CoAlign: Uncertainty Calibration of LLM for Geospatial Repartition

Zejun Xie[1], Zhiqing Hong[1], Wenjun Lyu[1],
Haotian Wang[2], Guang Wang[3]*, Desheng Zhang[1]

[1]Rutgers University, [2]JD Logistics, [3]Florida State University
{zx180, zh252, wl531, dz220}@cs.rutgers.edu, wanghaotian18@jd.com, guang@cs.fsu.edu

Abstract

With the rapid expansion of e-commerce and continuous urban evolution, *Geospatial Repartition*, dividing geographical regions into delivery zones, is essential to optimize various objectives, e.g., on-time delivery rate, for last-mile delivery. Recently, large language models (LLMs) have offered promising capabilities for integrating diverse contextual information that is beneficial for geospatial repartition. However, given the inherent uncertainty in LLMs, adapting them to practical usage in real-world repartition is nontrivial. Thus, we introduce CoAlign, a novel three-stage framework that calibrates LLM uncertainty to enable robust geospatial repartition by transforming the task into a ranking problem, integrating historical data with LLM-generated candidates. It first generates explainable candidate partitions with a multi-criteria strategy and then designs a novel conformal method to rank these candidates relative to historical partitions with coverage guarantees. Finally, CoAlign delivers candidates through an interactive decision support system. Extensive evaluation with real-world data shows that CoAlign effectively calibrates LLM uncertainty and generates partitions that better align with human feedback. Moreover, we have deployed CoAlign in one of the world's largest logistics companies, significantly enhancing their delivery operations by increasing candidate acceptance rates by 217% and improving on-time delivery rates by 3%. Our work provides a novel angle to address industrial geospatial decision-making tasks by calibrating LLM uncertainty.

1 Introduction

Geospatial Repartition refers to dynamically adjusting geographical regions into multiple delivery zones, supporting fundamental businesses, e.g., balanced order assignments, for logistics companies, e.g., Amazon (Amazon), SF Express (S.F. Express) and JD Logistics (JDL.COM). With rapid global e-commerce expansion, effective geospatial repartition is critical for ensuring online operational efficiency in logistics systems (Hong et al., 2022). Existing methods typically rely on manual adjustments by experts or algorithmic optimization using limited offline operational metrics, such as historical data, to balance order volumes or equalize working times (Guo et al., 2023; Zhang et al., 2024). In state-of-the-practice, algorithms generate multiple repartition candidates according to various offline metrics and recommend them to experts, who then decide to accept one candidate or manually devise an alternative. This operational paradigm has two major limitations: (i) Real-world operational constraints are significantly more complex and dynamic than offline metrics can capture, resulting in theoretically optimal partitions that are often infeasible for practical deployment (Figure 1 provides an illustrative example), leading to low acceptance rates in practice; (ii) Experts spend considerable time reviewing candidates to identify issues. Upon discovering problems, they must manually repartition, often leaving their valuable feedback unused. Our logistics partner reports candidate acceptance rates often below 10%, with experts spending over 15 hours monthly on reviews and manual repartition.

Facing these limitations, we identify an opportunity in the extensive contextual information—such as historical partitions and corresponding expert and worker feedback—accumulated by existing operational systems, reflecting real-world constraints. Recent advances have demonstrated the remarkable capability of large language models (LLMs) in extracting information, understanding context, and learning from interactive dialogues (Zhao et al., 2023; Manvi et al., 2024; Feng et al., 2024; Yamada et al., 2024). Thus, we aim to propose an interactive LLM approach capable of interpreting contextual information and interactively generating

*Corresponding author

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track), pages 254–263

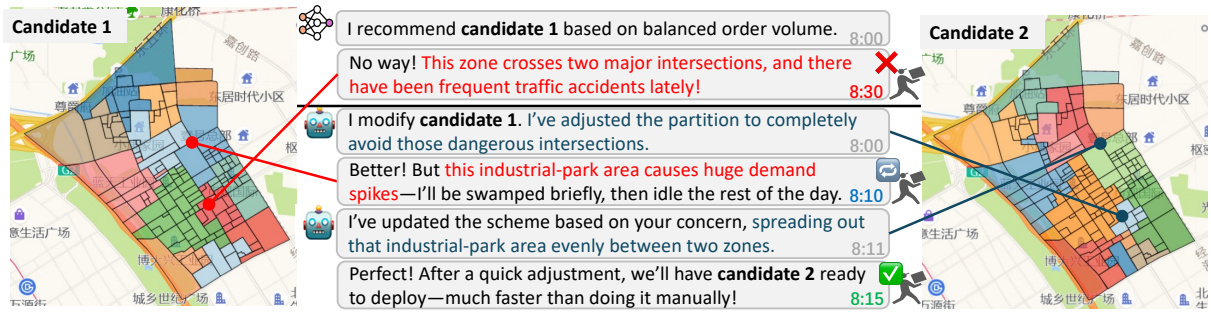

Figure 1: A real-world example demonstrates LLM-based interactive geospatial repartition. The dialogue highlights practical constraints, such as safety hazards at intersections and demand spikes in industrial areas, that current methods fail to capture. Our LLM-based approach understands these issues, enabling interactive refinement and enhancing the efficiency of experts through collaboration.

comprehensive, human-aligned repartitions, rather than merely optimizing offline metrics. Figure 1 illustrates differences between previous approaches and ours using one real-world example.

However, applying LLMs to geospatial repartition introduces uncertainty challenges, as LLM outputs inherently exhibit stochasticity, potentially producing plausible yet incorrect solutions without proper confidence measures. Given the excessive expert review time, an uncertainty-calibration method is necessary to ensure reliable high-quality candidates. Therefore, we present `CoAlign` (Conformal rAnking-based LLM Interactive Geospatial repartitioN), a novel three-stage framework. Firstly, we employ a Multi-Criteria pipeline that prompts an LLM to generate candidate partitions with detailed explanations and scoring across multiple metrics. Secondly, we design a conformal ranking algorithm to transform the initial LLM scores into rankings relative to historical partitions, and then create calibrated prediction sets with statistical coverage guarantees. Finally, we integrate these components into a human-in-the-loop decision-making system, enabling efficient and explainable collaboration between algorithmic candidates and human decision-makers.

Our contributions include: (i) A novel LLM-based framework, `CoAlign`, that integrates contextual information, provides comprehensive partition, and enables effective human-AI collaboration to address limitations in existing geospatial repartition systems; (ii) A novel conformal ranking design that transforms subjective LLM scores into reliable and explainable prediction sets with coverage guarantees; (iii) A comprehensive evaluation with real-world logistics data demonstrates that `CoAlign` effectively calibrates LLM uncertainty, achieving performance in offline metrics comparable to or surpassing state-of-the-art methods while producing partitions more closely aligned with human expert feedback. Furthermore, we deployed `CoAlign` across over **5,000** delivery stations in one of the largest logistics companies in the world. The A/B test results reveal significant improvements in online metrics (i.e., **3%** $\sim$ **10%**), candidate acceptance rates (i.e., **217%** increase), and decision efficiency (i.e., **56%** less human intervention and **25%** faster review).

2 Related Work

Geospatial Repartition. The expert manual partition approach leverages domain knowledge that performs well but is time-consuming. Algorithmic methods have evolved through several methodological paradigms. Traditional operations research approaches formulate this task as a combinatorial optimization problem with geometric constraints (Zhong et al., 2007; Carlsson and Devulapalli, 2013; Banerjee et al., 2022; Carlsson et al., 2024; Xie et al., 2025). More recently, data-driven methods offer improved scalability and automation, including graph neural networks (Guo et al., 2023) and deep reinforcement learning (Zheng et al., 2023b,b). However, these approaches optimize the partition with narrow offline metrics as objectives, failing to incorporate rich contextual information in real-world settings.

Uncertainty in LLM-based Decision Making. The application of LLMs to decision support systems has grown rapidly across domains including urban planning (Zhou et al., 2024; Li et al., 2024), and spatial-temporal data (Huang et al., 2022; Yang et al., 2024). These models excel at synthesizing complex, multi-modal information to generate creative solutions, but their deployment requires ro-

bust uncertainty quantification. Recent work introduced conformal prediction techniques (Shafer and Vovk, 2008; Vovk et al., 2005; Vovk, 2012) to measure and align uncertainty in LLM-based planners (Quach et al., 2023; Ren et al., 2023; Cherian et al., 2024) and they rely on LLM self-reported scores, which have shown inconsistency in complex tasks.

3 CoAlign Design

3.1 Intuition and Overview

Intuition. Extensive cognitive science and social choice research has consistently shown that humans provide more reliable comparative judgments (e.g., rankings) than absolute evaluations (e.g., scores) (Mussweiler, 2003; Arrow, 2012). Recent work on LLM-as-a-judge confirms this phenomenon in language models as well, showing higher consistency and robustness in relative ordering tasks (Liusie et al., 2024; Jiang et al., 2023; Wang et al., 2024). This insight inspired our approach: rather than calibrating raw LLM confidence scores directly, we developed a conformal prediction method tailored specifically for rankings (Luo and Zhou, 2024; Fermanian et al., 2025; Xu et al., 2025). By transforming the geospatial repartition problem into a relative ranking task between historical and newly generated partitioning schemes, we enable rigorous uncertainty quantification with statistical guarantees. This ranking-based paradigm integrates seamlessly with the existing uncertainty calibration method of LLM while addressing the uncertainty challenge of geospatial decision-making.

Overview. As shown in Figure 2, our geospatial repartition framework, CoAlign, includes three components: **Stage 1**: generating diverse partition candidates with a surrogate scoring model; **Stage 2**: calibrating uncertainty in candidate rankings via conformal prediction to form a reliable prediction set; and **Stage 3**: engaging a domain expert to review and decide the final partition based on this prediction set.

3.2 Multi-Criteria Partition Generation (MCPG)

In the geospatial repartition problem, each input instance X represents a geographic region with demands, constraints, and relevant attributes. The goal is to divide X into multiple delivery zones (partitions) that satisfy various operational criteria. In our approach, a large language model (LLM) is prompted to generate M candidate partitions

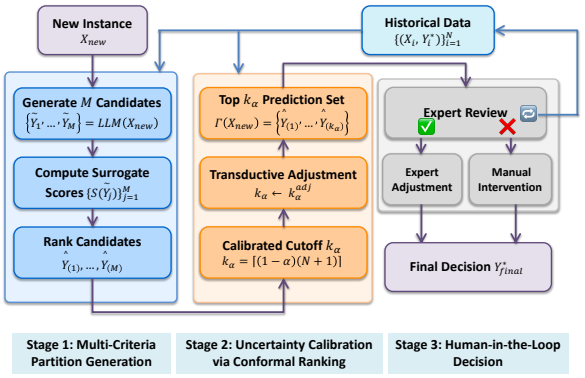

Figure 2: CoAlign Framework.

$\{\tilde{Y}_1, \ldots, \tilde{Y}_M\}$ for X. Each candidate partition $\tilde{Y}_j$ splits X into several zones, and each zone is composed of multiple *Areas of Interest* (AOIs) (e.g., neighborhoods or street clusters). We provide the LLM with rich context at both the region level and the AOI level: the prompt includes global features of X along with summary information for each AOI (such as historical demand profiles, road-network connectivity, or known bottleneck locations). This AOI-level contextual input helps the LLM reason about fine-grained spatial details when assigning AOIs to zones. The LLM generates each partition along with textual explanations and per-zone evaluations for multiple domain-specific criteria (for example, workload balance, demand coverage, or estimated travel time).

After generating candidate partitions, we verify spatial contiguity at the AOI level for each partition. We represent the region's AOIs as nodes in a graph $G = (V, E)$, where edges connect adjacent AOIs (for example, sharing a border or linked by a road). For each zone in a partition, we consider the set of AOIs assigned to that zone and check whether the induced subgraph is connected. In practice, we perform a graph traversal (e.g., breadth-first search) starting from one AOI in the zone and confirm that all other AOIs in the zone are reachable. If any zone is found to be disconnected (i.e., its AOI subgraph is not fully connected), the entire partition is rejected. To speed up this check, we employ simple heuristics. For instance, we precompute the connected components of G once per region; then any zone whose AOIs lie in more than one component can be immediately flagged as invalid without a full search. In our implementation, this filtering effectively removes partitions with non-contiguous zones while imposing minimal computational overhead.

Finally, we evaluate each candidate partition $\tilde{Y}_j$ by scoring it on each criterion $c_1, \ldots, c_L$ and combining these into a surrogate score $S(\tilde{Y}_j)$. This yields a ranked list of partitions.

3.3 Uncertainty Calibration via Conformal Ranking

We design a rank-based conformal prediction approach to ensure that our final set of top partitions (the *prediction set*) contains the true optimum Y^* with probability at least $1 - \alpha$. In essence, our algorithm treats the partition scoring model as directly producing a rank for the true partition among candidates and calibrates the uncertainty in that rank.

Calibration Set Design. From N historical instances $\{(X_i, Y_i^*)\}_{i=1}^N$, we run the same partition generation and scoring pipeline as used for new predictions. This yields a rank $\eta_i = \text{rank}(X_i, Y_i^*)$ for each instance i, where η_i is the position of the true optimum Y_i^* in the model's sorted list of candidate partitions for X_i. Intuitively, η_i represents the error made by the model on instance i—a small value means the model ranked the true partition highly, whereas a large η_i means the true partition was buried lower in the list.

Cutoff Determination. We sort the set of calibration ranks $\{\eta_i\}_{i=1}^N$ in nondecreasing order and determine the cutoff index $k_\alpha = \lceil (1-\alpha)(N+1) \rceil$. By construction, approximately $(1-\alpha)N$ of the calibration instances have Y^* ranked within the top-k_α positions of the candidate list. In other words, in most calibration examples the true optimum would be among the model's top-k_α predictions.

Transductive Adjustment. In practice, introducing a new instance can slightly shift the distribution of ranks because the model's ranking function may depend on the set of items being ranked. To safeguard the coverage guarantee in such a transductive setting, we adjust the cutoff k_α upward by a small margin if needed. Concretely, we simulate the effect of adding new test instances on the calibration ranks by randomly perturbing each η_i within a possible range of rank shifts, and choose an adjusted cutoff k_α^{adj} that still covers roughly $(1-\alpha)$ fraction of the simulated rank outcomes. This procedure yields a slightly larger prediction set size when necessary, ensuring our method remains valid even if the new instance(s) alter the ranking distribution.

Prediction Set Generation. For a new instance X_{new}, we generate M candidate partitions, compute each candidate's surrogate score $S(\tilde{Y}_j)$, and sort the candidates in descending order of S to obtain the ranked list $[\hat{Y}_{(1)}, \hat{Y}_{(2)}, \ldots, \hat{Y}_{(M)}]$. We then take the top k_α after any transductive adjustment as the *prediction set*: $\Gamma(X_{\text{new}}) = \{\hat{Y}_{(1)}, \ldots, \hat{Y}_{(k_\alpha)}\}$.

Theoretical Guarantee. Assume the calibration data $\{(X_i, Y_i^*)\}_{i=1}^N$ and the new instance(s) are exchangeable. Then for any $\alpha \in (0,1)$, the prediction set $\Gamma(X_{\text{new}})$ obtained by the above procedure satisfies

$$\Pr\{ Y_{\text{new}}^* \in \Gamma(X_{\text{new}}) \} \geq 1 - \alpha .$$

In other words, the method achieves the target marginal coverage level $1 - \alpha$ (Luo and Zhou, 2024). Moreover, consider a batch of m i.i.d. new instances with prediction sets constructed using the same calibration. With probability at least $1 - \beta$ (over the randomness of the calibration procedure), the false coverage rate is bounded as

$$\frac{1}{m} \sum_{i=1}^{m} \mathbf{1}\{ Y_{i,\text{new}}^* \notin \Gamma(X_{i,\text{new}}) \} \leq \alpha + \lambda_{N,m} ,$$

for some tolerance term $\lambda_{N,m} = O\left(\sqrt{\frac{\ln(Nm/\beta)}{Nm}}\right)$ that approaches 0 as $N, m \to \infty$ (Fermanian et al., 2025; Xu et al., 2025). In particular, in the limit of large sample sizes, the average miscoverage (error rate) on m new instances does not exceed α.

3.4 Human-in-the-Loop Decision

After constructing $\Gamma(X_{\text{new}})$ for a new instance, the system presents these top k_α candidate partitions to a domain expert, together with the scores $\{c_\ell(\hat{Y}_{(j)})\}$ and a brief LLM-generated explanation (if desired). The expert selects the best partition Y_{final}^* or indicates that none is satisfactory (in which case Y_{new}^* lies outside $\Gamma(X_{\text{new}})$—an event that, by design, should occur in at most α fraction of cases). Crucially, this feedback can be incorporated into the calibration set by adding $(X_{\text{new}}, Y_{\text{final}}^*)$ as a new example, along with its observed rank η_{new}.

Over time, the rank distribution and/or the surrogate score weights $\{w_\ell\}$ can be updated to better match expert preferences. In practice, if $\Gamma(X_{\text{new}})$ is empty or too small, we may adjust $\alpha \leftarrow \alpha + \Delta$ until at least one partition meets the expert's acceptance threshold, following the approach of (Vovk et al., 2005) for iterative significance tuning.

4 Evaluation in Last-mile Delivery

To evaluate the effectiveness of CoAlign, we conducted both offline evaluation and online deployment in collaboration with one of the world's largest logistics companies. Our evaluation aims to answer the following research questions:

RQ1: Operational Performance. How does CoAlign perform compared to methods specifically designed to optimize offline operational metrics?

RQ2: Uncertainty Calibration. Does CoAlign provide reliable prediction sets?

RQ3: Decision Efficiency. How does CoAlign improve human decision after deployment?

RQ4: Real-world Benefit. Does CoAlign improve the acceptance rate and online operational metrics after the deployment?

4.1 Data and Offline Evaluation Setup

Data Preparation. We conducted offline experiments using logistics-related data from October 2023 to June 2024 by our industry partner. Over this period, the company deployed various algorithm-generated partition recommendations across over 900 regions nationwide, logging 35,000+ repartitioning operations and 150,000+ algorithm-generated recommendations, with corresponding accept/reject decisions and comments by region managers. Each record includes: (i) **Context:** Station information (e.g., geospatial boundaries and operational constraints), current partition configuration, and historical logs (e.g., courier feedback); (ii) **Candidates:** Recommended partitions from prior heuristic algorithms; (iii) **Annotations:** Manager acceptance/rejection decisions, detailed feedback, and operational metrics (delivery order volume, courier working time, etc.).

Metrics. We evaluate our approach with 14 metrics across 4 types aligned with research questions. Table 1 summarizes directionalities and descriptions of these metrics. More detailed definitions of these metrics are provided in Appendix A.

Baselines and Training Setup. We split the dataset chronologically, using the first six months for training/calibration and the last two months for held-out testing. To contextualize our results, we compare CoAlign to several baselines that either represent the state-of-the-art or classic methods:

- **Heuristic-Only:** A manual or rule-based approach that divides regions via simple constraints.

Table 1: Evaluation metrics used in our experiments. *Metrics are defined as ratios to protect commercial privacy and normalized to [0,1] for easy comparison, except for those marked with *, which are non-negative.*

Type	Metric	Description
RQ1	↓ OVB	Coefficient of variation in order volume.
	↓ WTB	Coefficient of variation in working time.
	↓ WDB	Gini coef. of workload distribution.
	↑ MS	Maximum similarity between candidate set and deployed partition.
	↑ MSR*	Ratio of method MS to historical candidate MS.
RQ2	↓ PSR	Ratio of prediction set (PS) size to total candidate set size.
	↑ ECR	Proportion of true ranks covered in PS.
	↓ FCR	Proportion of ranks incorrectly covered in PS.
RQ3	↓ HIR	Proportion of cases requiring significant manual intervention.
	↓ RRT*	Ratio of current review time to historical average review time.
	↑ RAR	Proportion of algorithm recommendations accepted.
RQ4	↑ HER*	Ratio of post/pre-deploy HR efficiency.
	↑ PVR*	Ratio of post/pre-deploy pick-up volume.
	↑ OTR*	Ratio of post/pre-deploy on-time rate.

  We compare with two representative methods, CKmeans (Zhang et al., 2024) (a constrained clustering method) and CPSC (Joshi et al., 2012) (an A-star-based partitioning method).

- **DL-based Single/Multi Optimization:** Deep learning-based approaches that optimize one or multiple operational objectives (e.g., WTB or OVB). We compare with a DRL-based multi-optimization urban-planning method DRL (Zheng et al., 2023b,a) and a GNN-based single-optimization model E-partition (Guo et al., 2023), both trained on the same historical data without LLM-generated candidates.

- **LLM-Based Methods:** Two categories of LLM-based methods are used as baselines. The first does not include uncertainty calibration, including Vanilla (Zhao et al., 2023), the planner OPRO (Yang et al., 2024), and the multi-agent discussion-based solution LLM4PUP (Zhou et al., 2024). The second category incorporates uncertainty calibration for LLMs, specifically KnowNo (Ren et al., 2023), which uses conformal prediction in single-step uncertainty alignment (SUA) or multi-step uncertainty alignment (MUA) modes that differ from our conformal

ranking design.[1]

4.2 Offline Evaluation Results (RQ1&RQ2)

RQ1: Operational Performance. Table 2 presents 5 metrics on the test set. CoAlign performs best in WDB, MS and MSR, and second in OVB and WTB. Figure 3 illustrates CoAlign is competitive with DRL for OVB and E-partition for WTB in most regions, outliers cause minor variations. Hence, CoAlign (i) achieves comparable (OVB, WTB) or better (WDB) performance versus specialized DL approaches (DRL, E-partition), and (ii) delivers significantly stronger alignment (MS, MSR) than all baselines. We vary the LLM scale (4B, 10B, 81B) for selected LLM-based baselines. Table 3 reports MS and MSR. CoAlign and KnowNo-SUA with 81B models generally reach to exceed MSR=1.0, indicating that a larger model is critical for complex multi-criteria solutions. These results demonstrate that CoAlign effectively leverages human feedback to produce partitions closely aligned with humans, while simultaneously matching DL-based baselines in optimizing offline operational metrics.

Table 2: RQ1 results on five metrics. *MSR* > 1.0 *indicates* closer *alignment than historical recommendations.*

Method	OVB↓	WTB↓	WDB↓	MS↑	MSR↑
CKmeans	0.291	0.246	0.253	0.45	0.85
CPSC	0.318	0.278	0.269	0.44	0.80
DRL	**0.234**	0.182	0.227	0.59	0.98
E-partition	0.251	**0.165**	0.232	0.57	0.97
Vanilla	0.418	0.365	0.342	0.35	0.70
OPRO	0.368	0.302	0.321	0.38	0.75
LLM4PUP	0.385	0.287	0.311	0.39	0.78
KnowNo-SUA	0.283	0.244	0.211	0.65	1.05
KnowNo-MUA	0.321	0.278	0.290	0.46	0.92
CoAlign	0.245	0.168	**0.190**	**0.71**	**1.20**

RQ2: Uncertainty Calibration. Table 4 shows CoAlign achieves the best results in all 3 metrics. DRL shows the second-highest ECR but requires a larger set (PSR=0.32). KnowNo-SUA outperforms KnowNo-MUA, suggesting SUA is more stable for geospatial repartition tasks than MUA, likely due to weak causality between different times of historical records. Removing conformal ranking (CR) or mixing SUA/MUA consistently degrades coverage and inflates FCR. Notably, CoAlign achieves an FCR of 0.08, consistently below our preset threshold α=0.1. For $\alpha \in \{0.05, 0.10, 0.15\}$, smaller α boosts ECR but enlarges the prediction set (PSR). Setting $\alpha = 0.1$ is a balanced choice (ECR ≈ 0.92, PSR ≈ 0.20). Thus, CoAlign's use of CR indeed addresses geospatial repartition's complexity better than existing uncertainty alignment strategies.

[1]Due to our partner company's policy, we can only use its internal ChatRhino LLMs with 4B, 10B, and 81B parameter sizes. All LLM results use LLM-81B unless otherwise noted.

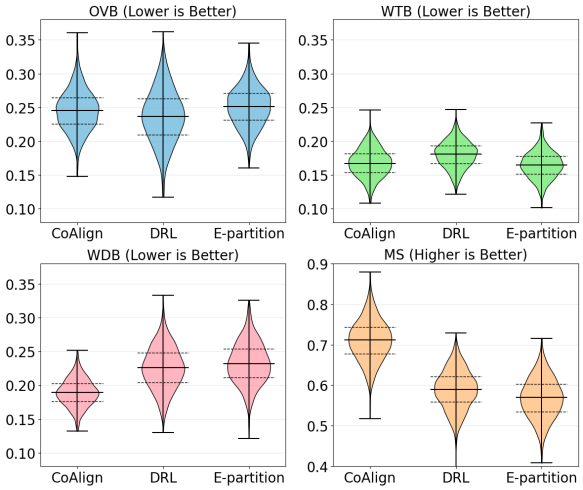

Figure 3: Offline Metrics of key baselines v.s. CoAlign.

Table 3: LLM size v.s. MS and MSR.

	MS↑ / MSR↑		
	4B	10B	81B
Vanilla	0.29 / 0.55	0.32 / 0.63	0.35 / 0.70
OPRO	0.32 / 0.60	0.35 / 0.68	0.38 / 0.75
LLM4PUP	0.34 / 0.61	0.33 / 0.65	0.39 / 0.78
KnowNo-SUA	0.43 / 0.82	0.50 / 0.87	0.65 / 1.05
KnowNo-MUA	0.40 / 0.72	0.42 / 0.75	0.46 / 0.82
CoAlign	0.44 / 0.89	0.58 / 0.95	**0.71 / 1.20**

4.3 Real World Deployment (RQ3 & RQ4)

We integrated CoAlign into a human-AI collaboration platform at over 5,000 stations. We performed an A/B test from July to August 2024, splitting stations into the **Control Group** (deployed

Table 4: RQ2 results for uncertainty quantification.

Method	PSR↓	ECR↑	FCR↓
DRL	0.32	0.85	0.22
LLM4PUP	0.27	0.65	0.25
KnowNo-SUA	0.28	0.78	0.15
KnowNo-MUA	0.35	0.72	0.20
CoAlign w/o CR	0.40	0.60	0.28
CoAlign w/o CR + MUA	0.38	0.70	0.23
CoAlign w/o CR + SUA	0.26	0.75	0.16
CoAlign	**0.20**	**0.92**	**0.08**

state-of-the-practice pipeline) and the **Experimental Group** (deployed CoAlign).

Results. Table 5 shows the results of the control group (Ctrl.) and the experimental group (Exp.) before (Pre.) and after (Post.) deploying CoAlign. All metrics in the control group remained stable before and after deploying CoAlign. The recommendation acceptance rate (RAR) in the experimental group increases from 0.06 to 0.19. Even when the recommended partition is not directly accepted, the AI-generated result remains close to the optimum and allows timely human feedback, lowering the final human intervention rate (HIR) from 0.94 to 0.41. Overall, the review time drops by about 25%, with 90% of cases requiring fewer than 3 interaction rounds. Meanwhile, the real-world benefit metrics all exceed 1.0, confirming notable gains in HR efficiency, pickup volume, and on-time rate.

Table 5: A/B test results of the CoAlign deployment.

	HIR↓	RRT↓	RAR↑	HER↑	PVR↑	OTR↑
Ctrl. (Pre.)	0.93	1.00	0.07	1.00	1.00	1.00
Ctrl. (Post.)	0.92	1.02	0.08	1.01	1.02	1.01
Exp. (Pre.)	0.94	1.00	0.06	1.00	1.00	1.00
Exp. (Post.)	**0.41**	**0.75**	**0.19**	**1.12**	**1.06**	**1.04**

Hence, CoAlign significantly improves acceptance (**RQ3**) and operational metrics (**RQ4**) compared to the baseline pipeline, largely due to its ability to incorporate human feedback effectively and produce partitions closer to expert preferences.

Remark. We observed intriguing patterns where LLM-generated partitions occasionally proposed "unorthodox" solutions characterized by near-equal zone sizes—partitions rarely produced by purely metric-driven baselines. Although these atypical recommendations were not always optimal by conventional standards, they enriched the solution space and were sometimes favored by experts for their ease of manual fine-tuning. For instance, as shown in Figure 4, experts actively encouraged LLMs to generate partitions that isolate the 4 areas highlighted by red circles. This observation suggests a broader insight: the value of LLMs extends beyond achieving higher acceptance rates through conventionally "correct" partitions; they also effectively address diverse practical requirements encountered in daily operations.

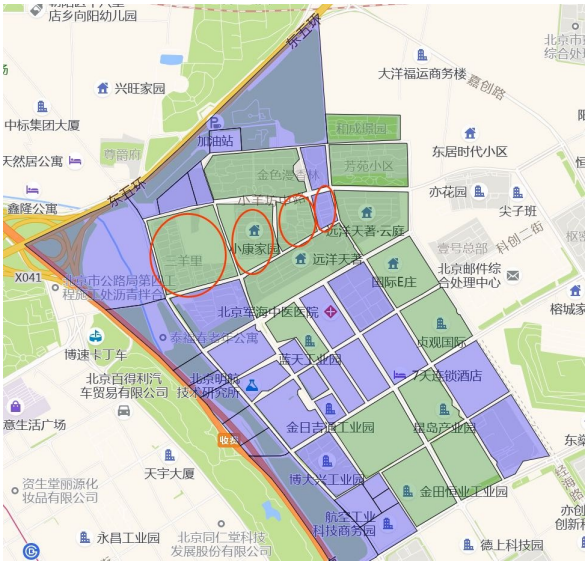

Figure 4: A real-world case of "unorthodox" partitions.

5 Conclusion and Limitation

We propose CoAlign for calibrating uncertainty in LLM explicitly for geospatial repartition. CoAlign generates comprehensive and human-aligned partitions via integrating diverse contextual information and maintains robust uncertainty calibration of LLM through a novel conformal ranking approach. Extensive offline evaluations demonstrate that CoAlign achieves superior performance across multiple offline metrics. More importantly, we have deployed CoAlign in a leading logistics company for geospatial repartition in over 5,000 delivery stations, generating positive societal and economic impact.

Although CoAlign already achieves strong performance suitable for real-world deployment without additional pre-/post-training of LLM, its success relies on the availability of rich, domain-specific data. For broader and more complex tasks, recent methods like RAG (Gao et al., 2023), CoT (Wei et al., 2022), or RLHF (Ouyang et al., 2022) could boost efficiency and performance, even with smaller models. Exploring these techniques is a promising direction for future work.

Acknowledgments

The authors would like to thank anonymous reviewers for their insightful comments and valuable suggestions. This work is partially supported by the National Science Foundation under Grant No. 2047822, 1952096, and 2411151.

References

Amazon. Amazon. Webpage.

Kenneth J. Arrow. 2012. *Social Choice and Individual Values*. Yale University Press, New Haven.

Dipayan Banerjee, Alan L. Erera, and Alejandro Toriello. 2022. Fleet sizing and service region partitioning for same-day delivery systems. 56(5):1327–1347.

John Carlsson and Raghuveer Devulapalli. 2013. Dividing a territory among several facilities. *INFORMS Journal on Computing*, 25:730–742.

John Gunnar Carlsson, Sheng Liu, Nooshin Salari, and Han Yu. 2024. Provably good region partitioning for on-time last-mile delivery. *Oper. Res.*, 72(1):91–109.

John Cherian, Isaac Gibbs, and Emmanuel Candes. 2024. Large language model validity via enhanced conformal prediction methods. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Jie Feng, Yuwei Du, Jie Zhao, and Yong Li. 2024. Agentmove: Predicting human mobility anywhere using large language model based agentic framework. *Preprint*, arXiv:2408.13986.

Jean-Baptiste Fermanian, Pierre Humbert, and Gilles Blanchard. 2025. Transductive conformal inference for ranking. *Preprint*, arXiv:2501.11384.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Baoshen Guo, Shuai Wang, Haotian Wang, Yunhuai Liu, Fanshuo Kong, Desheng Zhang, and Tian He. 2023. Towards equitable assignment: Data-driven delivery zone partition at last-mile logistics. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 4078–4088, New York, NY, USA. Association for Computing Machinery.

Zhiqing Hong, Guang Wang, Wenjun Lyu, Baoshen Guo, Yi Ding, Haotian Wang, Shuai Wang, Yunhuai Liu, and Desheng Zhang. 2022. Cominer: nationwide behavior-driven unsupervised spatial coordinate mining from uncertain delivery events. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '22, New York, NY, USA. Association for Computing Machinery.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. 2022. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pages 9118–9147. PMLR.

JDL.COM. Jdl.com. Webpage.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, Toronto, Canada. Association for Computational Linguistics.

Deepti Joshi, Leen-Kiat Soh, and Ashok Samal. 2012. Redistricting using constrained polygonal clustering. *IEEE Transactions on Knowledge and Data Engineering*, 24(11):2065–2079.

Zhonghang Li, Lianghao Xia, Jiabin Tang, Yong Xu, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024. Urbangpt: Spatio-temporal large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5351–5362.

Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, St. Julian's, Malta. Association for Computational Linguistics.

Rui Luo and Zhixin Zhou. 2024. Trustworthy classification through rank-based conformal prediction sets. *Preprint*, arXiv:2407.04407.

Rohin Manvi, Samar Khanna, Gengchen Mai, Marshall Burke, David B. Lobell, and Stefano Ermon. 2024. GeoLLM: Extracting geospatial knowledge from large language models. In *The Twelfth International Conference on Learning Representations*.

Thomas Mussweiler. 2003. Comparison processes in social judgment: mechanisms and consequences. *Psychological review*, 110(3):472.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. 2023. Conformal language modeling. *arXiv preprint arXiv:2306.10193*.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. 2023. Robots that ask for help: Uncertainty alignment for large language model planners. In *7th Annual Conference on Robot Learning*.

S.F. Express. S.f. express. Webpage.

Glenn Shafer and Vladimir Vovk. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3).

Vladimir Vovk. 2012. Conditional validity of inductive conformal predictors. In *Asian Conference on Machine Learning*, pages 475–490. PMLR.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*, volume 29. Springer.

Yikun Wang, Rui Zheng, Haoming Li, Qi Zhang, Tao Gui, and Fei Liu. 2024. Rescue: Ranking LLM responses with partial ordering to improve response generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 261–272, Bangkok, Thailand. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Zejun Xie, Wenjun Lyu, Yiwei Song, Haotian Wang, Guang Yang, Yunhuai Liu, Tian He, Desheng Zhang, and Guang Wang. 2025. Scalable area difficulty assessment with knowledge-enhanced ai for nationwide logistics systems. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25, page 2713–2724, New York, NY, USA. Association for Computing Machinery.

Yunpeng Xu, Mufang Ying, Wenge Guo, and Zhi Wei. 2025. Two-stage risk control with application to ranked retrieval. *Preprint*, arXiv:2404.17769.

Yutaro Yamada, Yihan Bao, Andrew Kyle Lampinen, Jungo Kasai, and Ilker Yildirim. 2024. Evaluating spatial understanding of large language models. *Transactions on Machine Learning Research*.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2024. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*.

Jinlei Zhang, Ergang Shan, Lixia Wu, Jiateng Yin, Lixing Yang, and Ziyou Gao. 2024. An end-to-end predict-then-optimize clustering method for stochastic assignment problems. *IEEE Transactions on Intelligent Transportation Systems*, 25(9):12605–12620.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *Preprint*, arXiv:2303.18223.

Yu Zheng, Yuming Lin, Liang Zhao, Tinghai Wu, Depeng Jin, and Yong Li. 2023a. Spatial planning of urban communities via deep reinforcement learning. *Nat. Comput. Sci.*, 3(9):748–762.

Yu Zheng, Hongyuan Su, Jingtao Ding, Depeng Jin, and Yong Li. 2023b. Road planning for slums via deep reinforcement learning. KDD '23, page 5695–5706, New York, NY, USA. Association for Computing Machinery.

Hongsheng Zhong, Randolph Hall, and Maged Dessouky. 2007. Territory planning and vehicle dispatching with driver learning. *Transportation Science*, 41:74–89.

Zhilun Zhou, Yuming Lin, Depeng Jin, and Yong Li. 2024. Large language model for participatory urban planning. *Preprint*, arXiv:2402.17161.

A Metric Definition

This section details the metrics used in our experiments, organized by research question (RQ). All metrics are either normalized to the range $[0, 1]$ or defined as ratios for ease of comparison. Metrics marked with '*' may exceed 1.0 or be nonnegative values rather than strictly bounded in $[0, 1]$.

RQ1: Operational Effectiveness

(i) **OVB (Order Volume Balance)**

$$\text{OVB} = \frac{\sqrt{\frac{1}{|\mathcal{Z}|}\sum_{z\in\mathcal{Z}}\left(v_z - \bar{v}\right)^2}}{\bar{v}},$$

where $\mathcal{Z}$ is the set of subregions, v_z is the order volume of subregion z, and $\bar{v}$ is the mean order volume. Lower OVB indicates better balance.

(ii) **WTB (Working Time Balance)**

$$\text{WTB} = \frac{\sqrt{\frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}\left(t_c - \bar{t}\right)^2}}{\bar{t}},$$

where $\mathcal{C}$ is the set of couriers, t_c is the working time of courier c, and $\bar{t}$ is the mean working time. Lower WTB indicates better time balance.

(iii) **WDB (Workload Distribution Balance)**

$$\text{WDB} = \text{Gini}\big(\{\, w_z \mid z \in \mathcal{Z}\}\big),$$

where w_z is the workload of subregion z. Lower WDB indicates more uniform workload distribution.

(iv) **MS* (Maximum Similarity)**

$$\text{MS} \;=\; \max_{Y \in \Gamma(X)} \; \text{sim}\big(Y,\, Y^*\big),$$

measuring the highest similarity (e.g. IoU or overlap) between the prediction set $\Gamma(X)$ and the deployed partition Y^*. Higher is better.

(v) **MSR* (Method Similarity Ratio)**

$$\text{MSR} \;=\; \frac{\text{MS(Method)}}{\text{MS(Historical)}},$$

the ratio of our method's MS to the historical candidate's MS. A value above 1.0 indicates the method produces partitions more aligned with final deployments than past baselines.

RQ2: Uncertainty Quantification

(i) **PSR (Prediction Set Ratio)**

$$\text{PSR} \;=\; \frac{\text{Avg size of prediction set}}{\text{Avg number of total candidates}},$$

indicating how large the top-k_α set is relative to all generated partitions. Lower PSR indicates a more selective set.

(ii) **ECR (Empirical Coverage Rate)**

$$\text{ECR} \;=\; \frac{\#\{\, X : Y^* \in \Gamma(X)\}}{\#\{\, X\}},$$

the fraction of instances whose true optimum Y^* appears in the prediction set. Higher ECR is better.

(iii) **FCR (False Coverage Rate)**

$$\text{FCR} \;=\; \frac{\#\{\text{incorrectly covered instances}\}}{\#\{\, X\}},$$

the fraction of instances where the prediction set includes a suboptimal or invalid partition that might mislead decisions. Lower is better.

RQ3: Decision Efficiency

(i) **HIR (Human Intervention Rate)**

$$\text{HIR} \;=\; \frac{\#\{\text{cases needing manual edits}\}}{\#\{\text{total cases}\}},$$

representing the proportion of partitions that required substantial manual adjustment beyond the recommended set.

(ii) **RRT* (Relative Review Time)**

$$\text{RRT} \;=\; \frac{T_{\text{current}}}{T_{\text{baseline}}},$$

where T_{current} is the average manager review time under the new system, and T_{baseline} is the pre-deployment average. A value below 1 indicates faster reviews.

(iii) **RAR (Recommendation Acceptance Rate)**

$$\text{RAR} \;=\; \frac{\#\{\text{accepted recommendations}\}}{\#\{\text{total recommendations}\}},$$

the fraction of algorithm-proposed partitions eventually adopted (with or without minor edits). Higher is better.

RQ4: Deployment Benefit

(i) **HER* (HR Efficiency Ratio)**

$$\text{HER} \;=\; \frac{\text{HR}_{\text{post}}}{\text{HR}_{\text{pre}}},$$

the ratio of post-deployment to pre-deployment human resource efficiency. A value above 1 implies improved workforce productivity.

(ii) **PVR* (Pick-up Volume Ratio)**

$$\text{PVR} \;=\; \frac{\text{PV}_{\text{post}}}{\text{PV}_{\text{pre}}},$$

the ratio of post- to pre-deployment pickup volume. Values above 1 indicate increased pickup throughput.

(iii) **OTR* (On-time Ratio)**

$$\text{OTR} \;=\; \frac{\text{OT}_{\text{post}}}{\text{OT}_{\text{pre}}},$$

the ratio of on-time deliveries post- vs. pre-deployment. Values above 1 reflect improved timeliness.