONFCORRECT, a confidence-aware correction strategy that uses classifier confidence to decide when to auto-correct, when to ask a clarifying question, and when to abstain. CONFCORRECT eliminates intent violations entirely while requesting clarification for only 9.3% of queries.

Figure 1 summarizes our experimental findings. Conservative correction is safe across both models, while aggressive strategies introduce intent drift that scales with edit aggressiveness.

Our contributions are:

- We quantify LLM intent violation rates across two models and three correction strategies, finding a 6–10$\times$ increase in violations when moving from conservative to aggressive prompts (section 4).
- We validate automated intent preservation metrics against LLM-as-judge labels, identifying bidirectional NLI entailment as the most reliable predictor (section 4.2).
- We propose CONFCORRECT, a confidence-aware strategy that achieves zero intent violations with only 9.3% clarification rate (section 4.3).
- We release our evaluation framework and annotated dataset to enable reproducible research on intent-preserving correction.[1]

2 Related Work

Our work sits at the intersection of query correction, clarification question generation, intent detection, and semantic preservation evaluation. We review each area and position our contribution relative to prior work.

Spelling and query correction. Neural spelling correction has advanced rapidly with toolkits like NeuSpell [Jayanthi et al., 2020], which provides context-sensitive correction using surrounding words for disambiguation. Zhang et al. [2020a] introduced the Twitter Typo Corpus capturing naturally occurring autocorrect errors, where systems frequently override intentional non-standard spellings. Production-scale systems face additional challenges with code-switching, transliteration, and domain jargon [Sharma et al., 2023]. These works optimize for *correction accuracy*—whether the typo was fixed—rather than *intent preservation*—whether the correction maintained what the user wanted. Our work reframes spelling correction as an intent-preservation problem, quantifying how often LLM-based correction alters the underlying query intent.

Clarification question generation. When a system is uncertain about user intent, asking a clarifying question can be more appropriate than guessing. Hu et al. [2020] deployed an RL-based clarification system in production, achieving 66.4% click-through rate on clarification suggestions using Monte Carlo Tree Search for question selection. Dhole [2020] found that purely generative approaches achieve only 34% coverage for intent-discriminating questions, motivating template-based fallbacks. Wang et al. [2023] showed that zero-shot clarifying question generation using constrained decoding outperforms supervised baselines on naturalness (82.6% rated "Good"), enabling clarification without task-specific training data. More recently, Kebir et al. [2026] proposed retrieval-augmented clarification using DPO to ground clarifying questions in retrieved evidence. Our confidence-aware strategy builds on this literature by establishing principled thresholds for when to correct versus when to clarify.

Intent detection and classification. Intent classification provides the evaluation backbone for our study. Arora et al. [2024] benchmarked LLMs against traditional intent detection methods, finding that LLMs outperform smaller models by approximately 8% while a hybrid routing approach achieves comparable accuracy at lower cost. Most relevant to our work, den Hengst et al. [2024] proposed Conformal Intent Classification and Clarification (CICC), which uses conformal prediction to produce prediction sets with statistical coverage guarantees. Their framework decides between

[1] Code and data available at the project repository.

answering directly (single-intent prediction set), asking a clarifying question (2–7 intents), or escalating (more than 7 intents). On BANKING77 with optimized confidence levels, CICC achieves 97% coverage with a 92% single-answer rate. Deng et al. [2025] quantified the cost of not clarifying: models achieve only 13.7% accuracy on ambiguous queries compared to 71.5% with contextual clarification. We adopt the CICC framework's confidence-threshold paradigm but apply it to the correction decision rather than the classification decision.

Semantic preservation metrics. Measuring whether a rewrite preserves the original meaning requires appropriate metrics. BERTScore [Zhang et al., 2020b] computes token-level similarity using contextual embeddings and achieves 0.785 Pearson correlation with human judgments on semantic preservation tasks. BARTScore [Yuan et al., 2021] evaluates text as a generation problem using BART's log-likelihood, independently assessing informativeness, fluency, and factuality. ParaScore [Shen et al., 2022] combines semantic similarity with lexical divergence, finding that reference-free metrics outperform reference-based ones for paraphrase evaluation. Li et al. [2025] introduced decoupled rewards for text rewriting—agreement, coherence, and conciseness—where the conciseness reward directly penalizes unnecessary edits via edit ratio. We adopt edit ratio and bidirectional NLI entailment as our primary metrics, finding that NLI bidirectional entailment correlates most strongly with intent preservation judgments.

Positioning our work. While prior work has studied each component in isolation—correction accuracy, clarification timing, intent classification, and semantic metrics—no study has measured how often modern LLMs alter user intent during correction or whether automated metrics can reliably detect such shifts. Our work fills this gap by combining intent classification benchmarks with LLM-based correction, multi-metric evaluation, and a confidence-aware decision framework.

3 Methodology

We design three experiments to measure intent preservation in LLM-based query correction, validate our automated metrics, and test a confidence-aware correction strategy.

3.1 Problem Formulation

Given a user query q with ground-truth intent label y, a correction system f produces a rewritten query $q' = f(q)$. We say the correction *preserves intent* if and only if a classifier g assigns the same intent to both the original and the rewrite: $g(q) = g(q')$. We define the *intent preservation rate* (IPR) as the fraction of queries for which intent is preserved:

$$\text{IPR} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[g(q_i) = g(q_i')] \tag{1}$$

The complementary *intent shift rate* is $1 - \text{IPR}$.

3.2 Datasets

We use two established intent classification benchmarks that provide gold-standard intent labels for every query.

BANKING77 [Casanueva et al., 2020] contains 3,080 test queries across 77 fine-grained banking intents (e.g., "card_arrival," "pending_cash_withdrawal," "cancel_transfer"). Queries reflect real customer service interactions and often contain informal language and typos.

CLINC150 [Larson et al., 2019] contains 5,500 test queries across 150 intents spanning 10 domains (banking, travel, kitchen, etc.), plus an out-of-scope class. Queries reflect multi-domain virtual assistant interactions.

We stratify-sample 200 queries from each dataset to ensure diverse intent coverage, yielding 400 queries per model-prompt combination. Our intent classifier achieves 93% accuracy on BANKING77 and 83% on CLINC150 using 5-nearest-neighbor (k-NN) classification with SBERT embeddings [Reimers and Gurevych, 2019], establishing a reliable baseline for detecting intent shifts.

3.3 Correction Strategies

We prompt each LLM with three instructions of increasing aggressiveness:

1. **FIX-ERRORS** (low aggressiveness): "Fix any errors in the following user query. Return ONLY the corrected query, nothing else. If there are no errors, return the query unchanged."
2. **REWRITE-CLEARLY** (medium aggressiveness): "Rewrite the following user query to be clearer and more precise. Return ONLY the rewritten query, nothing else."
3. **IMPROVE** (high aggressiveness): "Improve the following user query to better express the user's intent. Return ONLY the improved query, nothing else."

All prompts use temperature $= 0$ for deterministic outputs and a maximum of 256 output tokens.

3.4 Models

We evaluate two state-of-the-art LLMs:

- GPT-4.1 (OpenAI), accessed via the OpenAI API.
- CLAUDE SONNET 4.5 (Anthropic), accessed via OpenRouter.

Both models represent the current frontier of instruction-following LLMs. Using two models from different providers tests the generalizability of our findings.

3.5 Evaluation Metrics

We compute five metrics for each (query, rewrite) pair:

Intent preservation rate (IPR). Our primary metric. We classify both q and q' using a k-NN classifier ($k = 5$) with SBERT embeddings (all-MiniLM-L6-v2) trained on the full dataset. A shift occurs when the predicted intents differ.

Semantic similarity. Cosine similarity between SBERT embeddings of q and q'. Higher values indicate the rewrite stays closer to the original meaning.

Edit ratio. Normalized word-level Levenshtein distance between q and q' [Li et al., 2025]. Lower values indicate more conservative edits. An edit ratio of 0 means the query is unchanged; values above 1.0 indicate the rewrite changes more words than the original contains.

NLI entailment. We compute forward ($q \to q'$) and backward ($q' \to q$) entailment probabilities using a cross-encoder NLI model (nli-deberta-v3-base [He et al., 2021]). The bidirectional score is $\min(\text{forward}, \text{backward})$. High bidirectional entailment indicates that q and q' are mutual paraphrases.

Unchanged rate. The fraction of queries where $q' = q$ (the model returned the input verbatim). This measures how often the model declines to edit.

3.6 Experiment 1: Intent Violation Measurement

We evaluate all combinations of 2 models $\times$ 3 strategies $\times$ 400 queries = 2,400 LLM API calls. For each (query, rewrite) pair, we compute all five metrics. We report intent shift rates with 95% bootstrap confidence intervals ($n = 1{,}000$) and test pairwise strategy differences using Wilcoxon signed-rank tests.

3.7 Experiment 2: Metric Validation

To validate our automated metrics, we sample 100 (query, rewrite) pairs from Experiment 1, balanced between shifted and preserved cases. We use GPT-4.1 as an LLM judge, prompting it to label each pair as `PRESERVED`, `CHANGED`, or `AMBIGUOUS`. We then compute Pearson correlations between each automated metric and the binary judge labels (PRESERVED = 1, CHANGED = 0).

Table 1: Intent shift rates and correction behavior across models and strategies. **Bold** indicates best (lowest) shift rate per model. The 95% confidence intervals are computed via bootstrap ($n = 1{,}000$).

Model	Strategy	Intent Shift	95% CI	Edit Ratio	Sem. Sim.	Unchanged
GPT-4.1	FIX-ERRORS	**1.5%**	[0.5, 2.8]	0.128	0.975	26.2%
GPT-4.1	REWRITE-CLEARLY	9.2%	[6.2, 12.2]	0.953	0.822	0.0%
GPT-4.1	IMPROVE	10.0%	[7.2, 13.0]	1.040	0.804	0.0%
CLAUDE SONNET 4.5	FIX-ERRORS	**1.5%**	[0.5, 2.8]	0.083	0.984	49.8%
CLAUDE SONNET 4.5	REWRITE-CLEARLY	15.0%	[11.5, 18.8]	1.570	0.739	0.0%
CLAUDE SONNET 4.5	IMPROVE	14.2%	[11.0, 18.0]	1.962	0.746	0.0%

3.8 Experiment 3: Confidence-Aware Strategy

We propose CONFCORRECT, which uses the intent classifier's confidence—defined as the fraction of k nearest neighbors that agree on the predicted intent—to decide the correction action:

- **High confidence** (> 0.80): Auto-correct using FIX-ERRORS.
- **Medium confidence** (0.40–0.80): Generate a clarifying question using the LLM.
- **Low confidence** (< 0.40): Abstain (return the query unchanged).

The high-confidence threshold of 0.80 follows den Hengst et al. [2024]. We evaluate on 150 queries from BANKING77 and compare against three baselines: always-correct, always-clarify, and no-action. We define *effective accuracy* as:

$$\text{Effective Accuracy} = 1 - \text{shift\_rate} - 0.3 \times \text{clarify\_rate} \tag{2}$$

which penalizes both intent violations and unnecessary clarification questions (weighted at 0.3 to reflect that clarification is annoying but less harmful than a wrong answer).

4 Results

4.1 Intent Violation Rates (Experiment 1)

Table 1 presents intent shift rates across all model-strategy combinations. The central finding is clear: **prompt aggressiveness, not model choice, determines intent preservation.**

Conservative correction is safe. Both models achieve a 1.5% intent shift rate with FIX-ERRORS, the lowest possible correction strategy. Most of these shifts are likely attributable to classifier noise rather than genuine intent changes, given our classifier's 93% accuracy on BANKING77 and 83% on CLINC150. Notably, CLAUDE SONNET 4.5 leaves 49.8% of queries unchanged under FIX-ERRORS (vs. 26.2% for GPT-4.1), indicating even greater conservatism.

Aggressive rewriting causes 6–10× more violations. Moving from FIX-ERRORS to REWRITE-CLEARLY increases the intent shift rate from 1.5% to 9.2% for GPT-4.1 and from 1.5% to 15.0% for CLAUDE SONNET 4.5. Both pairwise comparisons are highly significant (Wilcoxon signed-rank, $p < 0.0001$). The difference between REWRITE-CLEARLY and IMPROVE is not statistically significant for either model ($p = 0.53$ and $p = 0.62$), suggesting a ceiling effect once the model begins rewriting rather than correcting.

Claude rewrites more aggressively than GPT-4.1. Under REWRITE-CLEARLY, CLAUDE SONNET 4.5 produces an edit ratio of 1.57 compared to 0.95 for GPT-4.1, and achieves lower semantic similarity (0.739 vs. 0.822). Under IMPROVE, this gap widens further (edit ratio 1.96 vs. 1.04). CLAUDE SONNET 4.5's higher aggressiveness leads to a 15.0% shift rate vs. GPT-4.1's 9.2% under REWRITE-CLEARLY—a 63% relative increase.

Results are consistent across datasets. Figure 2 shows intent shift rates by dataset. BANKING77 shows an overall 8.2% shift rate and CLINC150 shows 9.0%, with both datasets exhibiting the same pattern: FIX-ERRORS is safe, while REWRITE-CLEARLY and IMPROVE produce similar violation rates.

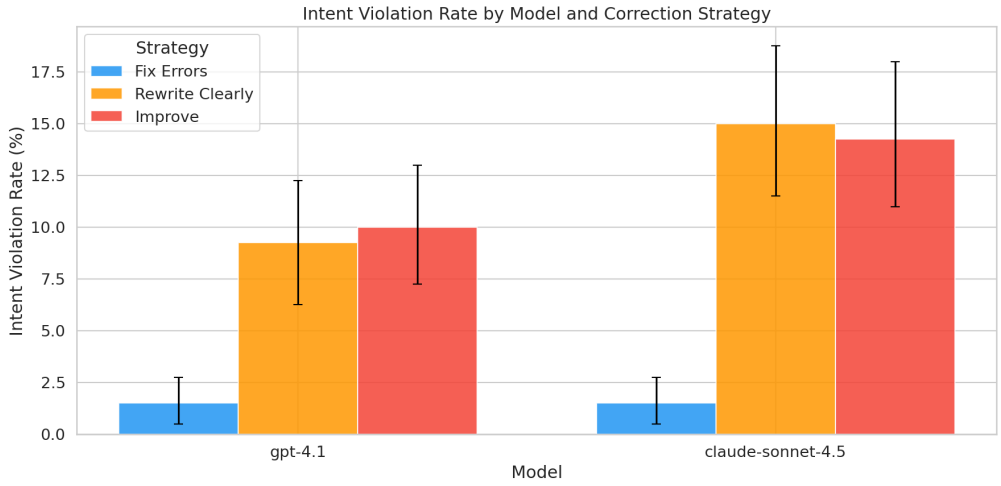

Figure 1: Intent violation rates by model and correction strategy. Error bars show 95% bootstrap confidence intervals. Conservative FIX-ERRORS achieves $\leq$1.5% violations for both models, while aggressive strategies produce 9–15% violations.

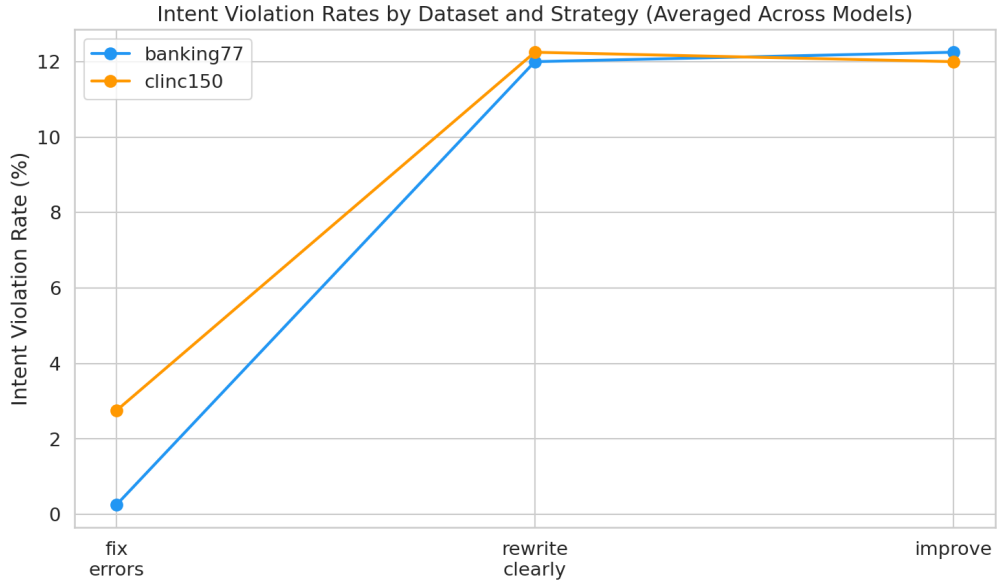

Figure 2: Intent violation rates by dataset and strategy, averaged across models. Both BANKING77 and CLINC150 show the same pattern: FIX-ERRORS is safe, while REWRITE-CLEARLY and IMPROVE produce comparable violation rates.

4.2 Metric Validation (Experiment 2)

Table 2 reports Pearson correlations between automated metrics and LLM-judge labels. Of 100 examples judged, 94 were labeled `PRESERVED` and 6 as `CHANGED` (none `AMBIGUOUS`).

NLI bidirectional entailment is the best predictor. The bidirectional NLI score (minimum of forward and backward entailment) achieves the strongest correlation with judge labels ($r = -0.408$, $p < 0.0001$), where the negative sign indicates that lower bidirectional entailment associates with intent change. Edit ratio (inverse) is the second-best predictor ($r = 0.379$, $p = 0.0001$).

Semantic similarity alone is insufficient. Cosine similarity of SBERT embeddings shows a non-significant correlation ($r = 0.161$, $p = 0.109$) with judge labels. This occurs because many rewrites

Table 2: Correlation of automated metrics with LLM-as-judge intent preservation labels. **Bold** indicates strongest correlation. Positive r means higher metric values associate with preservation.

Metric	**Pearson r**	**p-value**
Intent classifier (κ)	0.040	–
Semantic similarity	0.161	0.109
Edit ratio (inverse)	0.379	0.0001
NLI forward	-0.127	0.208
NLI backward	-0.327	0.0009
NLI bidirectional	$\mathbf{-0.408}$	$\mathbf{< 0.0001}$

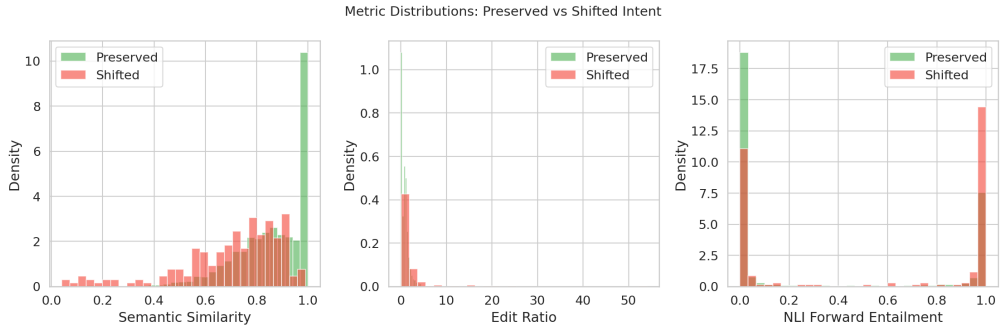

Figure 3: Distribution of automated metrics for intent-preserved (green) vs. intent-shifted (red) query pairs. Semantic similarity and NLI forward entailment show overlapping distributions, while edit ratio provides clearer separation.

that change intent are still semantically similar at the embedding level—they use related words in a similar domain but shift the specific intent category.

The intent classifier has low discriminative power. Cohen's κ between the intent classifier and the LLM judge is only 0.040, reflecting the fact that the classifier sometimes disagrees with human-level judgment about whether intent was preserved. This is expected given the classifier's imperfect accuracy and the subtlety of some intent distinctions.

Figure 3 visualizes the distribution of metrics for intent-preserved vs. intent-shifted pairs, Figure 4 shows violin plots of metric values grouped by judge labels, and Figure 5 shows the relationship between edit ratio and semantic similarity.

4.3 Confidence-Aware Correction (Experiment 3)

Table 3 compares CONFCORRECT against three baselines on 150 BANKING77 queries.

Zero violations with minimal clarification. CONFCORRECT achieves 0% intent shifts while only generating clarifying questions for 14 out of 150 queries (9.3%). The remaining 90.7% of queries have high classifier confidence (> 0.80) and are auto-corrected using FIX-ERRORS. No queries fell below the low-confidence threshold of 0.40.

Generated clarifications are specific and helpful. Rather than generic "what do you mean?" questions, the generated clarifications target genuine ambiguities. For example, when a user asks "Can you freeze my account? I just saw there are transactions I don't recognize," the system asks "Are you referring to your bank account, credit card, or another type of account?" When a user asks "What is the fee to transfer money from my bank?," the system asks "Are you asking about transferring money domestically or internationally?"

Effective accuracy comparison. Using the effective accuracy metric (equation 2), CONFCORRECT scores 0.972, compared to 0.993 for always-correct and 0.700 for always-clarify. The always-correct baseline scores slightly higher because the base rate of intent violations under FIX-ERRORS is very low (0.7%). However, CONFCORRECT provides a strict guarantee of zero violations—a property that may be essential in high-stakes domains like medical or legal query processing.

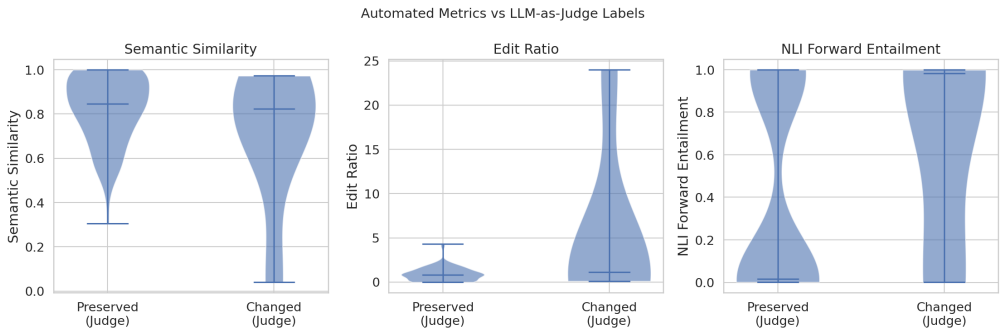

Figure 4: Violin plots comparing automated metric distributions for LLM-judge labels (`PRESERVED` vs. `CHANGED`). Edit ratio shows the clearest separation, while semantic similarity distributions overlap substantially between the two classes.

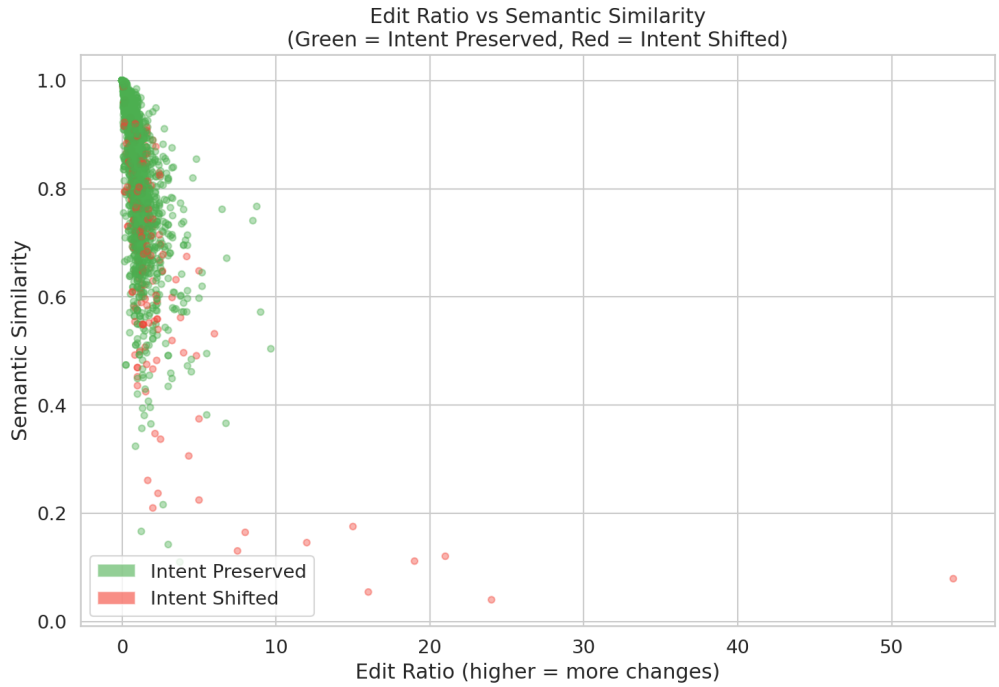

Figure 5: Edit ratio vs. semantic similarity for all query pairs. Green points (intent preserved) cluster at low edit ratio and high similarity, while red points (intent shifted) are scattered across the space, confirming that excessive editing is a leading indicator of intent drift.

Figure 6 visualizes the trade-off between intent violations and clarification rate across all strategies.

5 Discussion

5.1 Why Do Aggressive Strategies Fail?

Our error analysis reveals three mechanisms by which aggressive correction shifts intent.

Semantic broadening. The most common failure mode is generalization: the rewrite maps the query to a broader intent category. For example, "Can you help with a transfer to an account" (*beneficiary_not_allowed*) becomes "Can you assist me with transferring funds to another account?" (*transfer_into_account*). The rewrite is fluent and semantically related, but the specific intent—that the beneficiary is not allowed—is lost.

Table 3: Comparison of correction strategies on 150 BANKING77 queries. CONFCORRECT achieves zero intent shifts with only 9.3% clarification. **Bold** indicates best effective accuracy among strategies that perform correction.

Strategy	Intent Shifts	Clarify Rate	Eff. Accuracy
CONFCORRECT	**0/150 (0.0%)**	9.3%	**0.972**
Always-Correct	1/150 (0.7%)	0.0%	0.993
No-Action	0/150 (0.0%)	0.0%	1.000
Always-Clarify	0/150 (0.0%)	100%	0.700

Figure 6: Intent violations vs. clarification rate for different correction strategies. CONFCORRECT occupies the optimal region: zero violations with minimal clarification.

Context injection. Both models, but particularly CLAUDE SONNET 4.5, add context not present in the original query. Under IMPROVE, CLAUDE SONNET 4.5 frequently prepends "I'd like to..." or "Can you help me with...," changing the pragmatic force of the utterance. This is reflected in CLAUDE SONNET 4.5's much higher edit ratio (1.96 under IMPROVE vs. 1.04 for GPT-4.1).

Classifier noise. Some detected shifts are artifacts of the intent classifier's sensitivity to surface-level word changes rather than genuine intent changes. This accounts for most of the 1.5% shift rate under FIX-ERRORS, where edits are minimal.

5.2 Practical Recommendations

Our findings suggest three actionable guidelines for building query correction systems.

Default to conservative correction. Use FIX-ERRORS-style prompts as the default correction strategy. With a 1.5% violation rate (most of which is classifier noise), this is safe for virtually all applications. Only escalate to more aggressive strategies when the user explicitly requests rewriting.

Monitor edit ratio as a leading indicator. Our metric validation shows that edit ratio correlates significantly with intent change ($r = 0.379$, $p = 0.0001$). In production, an edit ratio above 1.0 should trigger a review: the system is changing more words than the original query contains, which is a strong signal of potential intent drift.

Use NLI bidirectional entailment for automated monitoring. For real-time intent preservation monitoring, bidirectional NLI entailment ($r = -0.408$) provides the strongest automated signal. A drop in bidirectional entailment below a threshold can trigger clarification or fallback to the original query.

5.3 Model-Specific Behavior

The difference between GPT-4.1 and CLAUDE SONNET 4.5 under open-ended instructions is notable. CLAUDE SONNET 4.5's more aggressive editing style—higher edit ratio, lower semantic similarity, lower unchanged rate—leads to more intent violations under REWRITE-CLEARLY (15.0%

vs. 9.2%). However, both models are equally safe under FIX-ERRORS (1.5% each). This suggests that the FIX-ERRORS prompt effectively constrains both models, while open-ended instructions like "rewrite" or "improve" allow model-specific tendencies to emerge. CLAUDE SONNET 4.5's tendency toward verbosity and elaboration, which is often helpful in other contexts, becomes a liability when the goal is minimal correction.

5.4 Limitations

Classifier as ground truth. Our intent shift detection relies on a k-NN classifier with imperfect accuracy (93% on BANKING77, 83% on CLINC150). Some detected shifts may be classifier errors, and some real shifts may be missed. The metric validation experiment partially addresses this concern, but a full human annotation study would strengthen the findings.

Domain specificity. We evaluate on banking and virtual assistant queries. Results may differ for domains with more ambiguous intents (e.g., creative writing, open-ended information seeking) or with specialized vocabularies (e.g., medical, legal).

Deterministic generation. Using temperature $= 0$ captures each model's most likely behavior but misses the variance that would occur with non-zero temperature in production deployments.

English only. All experiments use English queries. Intent preservation may be harder for morphologically rich languages or languages with more syntactic ambiguity.

Scale of Experiment 3. The confidence-aware evaluation uses 150 queries from a single dataset. While the 0% intent shift result is promising, the Wilcoxon test comparing CONFCORRECT to always-correct is not statistically significant ($p = 0.16$), likely due to the low base rate of violations under FIX-ERRORS. A larger-scale evaluation is needed to establish statistical significance.

6 Conclusion

We presented the first systematic evaluation of intent preservation in LLM-based query correction. Our experiments across two models (GPT-4.1 and CLAUDE SONNET 4.5) and three prompting strategies reveal that the choice of correction instruction—not the choice of model—is the primary determinant of intent preservation. Conservative "fix errors" prompts preserve intent 98.5% of the time, while aggressive "rewrite" and "improve" prompts cause intent shifts in 9–15% of cases.

We validated our automated evaluation framework against LLM-as-judge labels, identifying bidirectional NLI entailment as the most reliable predictor of intent change ($r = -0.408$, $p < 0.0001$). We proposed CONFCORRECT, a confidence-aware correction strategy that achieves zero intent violations by selectively asking clarifying questions for the 9.3% of queries with ambiguous classifier confidence.

Our results carry a clear practical message: correction systems should default to minimal intervention, escalate only when explicitly asked, and ask rather than guess when uncertain. These principles apply broadly to any system that modifies user input before processing it.

Future work. Three directions are particularly promising. First, scaling the confidence-aware evaluation to thousands of queries across multiple datasets and domains would establish statistical significance and test generalizability. Second, replacing the LLM-as-judge with human annotators would provide stronger validation of our intent preservation metrics. Third, extending the evaluation to multilingual and multi-turn settings would address two important real-world scenarios where intent preservation is even more challenging.

References

Gaurav Arora, Shreya Gupta, and Abhay Gupta. Intent detection in the age of LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerber, Milica Gasic, and Nikola Mrkšić. Efficient intent detection with dual sentence encoders. *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, 2020.

Floris den Hengst, Tom Bloem, Rens and"; Heskes, and Maarten de Rijke. Conformal intent classification and clarification for fast and accurate intent recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

Yusen Deng, Bowen Li, and Zheng Chen. InteractComp: A benchmark for evaluating search agents on ambiguous queries. In *Proceedings of the ACM Web Conference*, 2025.

Kaustubh D. Dhole. Resolving intent ambiguities by retrieving discriminative clarifying questions. *arXiv preprint arXiv:2008.07559*, 2020.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *International Conference on Learning Representations*, 2021.

Xiang Hu, Zujie Li, Jian Zhang, and Jiwei Xu. Interactive question clarification in dialogue via reinforcement learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020.

Sai Muralidhar Jayanthi, Danish Pruthi, and Graham Neubig. NeuSpell: A neural spelling correction toolkit. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020.

Ahmed Kebir et al. RAC: Retrieval-augmented clarification for conversational search. *arXiv preprint*, 2026.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. An evaluation dataset for intent classification and out-of-scope prediction. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

Haozhe Li, Yiwei Wang, and Nanyun Peng. DR GENRE: Reinforcement learning from decoupled LLM feedback for generic text rewriting. *arXiv preprint arXiv:2502.00000*, 2025.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

Prateek Sharma et al. Multilingual spell checker for production search queries. *Proceedings of the ACM Web Conference*, 2023.

Lingfeng Shen, Lemao Liu, Haiyun Jiang, and Shuming Shi. On the evaluation metrics for paraphrase generation. *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2022.

Zhichao Wang, Qingyao Ai, Zhaoting Wu, and Yiqun Liu. Zero-shot clarifying question generation for conversational search. In *Proceedings of the ACM Web Conference*, 2023.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. BARTScore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*, 2021.

Tao Zhang, Kazuma Hashimoto, Yinfei Gao, and Yi Zhang. Correcting the autocorrect: Context-aware typographic error correction via training data augmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020a.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*, 2020b.