CCF Transactions on Pervasive Computing and Interaction
https://doi.org/10.1007/s42486-026-00272-w

REGULAR PAPER

Context-aware task assignment in mobile crowdsensing via LLM-enhanced dual-view graph contrastive learning

Yujie Yan[1,2,3] · Haifeng Jiang[1,2,3] · Chaogang Tang[1,2,3] · Shuo Xiao[1,2,3] · Huaming Wu[4] · Ruidong Li[5]

Received: 7 July 2026 / Accepted: 1 September 2026
© China Computer Federation (CCF) 2026

Abstract
Mobile crowdsensing (MCS) is an important data collection paradigm for smart cities and Internet of Things applications. Existing task assignment methods often rely on physical distance or historical interactions, and therefore struggle with implicit semantic constraints, sparse interaction data, and cold-start participants. Although Large language models (LLMs) enhance semantic understanding, their direct deployment for online task assignment entails substantial inference overhead and offers no guarantee of feasibility in real-world execution. To address these problems, this paper proposes an LLM-DVGCL (Large Language Model-Enhanced Dual-View Graph Contrastive Learning) method for context-aware task assignment in MCS. LLM-DVGCL uses an LLM as an offline, evidence-bearing candidate-relation proposer rather than an online scheduler. The proposed framework maps LLM-proposed candidate semantic relations from task-side records into standardized ontology entities, constructs a heterogeneous semantic knowledge graph for graph learning, and initializes cold-start participants through inductive attribute aggregation. LLM-DVGCL further separates semantic requirement matching from contextual execution feasibility by modeling them as two complementary views, and optionally applies cross-view consistency regularization to candidate worker-task matching. Experiments on real interaction data and independently annotated public text evaluate the dynamic dual-view assignment component and the LLM-assisted semantic front end in two complementary tracks. The results support dynamic dual-view ranking and selective semantic enhancement while revealing clear limitations for equipment and skill extraction.

Keywords Mobile crowdsensing · Context-aware task assignment · Large language models · Heterogeneous graph learning · Dual-view contrastive learning · Smart city sensing

✉ Haifeng Jiang
jhfeng@cumt.edu.cn

Yujie Yan
ts25170130p31@cumt.edu.cn

Chaogang Tang
cgtang@cumt.edu.cn

Shuo Xiao
sxiao@cumt.edu.cn

Huaming Wu
whming@tju.edu.cn

Ruidong Li
liruidong@ieee.org

[1] School of Computer Science and Technology/School of Artificial Intelligence, China University of Mining and Technology, Xuzhou, China

[2] Mine Digitization Engineering Research Center of the Ministry of Education, Xuzhou, China

[3] Jiangsu Provincial Industrial Technology Engineering Center for Intelligent Sensing and Emergency IoT in Underground Space, Xuzhou, China

[4] The Center for Applied Mathematics, Tianjin University, Tianjin, China

[5] The Institute of Science and Engineering, Kanazawa University, Kanazawa, Japan

Published online: 26 October 2026

1 Introduction

With the rapid development of Internet-of-Things technologies and the widespread use of smart mobile devices, Mobile Crowdsensing (MCS) has become an important data collection paradigm for smart cities, environmental monitoring, and emergency response (Abdeddine et al. 2025; Capponi et al. 2019). Early MCS studies further define it as a human-powered sensing and computing paradigm that exploits participants, mobile devices, embedded sensors, and computing resources to collect large-scale, fine-grained, and low-cost data (Ganti et al. 2011; Guo et al. 2015). In this process, task assignment is the key bridge between sensing demand and human participants. The efficiency and matching quality of task assignment directly affect the quality of sensed data and the operational effectiveness of the overall system.

Unlike conventional online recommendation, MCS task assignment is constrained by the physical environment. Candidate suitability depends not only on historical preferences or textual similarity, but also on location, temporal availability, device status, mobility, and the surrounding environment. MCS task assignment is therefore a pervasive computing decision problem constrained by mobile context. It must jointly model task demand, participant capability, and real-world execution conditions.

Existing MCS task assignment methods include location-based heuristic strategies (Liu et al. 2025; Li et al. 2023), collaborative filtering (He et al. 2017), Deep Reinforcement Learning (DRL) (Liu et al. 2024), incentive mechanisms (Shang et al. 2024), and Graph Neural Networks (GNNs) (Anand and Maurya 2025). As sensing tasks become increasingly professional and complex, these methods still face three challenges. First, task descriptions often contain implicit equipment, skill, and execution-related semantic cues that are not explicitly recorded in structured fields (Han et al. 2023; Wang et al. 2019). Second, newly registered users or cold-start candidates have few historical interactions, causing weak cold-start representations. Third, semantic relevance does not necessarily imply real-world execution feasibility. A candidate may be semantically matched to a task but unable to complete it because of spatial or temporal constraints, temporary unavailability, device limitations, or dynamic operational conditions such as workload and recent activity. Ignoring these factors may lead to assignments that appear reasonable from text alone but fail in real-world deployment.

To address these challenges, we propose LLM-DVGCL (LLM-Enhanced Dual-View Graph Contrastive Learning). MCS task assignment requires a model that can capture task-side semantic information while satisfying real-world constraints such as distance, time, device status, and availability. Large Language Models (LLMs) provide strong natural language understanding and semantic reasoning abilities, and can assist in identifying candidate semantic relations from unstructured task descriptions. However, LLM inference is expensive, its outputs may be uncertain, and it does not directly enforce spatial–temporal feasibility. This paper therefore uses the LLM as an offline semantic enhancement module rather than an online scheduler. The proposed semantic relations are grounded into a predefined ontology, and then used to build a heterogeneous semantic knowledge graph. A dual-view graph contrastive learning module separately models semantic matching and contextual execution feasibility, and finally ranks candidate workers for each task.

The main contributions of this paper are as follows.

1. We propose LLM-DVGCL, a context-aware candidate worker matching framework for MCS. The framework unifies task-side semantic information, candidate attributes, and dynamic mobile context under the same task assignment objective, thereby addressing implicit constraint recognition, cold-start representation, and semantic-contextual mismatch in complex urban sensing tasks.
2. We design an offline LLM semantic enhancement and ontology grounding mechanism. It proposes candidate semantic relations from task descriptions and maps supported relations into standardized semantic entities. This avoids directly using free-form LLM outputs for online scheduling, reduces online inference burden, and makes semantic information easier to encode and audit.
3. We design a dual-view matching structure that decouples semantic matching from contextual execution feasibility. The semantic view captures the fit between task requirements and candidate capabilities, whereas the contextual execution feasibility view captures mobile context constraints such as location, time, device status, availability, workload and service quality. Cross-view consistency regularization is incorporated as an auxiliary training objective rather than a necessary source of performance improvement.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the system model and problem formulation. Section 4 describes the proposed LLM-DVGCL method. Section 5 reports the experimental results and analysis. Section 6 concludes the paper and discusses future work.

2 Related work

Task assignment in mobile crowdsensing has long focused on accurate worker-task matching under data sparsity, heterogeneous user capabilities, and complex task constraints (Zhao et al. 2023). Existing studies can be broadly categorized into MCS task allocation mechanisms, cold-start recommendation and graph representation learning, LLM-enhanced semantic matching, and multi-view graph contrastive learning. These studies provide important foundations, but they still have limitations in jointly handling implicit semantic constraints, mobile spatial–temporal constraints, dynamic availability, and cold-start participant representations.

Early MCS task assignment methods mainly focused on spatial distance and coverage. They usually adopted location-based greedy algorithms or heuristic strategies to assign tasks to nearby candidates or candidates with low coverage cost (Liu et al. 2025; Li et al. 2023). As task types became more complex, DRL-based task allocation methods and incentive-mechanism-based recruitment methods were introduced to improve allocation decisions and participant engagement (Liu et al. 2024; Shang et al. 2024). These methods can update allocation policies in dynamic environments, but they often require continuous trial-and-error. Under sparse positive samples, delayed feedback, and cold-start conditions, they may lead to inefficient exploration and fail to provide timely and reliable matching.

Recent studies published in CCF Transactions on Pervasive Computing and Interaction (TPCI) have also emphasized robust and context-aware task allocation in pervasive sensing scenarios. For example, multi-agent MCS with pervasive machines studies robust task allocation among heterogeneous machine participants (Luo et al. 2023), and ContinuousSensing studies human–robot collaborative task allocation with task migration (Li et al. 2024). These studies suggest that MCS task assignment should not be treated as a purely recommendation-oriented problem. It must handle participant heterogeneity, task migration, physical environment constraints, and execution failures. This observation is aligned with our motivation to model task assignment from a context-aware perspective, but our work focuses on the semantic interpretation of task-side logs and cold-start candidate matching under semantic and mobile contextual constraints.

To alleviate cold-start and data sparsity, transfer learning, meta-learning, and heterogeneous graph modeling have been widely explored (Zhang et al. 2501). Meta-transfer methods can transfer shared knowledge across tasks or scenarios and improve adaptation to new users or new tasks (Lee et al. 2019). However, when a task description contains complex unstructured implicit constraints, the transferred knowledge may fail to capture task-specific requirements accurately. Heterogeneous Information Networks (HINs) can incorporate users, tasks, devices, and skills into a unified graph and use explicit attributes to alleviate cold-start problems (Shi et al. 2019). Nevertheless, existing HIN methods often rely on relatively direct feature fusion. Under high negative-sample ratios, attribute noise, and semantic-contextual conflicts, such fusion can still produce mismatches.

From the perspective of recommendation, MCS task assignment can be viewed as a matching and ranking problem between task requirements and user capabilities. Collaborative filtering, matrix factorization, and LightGCN can model user-task interactions (Koren et al. 2009; He et al. 2020), but their performance depends heavily on existing interaction edges. For cold-start users, node representations easily degenerate because no historical neighbors are available. Graph contrastive learning methods such as LightGCL enhance sparse graph representations through structural perturbation and self-supervision (Wu et al. 2021; Cai et al. 2023). However, randomly augmented views do not necessarily correspond to the semantic requirements or execution constraints that govern MCS task assignment, making it difficult to interpret the source of mismatches in task assignment.

LLMs have recently been used to enhance recommendation and semantic matching (Hou et al. 2024; Lin et al. 2025). Methods such as LLMRec use LLM-based textual understanding to improve cold-start recall (Wei et al. 2024). These approaches can use semantic information in task descriptions, but in MCS task assignment, LLMs are better used as offline semantic enhancement modules rather than as direct online schedulers. The final assignment decision is constrained not only by textual similarity but also by geographic distance, time windows, device status, user availability, and quality-related mobile context (Zhao et al. 2023; Lin et al. 2025).

Multi-view graph representation learning provides a useful direction for modeling heterogeneous constraints. Existing graph contrastive learning methods usually construct augmented views through edge dropout, node masking, or structure perturbation (Wu et al. 2021; Cai et al. 2023). Disentangled graph collaborative filtering also shows that separating different latent factors can reduce feature entanglement (Wang et al. 2020). In MCS task assignment, however, views should not be generated only by random perturbation. They should correspond to meaningful constraints in the target system. This paper therefore treats semantic requirement matching and contextual execution feasibility as two task-specific views.

In summary, existing methods have strengths in online exploration, transfer adaptation, heterogeneous attribute modeling, graph structure enhancement, and semantic

understanding. However, they do not fully integrate implicit task-side semantic information, cold-start candidate representation, and semantic-contextual mismatch modeling. This paper integrates LLM-based semantic enhancement, heterogeneous semantic graph learning, and dual-view graph contrastive learning under a candidate worker-task matching objective. The LLM assists in proposing task-side semantic relations; attribute anchors initialize cold-start representations; and the dual-view structure decouples semantic matching from contextual execution feasibility for context-aware task assignment in complex MCS scenarios.

3 System model and problem formulation

To address the challenges of cold-start participants, underused task-side semantic information, and mobile-context modeling, we develop the LLM-DVGCL task assignment model. This section introduces the system architecture, problem definition, LLM-enhanced heterogeneous semantic knowledge graph, inductive node initialization, contextual execution feasibility constraints, and task assignment objective. The main notation used throughout the paper is summarized in Table 1.

3.1 Overall architecture and task assignment definition

As shown in Fig. 1, LLM-DVGCL contains four components: LLM-based implicit semantic extraction, heterogeneous graph construction and inductive aggregation, contextual execution feasibility modeling, and dual-view representation with matching output. These components correspond to task-side semantic relation proposal, cold-start representation initialization, and joint semantic and contextual matching. The candidate filtering and online ranking stage is also consistent with spatial crowdsourcing formulations in which task assignment is constrained by worker locations and online task arrivals (Kazemi and Shahabi 2012; Tong et al. 2016).

From the task assignment perspective, the system needs to generate a ranked list of candidate workers for each task. Semantic enhancement, contextual modeling, and dual-view contrastive learning are therefore all designed to improve the scoring and ranking of candidate worker–task pairs.

LLM-based implicit semantic extraction. MCS tasks are usually recorded through task type, short description, location, and time fields. The system uses an LLM as an offline semantic extraction module. It does not use the raw generated phrases as final attributes. Instead, it identifies candidate semantic relations from task-side records and maps retained relations into standardized ontology entities through ontology grounding:

$$\left(E_t^{req}, S_t^{req}\right) = f_{LLM}\left(x_t\right) \tag{1}$$

In Eq. (1), x_t denotes the raw task-side record of task t. The function $f_{LLM}(\cdot)$ first extracts candidate semantic relations from x_t and then grounds retained relations into the predefined ontology. The output E_t^{req} is the standardized equipment requirement set of task t, and S_t^{req} is the corresponding skill requirement set. The extracted phrases are then mapped to unified device and skill ontologies. These relations are treated as candidate semantic information rather than automatically verified ground-truth requirements. The final assignment result is produced jointly by graph representation learning, contextual execution feasibility filtering, and a matching scoring network.

Heterogeneous graph construction and inductive aggregation. Based on the standardized equipment and skill tags, the system constructs a heterogeneous semantic knowledge graph containing users, tasks, equipment, and skills. For users without historical interactions, LLM-DVGCL does not rely on historical user-task neighbors. Instead, it initializes their representations from their equipment and skill attributes, as formally defined in Sect. 3.3. The equipment and skill embeddings are learned from the global graph. This mechanism allows a new user to obtain a task-matching representation even when no historical interaction record is available.

Contextual execution feasibility modeling. LLM-DVGCL filters users who do not satisfy basic execution conditions. For a task t, the system uses available execution-context information, including distance, task time window, device status, user availability, and dynamic operational signals when available, to characterize contextual feasibility and derive the candidate set C_t. This step reduces invalid candidates for the following matching network and prevents semantically similar but physically infeasible users from entering the final recommendation list.

Dual-view representation and matching output. The system decouples semantic requirement matching and contextual execution feasibility into two views. The semantic view captures whether the capabilities of user u satisfy the requirements of task t, whereas the contextual execution-feasibility view represents whether the same candidate pair u, t can be executed under mobile context constraints. The model can additionally use an InfoNCE-based cross-view consistency objective as an auxiliary training regularizer. The semantic-view representation and the contextual execution-feasibility representation of the same candidate pair are treated as a positive pair, while representations from

Table 1 Main notation

Symbol	Description
U	Set of sensing users or candidate workers
U_{new}	Set of cold-start users
T	Set of sensing tasks
E	Set of equipment entities
S	Set of skill entities
x_t	Raw text or task-side record of task
$f_{LLM}(\cdot)$	LLM-based extraction and ontology-grounding function
$\left(E_t^{req}, S_t^{req}\right)$	Ontology-aligned equipment and skill candidate-relation sets retained for task
$G = (V, E_g)$	Heterogeneous semantic knowledge graph
V	Node set of the heterogeneous semantic knowledge graph
E_g	Edge set of the heterogeneous semantic knowledge graph
A^{ue}, A^{us}	User-equipment and user-skill adjacency matrices
A^{te}, A^{ts}	Task-equipment and task-skill semantic-relation adjacency matrices
R	Historical user-task interaction matrix
(E_u, S_u)	Equipment and skills owned by user u
$\boldsymbol{e}_i$	Learnable embedding of equipment entity
$\boldsymbol{s}_j$	Learnable embedding of skill entity j
$AGG(\cdot)$	Attribute aggregation function
h_u^{attr}	Attribute-based representation of user u obtained by aggregating equipment and skill embeddings
$\boldsymbol{id}_u$	Learnable-identity embedding of user u
$h_u^{(0)}$	Initial representation of user u after attribute-identity fusion
$h_i^{(0)}$	Initial representation of node , including $h_u^{(0)}$ for user u and $h_t^{(0)}$ for task
h_u^{sem}, h_t^{sem}	Semantic-view representations of user u and task
$z_{u,t}^{sem}$	Pair-level semantic-view representation of candidate pair u, t
$x_{u,t}^{ctx}$	Pair-level contextual execution-feasibility feature vector of candidate pair u, t
$z_{u,t}^{ctx}$	Pair-level contextual execution-feasibility representation of candidate pair u, t
$z_{u,t}$	Joint representation of candidate pair u, t
$d_{u,t}$	Geographic distance between user u and task
d_{max}	Maximum matching distance
C_t^d	Distance-feasible candidate user set for task
C_t	Final candidate user set for task after contextual filtering
$\overline{d}_{u,t}$	Normalized distance feature of candidate pair u, t
$\Delta t_{u,t}$	Matching degree between the availability of user u and the time window of task
$a_{u,t}$	Availability indicator of user u within the time window of task
$m_{u,t}^{equip}$	Equipment-satisfaction indicator or feature of user u for task
q_u	Historical response quality or service-quality proxy of user u
$x_{u,t}^{dyn}$	Available pre-decision dynamic operational feature vector of candidate pair u, t
$p_{u,t}$	Predicted matching probability of candidate pair u, t
$y_{u,t}$	True or proxy matching label of candidate pair u, t
Ω	Set of training candidate user-task pairs
B	Training batch used for in-batch contrastive learning
$sim(\cdot, \cdot)$	Cosine similarity function
L_{task}	Supervised task assignment loss
L_{cl}	Cross-view consistency regularization loss
L_{total}	Joint optimization loss
$\alpha, \beta, \lambda, \tau$	Attribute-identity fusion weight, positive-sample compensation weight, contrastive regularization coefficient, and contrastive temperature
P	Predicted user-task matching probability matrix

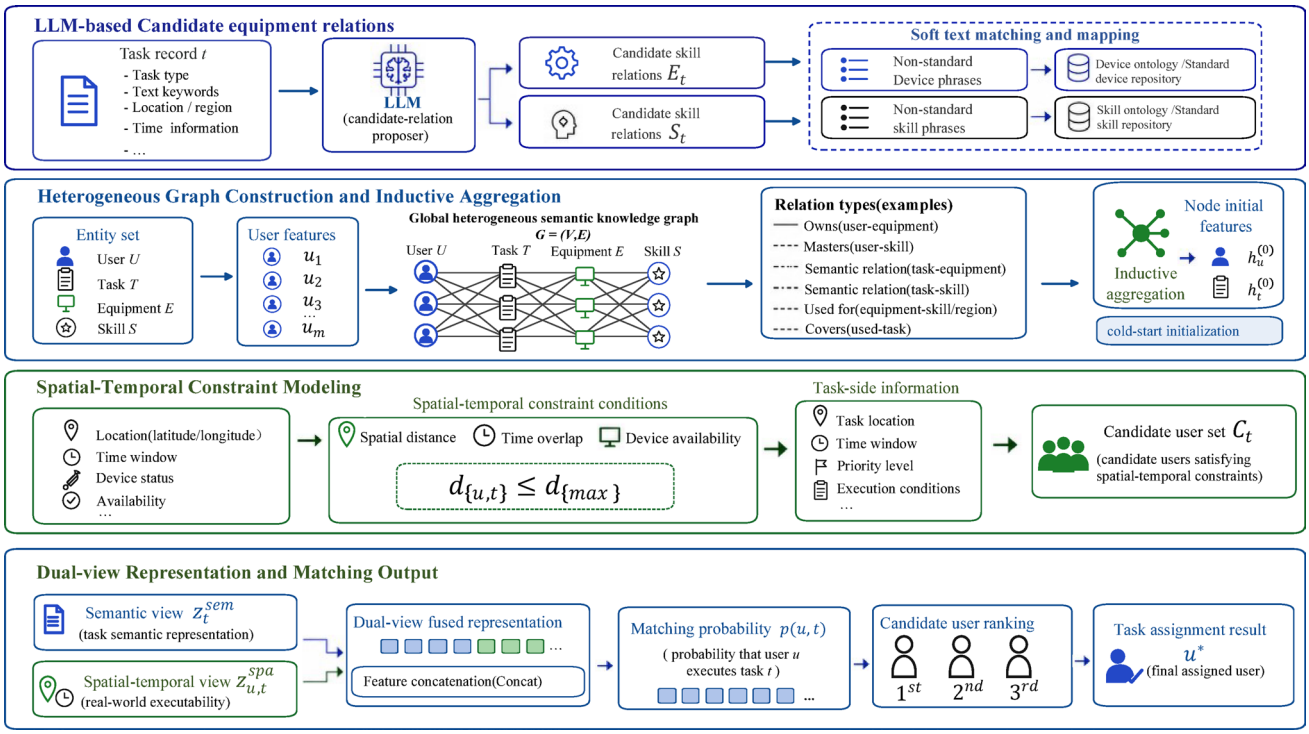


Fig. 1 Overall architecture of LLM-DVGCL

different candidate pairs in the same batch are treated as in-batch negatives.

3.2 LLM-enhanced heterogeneous semantic knowledge graph

We construct a heterogeneous semantic knowledge graph containing users, tasks, equipment, and skills. The graph carries ontology-aligned semantic relations extracted from task-side records and retained after semantic processing, and provides the basis for inductive representation and semantic-view construction.

The node set V consists of four types of entities, namely $V = U \cup T \cup E \cup S$. The user set is $U = \{u_1, u_2, \ldots, u_N\}$, where N denotes the number of users. The task set is $T = \{t_1, t_2, \ldots, t_M\}$, where M denotes the number of tasks. The equipment entity set is $E = \{e_1, e_2, \ldots, e_P\}$, where P denotes the number of equipment types. The skill entity set is $S = \{s_1, s_2, \ldots, s_Q\}$, where Q denotes the number of skill types.

For tensor computation in GNNs, graph relations are formalized as four binary adjacency matrices. The user-equipment ownership matrix is $A^{ue} \in \{0,1\}^{N \times P}$, where $A^{ue}_{i,j} = 1$ if user u_i owns equipment entity j, and $A^{ue}_{i,j} = 0$ otherwise. The user-skill matrix is $A^{us} \in \{0,1\}^{N \times Q}$, where $A^{us}_{i,j} = 1$ if user u_i masters skill entity j, and $A^{us}_{i,j} = 0$ otherwise. Similarly, the task-equipment semantic-relation matrix $A^{te} \in \{0,1\}^{M \times P}$ records whether an ontology-aligned equipment relation is retained for graph construction. Specifically, $A^{te}_{i,j} = 1$ if a semantic relation between task t_i and equipment entity j is retained after semantic processing, and $A^{te}_{i,j} = 0$ otherwise. The task-skill semantic-relation matrix $A^{ts} \in \{0,1\}^{M \times Q}$ is defined analogously, where $A^{ts}_{i,j} = 1$ if task t_i is inferred to require skill entity j, and $A^{ts}_{i,j} = 0$ otherwise. In addition, the system retains the historical assignment interaction matrix $R \in \{0,1\}^{N \times M}$ to record successful user-task matches.

3.3 Inductive node representation initialization

In MCS task assignment, cold-start means that a new user has no historical task acceptance record, making it difficult to infer execution capability. Traditional GNNs rely on user-task interaction edges for message propagation. When a new user has no historical neighbors, the representation can become unreliable. Given a cold-start user set and historical interaction matrix, the historical interaction row of any new user satisfies

$$R_{u,:} = 0, u \in U_{new} \tag{2}$$

where $R_{u,:}$ denotes the historical interaction row of user u and U_{new} denotes the cold-start user set. This condition indicates that a cold-start user u has no historical task neighbors

for message propagation. To alleviate this problem, LLM-DVGCL uses semantic attributes such as equipment and skills for inductive initialization. For a cold-start user, we first construct an attribute-based representation from the known equipment set and skill set:

$$h_u^{attr} = \mathrm{AGG}\left(\{\boldsymbol{e}_i \mid i \in E_u\} \cup \{\boldsymbol{s}_j \mid j \in S_u\}\right) \tag{3}$$

In Eq. (3), E_u is the equipment set of user u, and S_u is the skill set of user u. For each equipment entity $i \in E_u$, $\boldsymbol{e}_i$ denotes its learnable embedding. For each skill entity $j \in S_u$, $\boldsymbol{s}_j$ denotes its learnable embedding. The function $AGG(\cdot)$ aggregates these attribute embeddings to obtain the attribute-based representation h_u^{attr}. The final initial representation $h_u^{(0)}$ is obtained by fusing h_u^{attr} with a learnable identity embedding, as described in Sect. 4.2. This attribute-based representation allows users without historical interactions to obtain a meaningful initialization for task matching. This initialization follows graph convolution and inductive graph representation learning, where local structure and node attributes are used to construct embeddings (Kipf and Welling 2017; Hamilton et al. 2017).

3.4 Mobile contextual execution feasibility constraints

MCS tasks differ from ordinary online assignment because their execution is constrained not only by physical space and time but also by changing operational conditions. Even if a candidate is semantically compatible with a task, a long distance, an incompatible time window, temporary unavailability, or a high workload may reduce execution feasibility. Therefore, this paper models contextual feasibility before candidate ranking.

Specifically, the geographic distance between the location of user u and the location of task t is calculated from their latitude and longitude using the Haversine formula (Sinnott 1984), and a maximum reachable radius is set. Only users satisfying the distance constraint enter the candidate set:

$$C_t^d = \{u \in U \mid d_{u,t} \le d_{max}\} \tag{4}$$

where C_t^d denotes the distance-feasible candidate set for task t, $d_{u,t}$ is the geographic distance between user u and task t, and d_{max} is the maximum acceptable range. On this basis, the system applies basic hard constraints on time, device status, and availability:

$$C_t = \{u \in U \mid d_{u,t} \le d_{max},\ a_{u,t} = 1,\ m_{u,t}^{equip} = 1\} \tag{5}$$

where C_t denotes the final candidate set after contextual filtering. The indicator $a_{u,t} = 1$ means that user u satisfies the time-window or availability constraint of task t, and $m_{u,t}^{equip} = 1$ means that the device status or capability of user u satisfies the requirement of task t. This constraint reduces candidates that are semantically relevant but contextually infeasible.

These hard constraints define a basic feasible candidate set. Additional dynamic operational signals, such as historical location, regional or hourly affinity, workload or backlog, recent acceptance, and historical completion characteristics, are incorporated into the contextual representation when available. All such historical features are computed using information available before the evaluated task is accepted.

3.5 Dual-view representation and task assignment objective

Based on the heterogeneous semantic graph, contextual constraints, and inductive initialization, task assignment is modeled as a dual-view user-task matching problem. The semantic view captures the compatibility between task-side semantic relations and user attributes. The contextual execution feasibility view captures pair-level conditions such as distance, time window, device status, service quality, and available dynamic operational signals. Different from directly concatenating all features, dual-view modeling preserves the meaning of different constraints and enables the model to learn both semantic compatibility and contextual execution feasibility.

The input includes the global heterogeneous semantic graph and its adjacency matrices, including user-equipment, user-skill, task-equipment, and task-skill relations. The system also generates candidate user sets according to contextual constraints. For a given task t and its candidate users, the model outputs a matching probability $p_{u,t}$ for each candidate pair u, t. This probability is computed by the dual-view matching network defined in Sect. 4.5 and is used for ranking and assignment.

Because positive matching labels are often much fewer than negative or sampled alternative pairs, this paper uses Weighted Binary Cross-Entropy (WBCE) as the task matching loss:

$$L_{task} = -\sum_{(u,t)\in\Omega} \left[\beta y_{u,t} \log p_{u,t} + (1 - y_{u,t}) \log (1 - p_{u,t})\right] \tag{6}$$

In Eq. (6), Ω denotes the set of training candidate pairs. The variable $y_{u,t}$ is a binary matching label whose interpretation depends on the evaluation setting; it may denote a proxy match in the legacy controlled benchmark or an observed interaction label in the real-interaction evaluation. The variable $p_{u,t}$ is the predicted matching probability, and β is the positive-sample compensation weight. When cross-view contrastive regularization is enabled, the supervised

assignment loss and the cross-view alignment loss are jointly optimized:

$$L_{total} = L_{task} + \lambda L_{cl} \tag{7}$$

where L_{total} is the final training objective, L_{task} is the supervised matching loss, L_{cl} is the cross-view contrastive loss, and λ controls the strength of the optional cross-view regularizer. Setting $\lambda = 0$ reduces the objective to supervised dual-view matching without contrastive regularization. The role of this regularizer is evaluated separately in the experiments and is not assumed to provide a consistent performance improvement.

4 The proposed LLM-DVGCL method

Based on the model definition in Sect. 3, this section presents the LLM-DVGCL algorithm. As shown in Fig. 2, the method includes input representation, LLM-based implicit semantic extraction, inductive semantic feature fusion and heterogeneous knowledge graph construction, contextual execution feasibility encoding, cross-view contrastive alignment, cross-view representation concatenation, matching scoring, and optimization. These components jointly support context-aware scoring of candidate worker–task pairs. In particular, the LLM-based implicit semantic extraction module is executed offline before graph construction and candidate ranking, rather than being involved in the online scheduling process.

4.1 Input representation

Given the sensing user set U and task set T, this paper models task assignment as a matching scoring problem for a candidate user-task pair u, t. For each candidate pair u, t, the model learns the matching probability $p_{u,t}$ from semantic attributes, including available equipment- and skill-related information, and contextual attributes, including location, time, and available dynamic operational signals. Candidates are ranked according to $p_{u,t}$, and the recommended worker is selected under task-specific constraints. This formulation ensures that semantic enhancement and contextual modeling serve the same task assignment objective, while contrastive learning can additionally be used as an auxiliary regularization mechanism.

4.2 Inductive semantic feature fusion and semantic-view construction

The textual semantic view does not directly encode raw text. It is constructed based on LLM-based implicit semantic extraction and inductive feature fusion. Traditional task assignment methods mainly rely on explicit geographic coordinates, task categories, or historical interactions. They struggle to identify equipment-, skill-, modality-, or

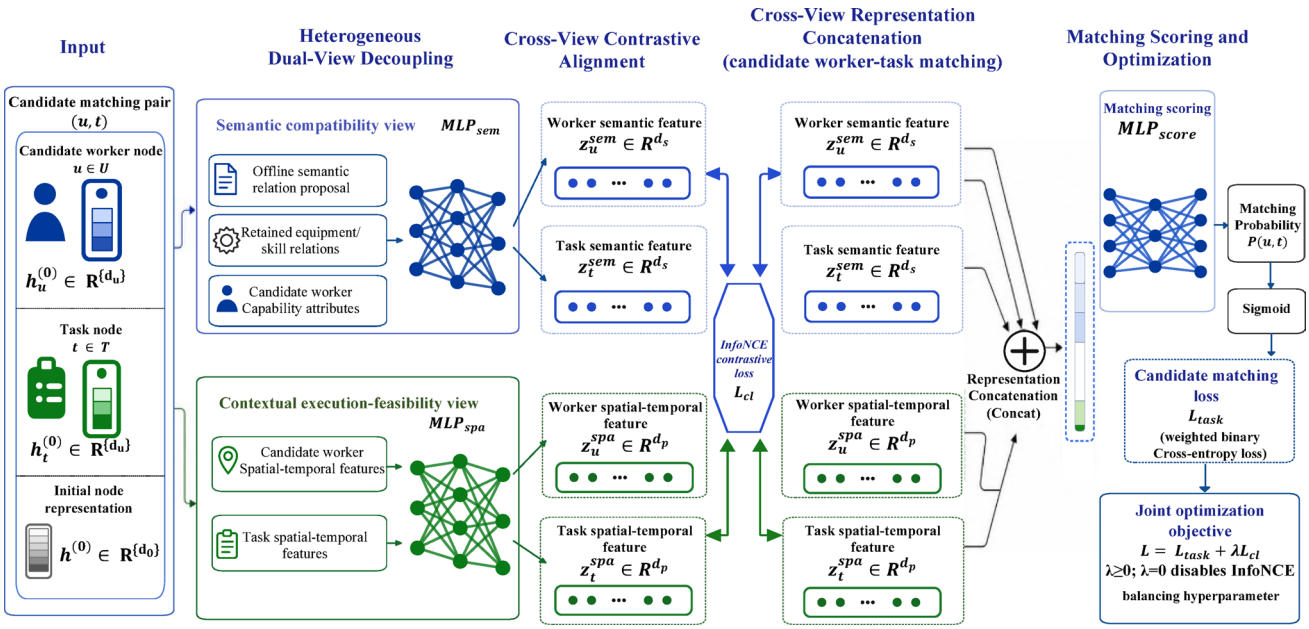


Fig. 2 Architecture of the LLM-DVGCL task assignment algorithm

domain-related semantic information expressed implicitly in task descriptions. For example, two road-maintenance or environmental inspection tasks may require different positioning accuracy, camera equipment, and on-site experience. Therefore, this paper uses an LLM to identify candidate semantic relations from the task-side text and then maps the retained relations into standardized attributes in the graph.

The LLM-based semantic extraction module is configured using separate development data before final evaluation, without fine-tuning the underlying language model. Given task-side text, the semantic front end produces structured candidate semantic relations together with supporting source-text evidence. These outputs are not treated as verified task constraints or used directly for assignment. Instead, candidate relations are checked according to their evidence, output validity, and confidence or review status before being retained for ontology grounding. Low-confidence, unsupported, or malformed outputs can be rejected, abstained from, or directed to review rather than being automatically inserted into the graph. Final assignment is still performed by the graph model combining retained semantic information with contextual information.

To align retained semantic relations with graph nodes, an ontology-grounding process is added after extraction. Nonstandard natural language phrases are mapped to the equipment ontology and skill ontology. For example, different expressions may be mapped to standardized entities. After normalization, task nodes are connected to task–attribute edges, converting task-side text into graph inputs usable by GNNs.

After implicit entity normalization, the algorithm further transforms retained task-side semantic relations and user attributes into inductive semantic representations. The model first applies sum pooling over attribute neighbors to aggregate user equipment and skill features. As defined in Eq. (3), the model first aggregates the equipment and skill embeddings connected to user u to obtain the attribute-based representation h_u^{attr}. It then uses a weighted residual mechanism to fuse this attribute-based representation with a learnable identity embedding:

$$h_u^{(0)} = \alpha h_u^{attr} + (1-\alpha)\,\boldsymbol{id}_u \tag{8}$$

In Eq. (8), $\boldsymbol{id}_u$ denotes the learnable identity embedding of user u, and h_u^{attr} denotes the attribute-based representation of user u. α controls the fusion between attribute aggregation and identity representation. In sparse experimental settings, this weight is set relatively high so that cold-start nodes rely more on semantic attributes.

4.3 Contextual execution feasibility-view construction

Besides semantic constraints, MCS task assignment must account for spatiotemporal execution feasibility and dynamic operational conditions. This section explains how these constraints are organized in the forward computation. The contextual view does not regenerate semantic attributes. Instead, it describes the real-world execution conditions of a candidate assignment pair using the contextual information available before the assignment decision.

In the algorithm, basic feasibility signals such as user location, task location, reachable distance, time window, device status, and availability are used for candidate filtering when available. Candidate filtering produces C_t, the feasible candidate set for task t. This filtering step reduces clearly infeasible candidates before scoring. Unlike inductive node representation, the contextual view constructs relation-level executability features around each candidate user-task pair and can additionally incorporate pre-decision dynamic operational signals. For any candidate pair, the contextual feature is defined as

$$x_{u,t}^{ctx} = \left[\overline{d}_{u,t}, \Delta t_{u,t}, a_{u,t}, m_{u,t}^{equip}, q_u, x_{u,t}^{dyn}\right] \tag{9}$$

The contextual feature vector $x_{u,t}^{ctx}$ combines basic pair-level feasibility features with available dynamic operational information. Specifically, $d_{u,t}$ is the normalized distance between user u and task t. The variable $\Delta t_{u,t}$ measures the matching degree between user availability and the task time window. The variable $a_{u,t}$ indicates whether user u is available during the required time window. The variable $m_{u,t}^{equip}$ represents equipment-related compatibility between user u and task t, when such information is available. The variable q_u denotes a historical service-quality proxy. The vector $\boldsymbol{x}_{u,t}^{dyn}$ contains available pre-decision dynamic operational signals, which may include historical location, regional or AOI affinity, hourly affinity, historical activity level, workload or backlog, recent acceptance behavior, historical completion duration, and cold-start status. These features jointly describe spatial, temporal, equipment, and historical quality conditions. The contextual encoder maps this relation-level feature into a contextual-view representation:

$$z_{u,t}^{ctx} = \mathrm{MLP}_{ctx}\left(x_{u,t}^{ctx}\right) \tag{10}$$

where $z_{u,t}^{ctx}$ denotes the contextual-view representation of candidate pair u, t, and $MLP_{ctx}(\cdot)$ is the contextual encoder. This representation models the execution suitability of user u for task t under the available mobile and operational

context. It complements the semantic-view representation by encoding feasibility factors such as distance, time, device status, availability, and available dynamic operational conditions.

4.4 Cross-view contrastive alignment

The semantic view and the contextual execution-feasibility view encode complementary information. If semantic features and contextual features are directly mixed as a single input vector, low-dimensional continuous context features may interfere with high-dimensional semantic features. To preserve the distinction between the two views while allowing their representations to interact during training, LLM-DVGCL applies cross-view alignment at the candidate-pair level rather than only at the node level. The semantic matching representation is constructed as:

$$z_{u,t}^{sem} = \mathrm{MLP}_{sem}\left([h_u^{sem}; h_t^{sem}; h_u^{sem} \odot h_t^{sem}]\right) \tag{11}$$

In Eq. (11), h_u^{sem} is the semantic-view embedding of user u , and h_t^{sem} is the semantic-view embedding of task t. The operator $\odot$ denotes element-wise multiplication. The projection network $MLP_{sem}(\cdot)$ maps the combined user-task semantic interaction into the pair-level semantic-view representation $z_{u,t}^{sem}$.

As an optional cross-view regularizer, InfoNCE contrastive learning treats the semantic and contextual representations of the same candidate pair as a positive pair. This mechanism does not replace the supervised matching loss; instead, it provides an auxiliary objective that encourages representation alignment between the two views for the same candidate pair. Its empirical contribution is evaluated separately rather than assumed to provide a consistent performance improvement. The cosine similarity and contrastive loss are defined as:

$$\mathrm{sim}\left(z_{u,t}^{sem}, z_{u,t}^{ctx}\right) = \frac{\left(z_{u,t}^{sem}\right)^{\top} z_{u,t}^{ctx}}{\| z_{u,t}^{sem} \|_2 \, \| z_{u,t}^{ctx} \|_2} \tag{12}$$

$$L_{cl} = -\frac{1}{|B|} \sum_{(u,t)\in B} \log \frac{\exp\left(\mathrm{sim}\left(z_{u,t}^{sem}, z_{u,t}^{ctx}\right)/\tau\right)}{\sum_{(u',t')\in B} \exp\left(\mathrm{sim}\left(z_{u,t}^{sem}, z_{u',t'}^{ctx}\right)/\tau\right)} \tag{13}$$

where $sim(\cdot,\cdot)$ is cosine similarity, $\|\cdot\|_2$ denotes the L2 norm, B is the training batch, and τ is the contrastive temperature coefficient. The contrastive loss is used as an optional regularization term that encourages the semantic and contextual representations of the same candidate pair to remain aligned in the representation space. This design follows the common contrastive-learning principle of pulling positive pairs closer while pushing them away from negative samples (Oord et al. 1807; Velickovic et al. 2019; Chen et al. 2020).

4.5 Cross-view representation concatenation and matching score

After encoding the semantic and contextual views, the model concatenates the two representations of a candidate pair to construct the joint matching representation:

$$z_{u,t} = \left[z_{u,t}^{sem}; z_{u,t}^{ctx}\right] \tag{14}$$

In Eq. (14), $z_{u,t}$ denotes the joint representation of candidate pair u, t. It is obtained by concatenating the semantic-view representation $z_{u,t}^{sem}$ and the contextual execution-feasibility representation $z_{u,t}^{ctx}$.

The joint representation is then fed into a multi-layer perceptron, and a Sigmoid function outputs the matching probability between user u and task t:

$$p_{u,t} = \sigma\left(\mathrm{MLP}_{match}\left(z_{u,t}\right)\right) \tag{15}$$

where $p_{u,t}$ is the final matching probability, $\sigma(\cdot)$ is the Sigmoid activation function, and $MLP_{match}(\cdot)$ is the final matching network. The semantic representation $z_{u,t}^{sem}$ captures the compatibility between task-side semantic information and the attributes of user u, whereas the contextual representation $z_{u,t}^{ctx}$ captures the execution suitability of user u for task t under the available mobile and operational context. Joint scoring combines these complementary sources of information rather than relying only on semantic similarity or spatial proximity.

4.6 Joint optimization objective and assignment strategy

Because positive matching labels are often much fewer than negative or sampled alternative pairs, standard binary cross-entropy may be biased toward the majority class. We therefore use WBCE as the supervised matching loss. When enabled, the dual-view contrastive loss is added as an auxiliary alignment regularizer. The final training objective follows Eq. (7), where λ controls the contribution of contrastive regularization and $\lambda = 0$ corresponds to supervised dual-view matching without InfoNCE. After training, for a task whose semantic information has been preprocessed when open text is available, the model constructs a candidate user set from the available pre-decision context, computes matching probabilities for candidate user–task pairs, and outputs the ranked candidates according to the matching scores and applicable feasibility constraints.

Input: task-side record set $X_T = \{x_t \mid t \in T\}$; sensing user set $U$and cold-start user set U_{new}; historical interaction records R; available pre-decision contextual records; pretrained LLM mapping function f_{LLM}; predefined ontology resources, including the equipment ontology O_E and skill ontology O_S; maximum distance threshold d_{max}; training hyperparameters $\alpha, \beta, \lambda, \tau,$ MaxEpoch and Patience

Output: user–task matching probability matrix $\boldsymbol{P}$ and ranked candidate users, including the top-ranked candidate u^*

1. **Stage 1: offline LLM-based implicit semantic extraction and ontology grounding.** For each task $t \in T$ with available open text, use $f_{LLM}(x_t)$ to produce structured candidate semantic relations together with supporting source-text evidence. Check the output validity and the associated acceptance, abstention, or review state. Map only retained relations to the corresponding ontology entities, including equipment or skill entities when applicable, and construct the retained task–attribute edges.
2. Combine user attributes, standardized task constraints, and historical interactions to construct the heterogeneous semantic graph G and its adjacency matrices.
3. **Stage 2: inductive representation initialization.** For each cold-start user $u \in U_{new}$, compute h_u^{attr} from equipment and skill neighbors and obtain $h_u^{(0)}$ through attribute-identity fusion. For each task t, initialize $h_t^{(0)}$ from its requirement attributes. Prepare the available pre-decision contextual features separately for candidate-pair contextual encoding.
4. **Stage 3: dual-view projection and model training.** Set semantic projection head MLP_{sem}, contextual execution feasibility encoder MLP_{ctx} and matching network MLP_{match}. Initialize model parameters and optimizer. When $\lambda > 0$, additionally compute L_{cl} and optimize $L_{total} = L_{task} + \lambda L_{cl}$; when $\lambda = 0$, optimize L_{task} without contrastive regularization. Update the trainable parameters by back-propagation and stop early if the validation loss does not decrease for Patience epochs.
5. **Stage 4: online task assignment.** Given a task $t$whose semantic relations have already been preprocessed when open text is available, obtain the available pre-decision contextual signals and apply the applicable basic feasibility filters to construct C_t. For each $u \in C_t$, construct $x_{u,t}^{ctx}$, compute $p_{u,t}$, rank the candidates, and output the top-ranked candidate u^*. The LLM is not invoked inside this ranking stage.

The coefficient λ controls the optional contrastive regularization term, and setting $\lambda = 0$ disables InfoNCE while retaining the dual-view assignment architecture. The outputs of the offline LLM-based semantic extraction module are treated as fixed preprocessing inputs and are not updated through back-propagation. Only the graph encoders, projection heads, contextual encoder, and matching network are optimized under the assignment objective.

Algorithm 1 LLM-DVGCL-based mobile crowdsensing task assignment

4.7 Complexity analysis

The computational cost of LLM-DVGCL is analyzed from three stages: offline LLM-based implicit semantic extraction and ontology grounding, graph representation learning, and candidate-pair scoring and ranking. Let $|V|$ denote the total number of user, task, equipment, and skill nodes, and let $|E_g|$ denote the number of heterogeneous graph edges. Let d denote the representation dimension, L denote the number of graph propagation layers, $|\Omega|$ denote the number of training candidate pairs, and $|C_t|$ denote the candidate set size after contextual filtering for a single task.

First, LLM-based implicit semantic extraction and ontology grounding are performed as offline preprocessing for tasks with available open text. This stage produces structured candidate semantic relations and supporting source-text evidence, checks output validity and review status, and maps retained relations to ontology entities. Its computational cost depends on the number of text records, the language-model inference cost, and any required validation or retry operations. Although this cost is outside the candidate-ranking loop, it remains part of the end-to-end task-processing cost and is therefore reported separately in the experiments. Second, graph representation learning performs message propagation and attribute aggregation among user, task, equipment, and skill nodes. With L graph propagation layers, the main complexity is $O(L \mid E_g \mid d)$, which is approximately linear in the number of graph edges. Third, candidate-pair training computes semantic and contextual representations and matching scores for the training pairs. For $\mid \Omega \mid$ training pairs, the scoring complexity is approximately $O(\mid \Omega \mid d)$. When in-batch InfoNCE is enabled, pairwise similarity computation adds approximately $O\left(\mid B\mid^2 d\right)$ cost for batch size B.

During candidate ranking, the system applies the available basic feasibility filters and retrieves the pre-decision contextual information required to construct $x_{u,t}^{ctx}$, after which it scores only the users in C_t. The ranking complexity for a single task is approximately $O(\mid C_t \mid d + \mid C_t \mid \log K)$, where K denotes the number of top-ranked candidates returned. When $\mid C_t \mid$ is smaller than the full candidate population, the scoring cost grows approximately linearly with the filtered candidate-set size. Overall, LLM-DVGCL separates the computationally expensive semantic front end from the latency-sensitive candidate-ranking stage. The ranking-stage complexity therefore excludes LLM

inference, whereas the complete pipeline cost includes both offline semantic processing and subsequent graph-based scoring.

5 Experiments and evaluation

5.1 Experimental design and evidence tracks

To evaluate LLM-DVGCL while respecting the evidence available in different public datasets, we organize the experiments into two complementary but non-equivalent tracks. The first track evaluates dynamic task assignment using observed task–courier interactions and pre-decision operational context. The second track audits the offline LLM-based implicit semantic extraction module using independently annotated public task-side text. The legacy Improve Detroit/SeeClickFix construction is retained only as a controlled proxy benchmark and reproducibility audit. Because no dataset used in this study simultaneously contains open task text, observed task–worker interactions, dynamic operational states, and completion evidence, the two experimental tracks are interpreted separately and do not constitute an end-to-end validation of the complete pipeline.

5.1.1 Real-interaction dynamic assignment dataset

For the real-interaction assignment track, we use task–courier interaction records from LaDe-D and conduct cross-city evaluation on Jilin and Yantai. An observed task–courier pair is treated as an implicit positive, whereas deterministically sampled alternative couriers are treated as implicit negatives for ranking evaluation. These implicit negatives indicate that the corresponding alternatives were not observed for the evaluated task; they are not interpreted as evidence that the alternative couriers were physically unable to execute the task. LaDe-D does not contain open task text or explicit equipment and skill requirements and is therefore used to evaluate the dynamic assignment component rather than the LLM-based semantic extraction component.

All historical and dynamic features are computed using information available before the acceptance time of the evaluated task. The available features include historical location, regional or Area of Interest (AOI) affinity, hourly affinity, historical order volume, workload or backlog, acceptance records over recent time windows, historical completion duration, and cold-start status. In this evaluation, the compatibility view is instantiated from historical interaction and regional or temporal affinity information, whereas the contextual execution-feasibility view incorporates the available pre-decision operational signals. No semantic fields or annotations from the separate public-text audit are introduced into the LaDe-D experiments. The complete candidate protocol and time-safety audit are reported in Table S1.

For each evaluated task, alternative couriers are sampled deterministically from couriers with available historical records to construct the candidate-ranking set. The main evaluation uses the resulting fixed candidate sets. To examine whether the results depend only on easily distinguishable alternatives, we additionally conduct a post-test exploratory hard-negative diagnostic. For each task, this diagnostic selects, from the already sampled alternatives, the courier most similar to the observed courier according to eleven standardized pre-decision features. Because this diagnostic was defined after the main test results had been examined, it is reported as exploratory rather than confirmatory. The sampled alternatives remain implicit negatives and are not treated as ground-truth infeasible assignments.

Assignment performance is evaluated using pairwise accuracy, Mean Reciprocal Rank (MRR), Normalized Discounted Cumulative Gain at 5 (NDCG@5), and Hit Rate at 1 (Hit@1). Uncertainty in the main comparison is estimated using task-level bootstrap confidence intervals. Because separate confidence intervals for individual models do not directly quantify the difference between two methods, we additionally report paired performance differences relative to the HGB baseline with corresponding bootstrap confidence intervals. The dual-view model without InfoNCE is treated as the primary assignment variant, while the version with InfoNCE is retained to evaluate the contribution of the optional contrastive regularizer.

5.1.2 Semantic-front-end audit dataset and annotation protocol

For the semantic-front-end audit, we use 1,415 public Helsinki Open311 civic-feedback records written in Finnish and construct a frozen adjudicated benchmark containing 180 records. The benchmark was fixed before the final semantic-front-end evaluation. These records provide public source text for evaluating evidence-bearing semantic candidate relations, but they do not contain observed worker identities, worker–task assignment decisions, or complete task-execution chains. Therefore, the Helsinki benchmark is used only to audit the offline LLM-based implicit semantic extraction module and is not used to evaluate end-to-end MCS task assignment.

Two annotators independently labeled all 180 frozen records without access to the semantic-front-end predictions. The annotation covered sensing modality, application domain, equipment requirement state, and skill requirement state, together with the shortest supporting span available in the Finnish source text. After independent annotation, 91

records had complete agreement, whereas the remaining 89 records were adjudicated to produce the final gold labels. Cohen's kappa is used as the nominal-label agreement coefficient (Cohen 1960). Field-level agreement and gold-label provenance are reported in Table S6.

The exact agreement rates were 0.8722 for modality, 0.7500 for domain, 0.9056 for equipment requirement state, and 0.9167 for skill requirement state. The corresponding pooled Cohen's kappa values were 0.8885, 0.8781, 0.7733, and 0.2500, respectively. The low kappa for skill requirement state is interpreted cautiously because explicit positive skill evidence is rare in the frozen benchmark. Skill-related results are therefore reported as a limited diagnostic rather than strong evidence of semantic-front-end superiority.

Model and prompt selection were conducted only on separate development and calibration data that did not overlap with the 180-item frozen benchmark. The final semantic front end used a representative multilingual instruction-tuned LLM accessed through an API, together with a structured six-field prompt and six few-shot examples selected from calibration data. The inference configuration was frozen with temperature set to 0, thinking disabled, a maximum output length of 200 tokens, and at most one retry for an invalid response. After the model, prompt, examples, and inference settings had been fixed, the final configuration was evaluated once on the adjudicated benchmark. The benchmark gold labels were not used to select or revise the final prompt. This study does not compare alternative LLM backbones or claim that the selected model is optimal; the reported semantic results apply only to the frozen configuration evaluated in this study.

The frozen multilingual LLM configuration is compared with a fixed and transparent Finnish lexical baseline on the same 180 records. Semantic relation quality is evaluated using F1-score for modality and domain. Equipment and skill requirement-state prediction is evaluated using both accuracy and macro-F1 to avoid conclusions based only on majority-class accuracy. We additionally report valid-JSON rate, mean inference latency per item, and prompt and completion token usage. Paired bootstrap confidence intervals quantify multilingual LLM-minus-lexical differences. Fine-grained per-class precision, recall, F1-score, and confusion matrices for equipment and skill requirement states are reported in Table S5 of the Supplementary Material because these fields contain rare positive classes and are operationally sensitive to false acceptance.

Table 2 Input–label leakage audit for proxy-label construction

Variable	Used for proxy label	Used as model input	Leakage risk	Treatment
request_group	Yes	Yes	Medium	Used only as type or semantic input
distance	Yes	Yes	High	Retained as core executability variable
Equipment coverage	Yes	Yes	High	Used as core task equipment constraint
closed_at	Yes	No	High	Used only for proxy label construction
num_hours_to_close	Yes	No/ training statistics[a]	High	Test tasks do not enter the model
q_u/historical quality[b]	Yes	Yes	Medium–high	Aggregated only from the training window in temporal split

[a]Training-window statistics are used only when they are available from historical data and are not used as direct test-time outcome variables. [b]q_u denotes the service quality proxy aggregated from historical records in the training window

5.1.3 Legacy proxy benchmark and reproducibility audit

The legacy controlled benchmark is derived from Improve Detroit/SeeClickFix task-side logs. The original collection contains 717,197 records; after filtering and deduplication, 646,892 usable records remain. Using random seed 42, 12,000 records are selected through stratification by request_group and status. Closed or archived records are grouped by request_group and zip_code, and groups with at least 12 records are retained, producing 221 service-cell proxies. These proxies are rule-derived aggregation units rather than observed workers or agencies. The 12,000-record sample and the minimum-support threshold of 12 are empirical computational and support choices and are not claimed to be optimal.

Proxy labels use task category, spatial distance, equipment- or skill-related coverage, historical quality, and closure information. Consequently, several variables used to construct the labels also appear as model inputs, and the benchmark is partly recoverable through explicit rules. The service-cell profiles are aggregated over the available history and are not time-safe in the same sense as the LaDe-D evaluation. The Detroit benchmark is therefore retained only for controlled diagnosis, sample-size sensitivity, and reproducibility analysis; it is not used as evidence of real worker acceptance or online task-assignment performance.

Table 3 Legacy proxy benchmark parameters and data settings

Parameter	Value
Dataset	Improve Detroit/SeeClickFix
Usable raw task records	646,892
Sampled tasks	12,000
Candidate service units	221
Equipment/skill ontology size	8/6
Graph edges	40,248
Positive/negative samples in 1:5 setting	6,708/33,540
Hidden dimension	64
Optimizer	Adam
Learning rate	0.005
Weight decay	1 × 10^-5
Epochs	50
Dropout rate	0.3
Train/test split	8:2
Maximum physical matching distance	8.0 km
Positive-sample weight β	5.0
Contrastive coefficient λ	0.1
Contrastive temperature τ	0.5

Table 2 makes the proxy benchmark's input–label overlap explicit. Outcome variables such as closed_at and num_hours_to_close are excluded from direct test-time input, but distance, semantic coverage, and historical-quality variables remain related to the rule-derived labels. The benchmark should therefore be interpreted as a leakage-aware controlled diagnostic rather than an independent estimate of real deployment performance.

The legacy experiments retain the original 1:5 and 1:1 positive-to-negative constructions. The main 1:5 setting contains 6,708 proxy-positive and 33,540 proxy-negative pairs. Its semantic-tag noise, attribute reporting errors, location drift, temporary unavailability, and execution-failure variables are simulated perturbations rather than observed operating events. The original settings are retained to support reproducibility of the previously reported controlled analyses (see Table 3).

5.2 Compared methods and implementation settings

For the LaDe-D assignment track, we compare the Heterogeneous Graph Baseline (HGB) with two variants of the proposed dual-view residual model: the primary version without InfoNCE and the version with InfoNCE. All methods use the same fixed candidate sets, temporal splits, and pre-decision information. The one-time city tests are interpreted together with paired task-bootstrap confidence intervals, and the optional InfoNCE mechanism is additionally examined across ten development seeds.

For the Helsinki semantic audit, the frozen multilingual LLM configuration is compared with the fixed Finnish lexical baseline. Both methods receive the same source text and are evaluated against the same adjudicated labels. The lexical baseline provides a low-cost and transparent reference for determining whether the LLM adds value beyond direct term matching.

For the legacy Detroit diagnosis, the original learning-based baselines MAB-DRL, Meta-Transfer, HIN, LightGCL, and LLMRec are retained, together with distance and semantic rules and tabular models where applicable. These comparisons characterize how strongly the proxy construction can be recovered from explicit rules and tabular variables; they are not treated as the primary evidence for real assignment performance. Neural components are optimized with Adam (Kingma and Ba 2015), and the GBDT diagnostic follows the gradient-boosting framework (Friedman 2001).

5.3 Experimental results and analysis

5.3.1 Real-interaction dynamic assignment results

Across both cities, the dual-view residual model without InfoNCE improves pairwise ranking relative to HGB, and the paired task-bootstrap confidence intervals for the

Table 4 Real-interaction dynamic assignment results on LaDe-D

City	Test tasks	Method	Pairwise accuracy (95% CI)	MRR	NDCG@5	Hit@1	Paired Δ vs HGB (95% CI)
Jilin	6,283	HGB	0.9876 [0.9865, 0.9886]	0.9498	0.9628	0.9045	Reference
Jilin	6,283	Dual-view residual (without InfoNCE)	0.9888 [0.9878, 0.9897]	0.9543	0.9661	0.9125	+0.0012 [+0.0007, +0.0018]
Jilin	6,283	Dual-view residual (with InfoNCE)	0.9882 [0.9872, 0.9892]	0.9517	0.9642	0.9075	+0.0007 [+0.0003, +0.0010]
Yantai	41,287	HGB	0.9713 [0.9705, 0.9721]	0.9105	0.9311	0.8466	Reference
Yantai	41,287	Dual-view residual (without InfoNCE)	0.9768 [0.9761, 0.9776]	0.9264	0.9434	0.8724	+0.0055 [+0.0050, +0.0059]
Yantai	41,287	Dual-view residual (with InfoNCE)	0.9770 [0.9763, 0.9777]	0.9267	0.9438	0.8726	+0.0057 [+0.0052, +0.0061]

Observed task–courier pairs are implicit positives; sampled alternatives are implicit negatives and are not asserted to be infeasible

Square brackets report 95% task–bootstrap confidence intervals from 2,000 replicates

Table 5 Frozen semantic-front-end audit on Helsinki Open311

Method	Modality F1	Domain F1	Equipment accuracy/macro-F1	Skill accuracy/macro-F1	Valid JSON	Mean latency per item	Prompt/completion tokens
Finnish lexical baseline	0.2000	0.5709	0.7944/0.2951	0.9778/0.4944	100.0%	0.00 s	0.0/0.0
Multilingual LLM	0.6988	0.8123	0.8000/0.2489	0.9722/0.3286	99.4%	7.98 s	1341.8/99.8

Paired−bootstrap differences (multilingual LLM minus lexical baseline) are +0.4988 [95% CI: 0.4087, 0.5882] for modality F1, +0.2414 [0.1817, 0.3036] for domain F1, +0.0056 [−0.0111, 0.0278] for equipment accuracy, and −0.0056 [−0.0167, 0.0000] for skill accuracy (5,000 replicates)

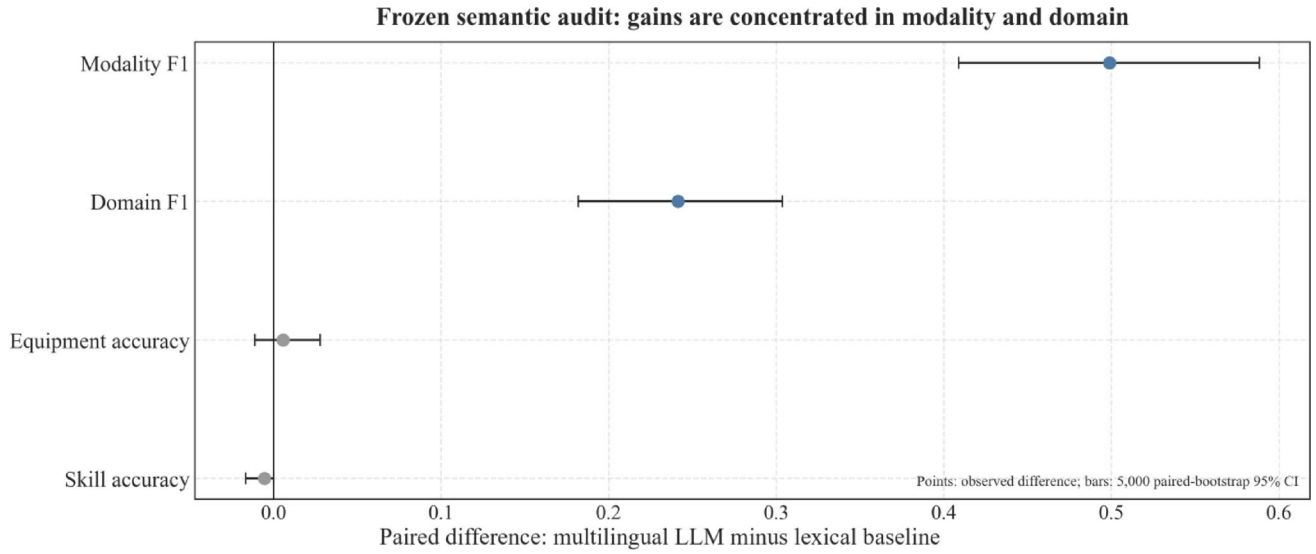


Fig. 3 Paired semantic-front-end effects of multilingual LLM relative to the lexical baseline

primary comparison exclude zero. The relative order of the with- and without-InfoNCE variants differs across cities and the differences are small. The real-interaction evidence therefore supports the dynamic dual-view assignment architecture, but it does not establish an end-to-end contribution from the LLM semantic front end because LaDe-D contains no open task text (see Table 4).

The post-test hard-negative analysis is reported only as an exploratory diagnostic in Table S2 of the Supplementary Material. Its direction is consistent with the main comparison, but the analysis was specified after the primary test results had been examined and is not treated as confirmatory evidence. Candidate-pool composition is reported in Table S3, and the ten-seed InfoNCE mechanism study is reported in Table S4 and Fig. S1.

5.3.2 Frozen semantic-front-end audit results

The frozen audit shows that the multilingual LLM provides a clear advantage over the lexical baseline for sensing modality and application-domain relations. The same conclusion does not extend to equipment and skill requirement states: their accuracy and macro-F1 results do not establish a stable multilingual LLM advantage, and the rare supported classes have low recall. These fields therefore require conservative evidence checks, abstention, or manual review rather than automatic insertion into the graph (see Table 5 and Fig.3).

Table 6 Top-K candidate recommendation performance under the 1:5 cold-start setting (%)

Method	P@10[a]	R@10	N@10	H@10	MRR	R@5
MAB-DRL	15.80	11.85	17.78	68.08	38.20	6.85
Meta-Transfer	15.91	12.03	19.15	68.11	41.44	6.76
HIN	29.04	21.66	32.66	82.08	59.28	11.46
LightGCL	15.44	11.74	17.57	63.84	37.14	6.73
LLMRec	20.96	15.50	22.47	59.83	35.99	9.99
LLM-DVGCL	45.09	34.91	47.28	95.22	68.14	15.90

[a]P@K, R@K, N@K, H@K, and MRR denote Precision@K, Recall@K, NDCG@K, Hit@K, and mean reciprocal rank, respectively

Table 7 Rule-based analysis of proxy candidate labels (%)

Category	Rule	NDCG@10[a]	MRR	Random split F1	Chronological split F1
Rule-based diagnostic	Distance rule	11.45	31.57	76.28	75.76
Rule-based diagnostic	Semantic rule	61.97	76.14	65.64	73.40
Rule-based diagnostic	Distance+semantic rule	77.08	91.88	83.87	84.23

[a]NDCG@10 and MRR are ranking metrics; F1 values are computed under the corresponding random or chronological split

Table 8 Warm-start assignment results under the 1:5 positive-to-negative ratio (%)

Method	Accuracy	Precision	Recall	F1-score
MAB-DRL	73.78	32.11	52.71	39.91
Meta-Transfer	72.65	33.14	64.44	43.77
HIN	77.44	40.08	73.83	51.96
LightGCL	73.81	31.01	47.74	37.60
LLMRec	66.06	30.72	83.98	44.99
LLM-DVGCL	82.66	48.65	89.62	63.07

Table 9 Cold-start binary matching results under the 1:5 positive-to-negative ratio (%)

Method	Accuracy	Precision	Recall	F1-score
MAB-DRL	66.23	11.34	24.22	15.45
Meta-Transfer	67.71	10.22	19.72	13.46
HIN	77.83	20.26	25.23	22.47
LightGCL	72.07	5.98	8.11	6.88
LLMRec	70.47	24.58	63.76	35.48
LLM-DVGCL	82.70	40.58	77.18	53.19

The multilingual LLM also introduces model-inference latency and token cost, whereas the lexical baseline is deterministic and inexpensive. The semantic front end is therefore interpreted as a selective and auditable enhancement layer, not as a universal replacement for lexical rules or human review. Detailed cost, failure, and case-level audit results are reported in Table S7.

5.3.3 Legacy proxy Top-K and rule diagnostics

The following Top-K results are retained to reproduce the original Detroit service-cell benchmark. They rank rule-derived proxy candidates rather than observed workers and must be interpreted within the proxy-label construction. Precision@K, Recall@K, NDCG@K, MRR, and Hit@K follow the definitions used in the original analysis (Järvelin and Kekäläinen 2002) (see Table 6).

Within this legacy benchmark, LLM-DVGCL obtains the strongest ranking values among the retained learning-based baselines. This result shows that the model can fit the structured proxy task, but it does not establish superiority in real worker assignment because the candidate identities and labels are rule-derived.

The high scores of rule-based diagnostics on proxy labels show that the proxy label construction is strongly guided by interpretable conditions such as category, distance, semantic coverage, and reputation (see Table 7). These rules are therefore used to reveal the boundary of the proxy benchmark, not as performance upper bounds for real online task assignment.

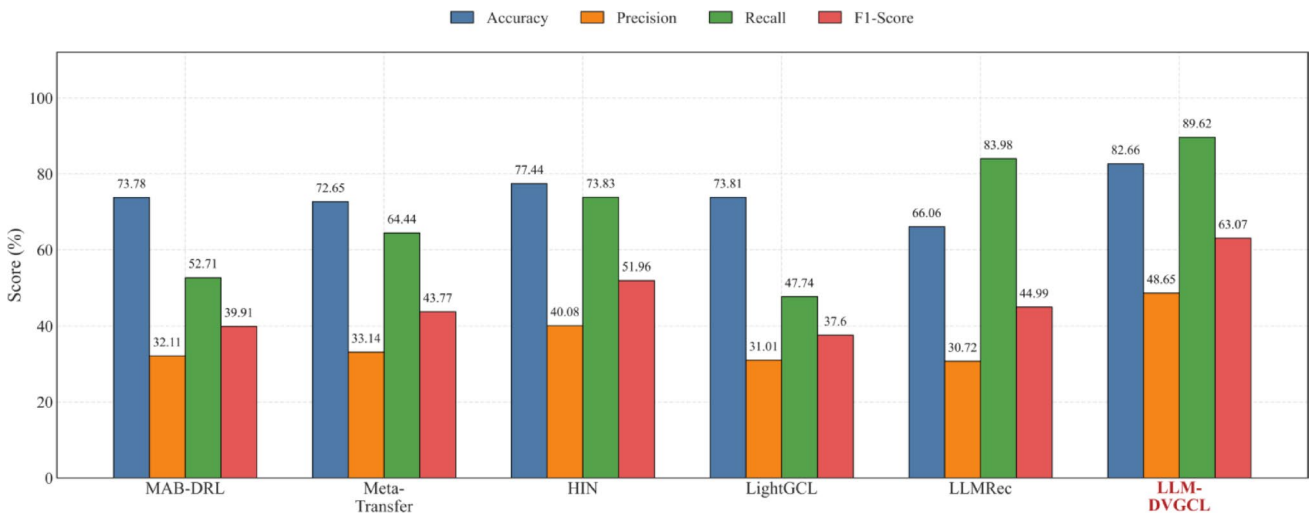


Fig. 4 Warm-start binary matching under a 1:5 positive-to-negative ratio

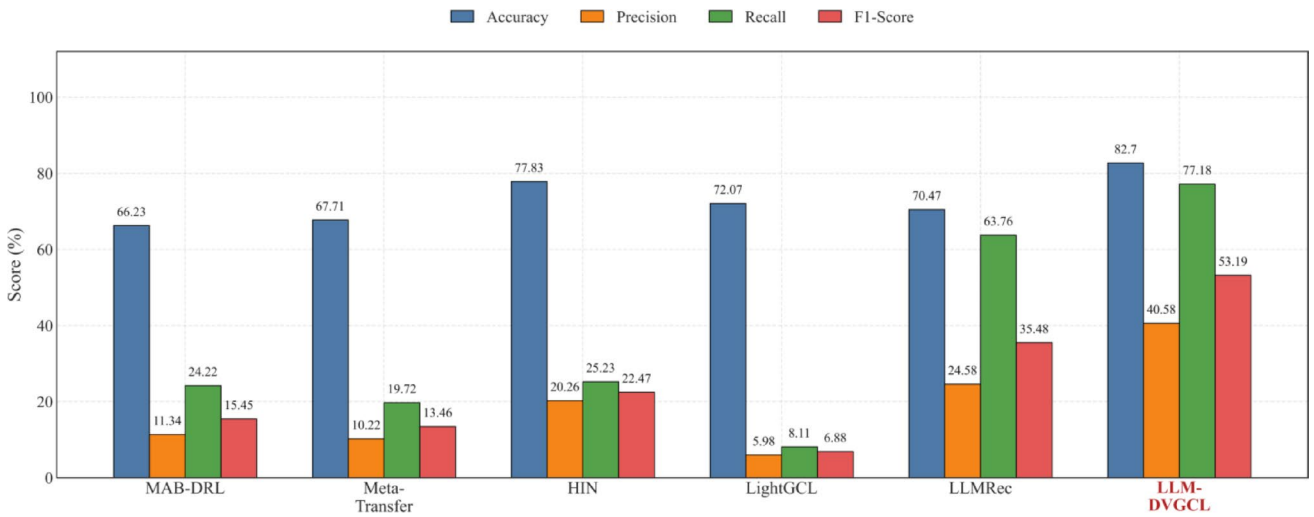


Fig. 5 Cold-start binary matching under a 1:5 positive-to-negative ratio

Table 10 Chronological 8:2 split results (%)

Category	Method	Accuracy	Precision	Recall	F1-score
Learning-based method	HIN	79.09	46.67	73.43	57.07
Learning-based method	LightGCL	74.41	37.74	54.10	44.46
Learning-based method	LLMRec	64.88	33.73	88.69	48.88
Learning-based method	LLM-DVGCL	81.67	50.90	89.66	64.93
Rule-based diagnostic	Distance+semantic rule	93.91	82.56	85.97	84.23

5.3.4 Legacy binary matching under the 1:5 positive-to-negative ratio

The original 1:5 proxy setting is retained as a controlled warm-start and cold-start diagnostic. The class imbalance, semantic noise, and availability variables in this section are generated by the legacy benchmark and should not be interpreted as observed platform behavior.

Table 8 and Fig. 4 show the relative behavior of the retained learning-based methods within the warm-start proxy setting. LLM-DVGCL provides the most balanced precision–recall result in this constructed benchmark, while LLMRec's higher recall is accompanied by lower precision. These findings remain conditional on the proxy labels.

Table 9 and Fig. 5 show that interaction-dependent baselines decline under the constructed cold-start condition, whereas attribute-based models retain more signal. LLM-DVGCL obtains the highest F1-score among the retained learning-based methods in this proxy setting; the result should be read as a controlled cold-start diagnostic rather than evidence from real new workers.

Table 11 Performance comparison on the semantic–contextual conflict subset (%)

Category	Method	Accuracy	Precision	Recall	F1-score
Tabular baseline	Logistic Regression	78.42	46.35	50.20	48.20
Tabular baseline	MLP	79.15	48.10	52.25	50.10
Tabular baseline	GBDT	80.36	50.88	54.00	52.40
Rule-based diagnostic	Distance+semantic rule	76.90	43.75	47.60	45.60
Learning-based method	LLM-DVGCL	84.59	43.14	66.07	52.19

Table 12 F1-score on rule-conflict subsets (%)

Subset	Distance+semantic rule	GBDT	LLM-DVGCL
Nearby semantic conflict	48.60	52.40	64.85
Semantic match but long distance	45.35	50.72	62.18
Same-type conflict	51.80	55.64	66.27
Long-tail tasks	46.92	53.31	63.40

Because accuracy is dominated by proxy-negative pairs in the 1:5 setting, precision, recall, and F1-score are more informative. The reported values quantify discrimination within the rule-derived benchmark and do not convert the sampled negatives into verified infeasible assignments.

To avoid potential temporal leakage caused by random 8:2 split, this paper further conducts a chronological 8:2 split. Tasks are sorted by created_at; the first 80% are used as the training window and the last 20% as the future test

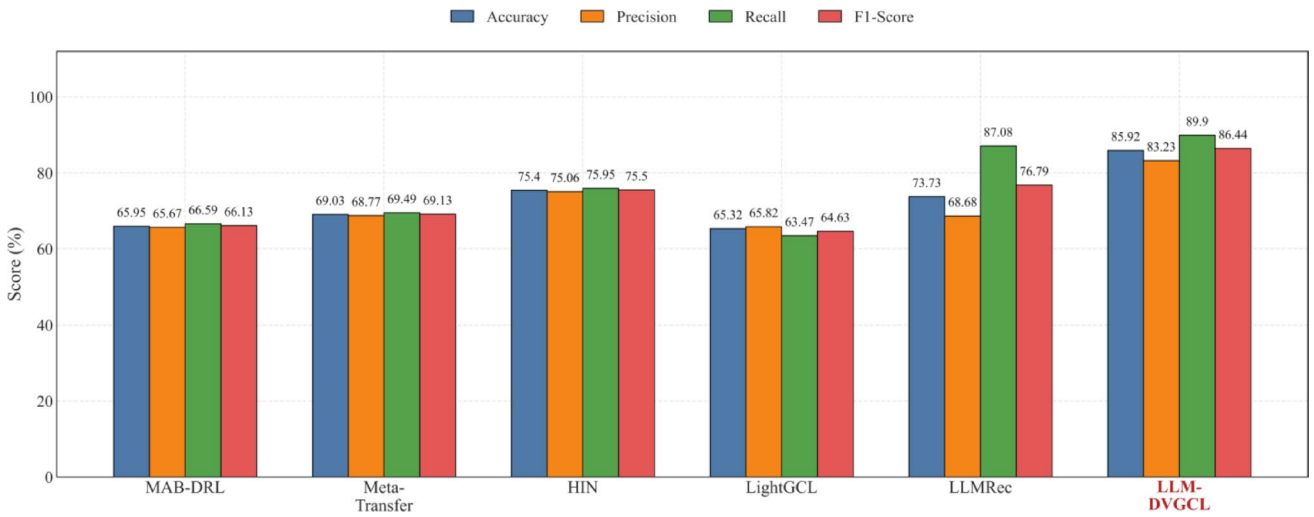


Fig. 6 Warm-start binary matching under a 1:1 positive-to-negative ratio

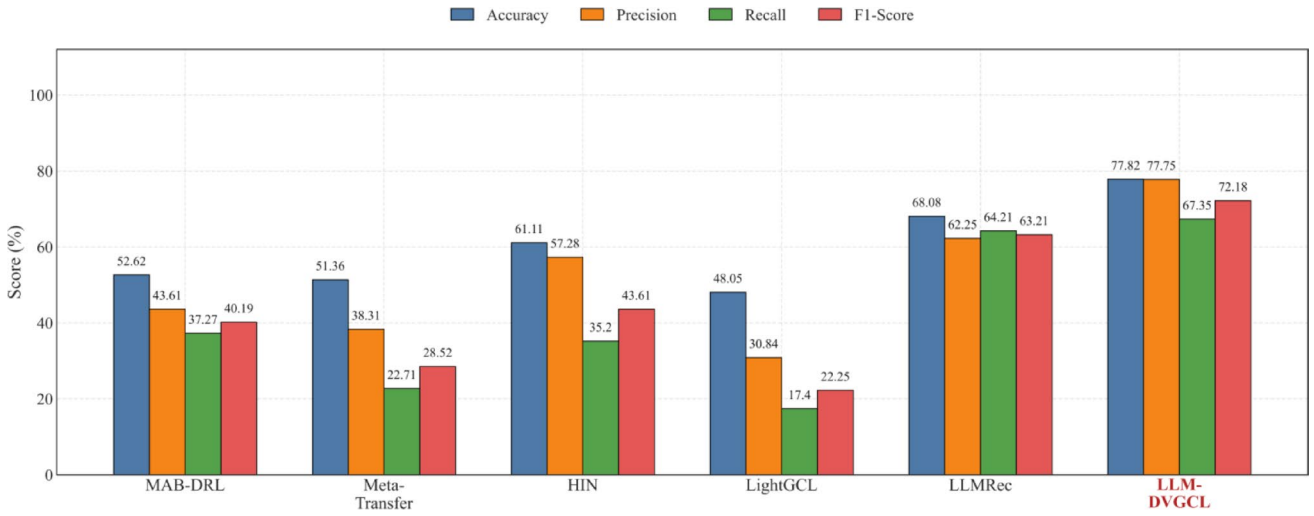


Fig. 7 Cold-start binary matching under a 1:1 positive-to-negative ratio

Table 13 Warm-start binary matching results under the 1:1 positive-to-negative ratio (%)

Method	Accuracy	Precision	Recall	F1-score
MAB-DRL	65.95	65.67	66.59	66.13
Meta-Transfer	69.03	68.77	69.49	69.13
HIN	75.40	75.06	75.95	75.50
LightGCL	65.32	65.82	63.47	64.63
LLMRec	73.73	68.68	87.08	76.79
LLM-DVGCL	85.92	83.23	89.90	86.44

Table 14 Cold-start binary matching results under the 1:1 positive-to-negative ratio (%)

Method	Accuracy	Precision	Recall	F1-score
MAB-DRL	52.62	43.61	37.27	40.19
Meta-Transfer	51.36	38.31	22.71	28.52
HIN	61.11	57.28	35.20	43.61
LightGCL	48.05	30.84	17.40	22.25
LLMRec	68.08	62.25	64.21	63.21
LLM-DVGCL	77.82	77.75	67.35	72.18

window. Candidate service unit profiles are induced only from the training window. This setting simulates future task matching based only on historical logs.

Table 10 shows that LLM-DVGCL has the highest F1-score among the retained learning-based methods under the chronological split. However, the distance+semantic rule remains higher because the proxy labels are constructed from overlapping rules. The chronological split reduces direct temporal mixing but does not eliminate the benchmark's input–label dependence or full-history proxy construction.

To further evaluate discrimination in less rule-dominated candidate matching, this paper constructs a semantic-contextual conflict subset from the test set. It contains four types of candidate pairs: nearby candidates with mismatched equipment or skills, semantically matched candidates with unsatisfied distance or time constraints, candidates with the same request_type but different attribute coverage and historical quality, and long-tail tasks with low-frequency request_type. This subset diagnoses generalization under explicit rule conflicts and does not replace the full proxy benchmark.

Table 11 is retained as a rule-conflict diagnostic. The tabular and rule baselines decline on this selected subset, while LLM-DVGCL maintains higher recall. Because the subset is derived from the same proxy construction, it provides controlled stress evidence rather than an independent real-interaction test.

Table 12 shows the same directional pattern across the four legacy conflict subsets. The result supports the usefulness of separating compatibility and context within the constructed benchmark, but it is not used as the primary proof of deployment generalization.

5.3.5 Legacy binary matching under the 1:1 positive-to-negative ratio

The 1:1 proxy setting is retained to show how the original models behave under a balanced constructed class distribution. It is a sensitivity analysis of the legacy benchmark rather than a separate real-world evaluation (see Fig. 6, Fig. 7).

Table 15 Ablation results from a single run (%)

Method	Accuracy	Precision	Recall	F1-score
LLM-DVGCL	83.07	41.11	76.18	53.40
w/o InfoNCE	83.86	42.32	73.67	53.76
w/o Dual-View	83.19	41.16	74.57	53.04
w/o LLM	75.74	6.20	6.41	6.30

Table 16 Multi-seed stability results for binary matching (%)

Method/Setting	F1 mean ± std	Recall mean ± std	Precision mean ± std
LLM-DVGCL, $\lambda = 0.1$	59.88 ± 3.48	83.35 ± 7.79	46.86 ± 2.86
w/o InfoNCE, $\lambda = 0$	59.19 ± 4.08	81.41 ± 6.86	46.63 ± 3.69
LLM-DVGCL, $\lambda = 0.2$[a]	60.57 ± 3.08	85.07 ± 4.14	47.12 ± 3.32

[a]This row is a sensitivity setting with stronger contrastive regularization; the default setting in the main experiments is $\lambda = 0.1$

Table 17 Fine-grained diagnostic analysis of the dual-view mechanism (%)

Variant	Accuracy	Precision	Recall	F1-score
Semantic view only	83.72	42.18	67.40	51.89
Contextual view only	83.55	41.92	66.85	51.55
Direct concatenation	84.08	42.76	69.10	52.82
Dual-view without InfoNCE	84.17	42.84	72.77	53.93
Dual-view + InfoNCE	85.36	45.20	74.85	56.35

Table 13 and Table 14 show that LLM-DVGCL remains strongest among the retained learning-based baselines in the balanced proxy setting. The difference between warm-start and cold-start results is consistent with the role of attribute initialization, but the absolute values remain tied to the synthetic positive-to-negative construction.

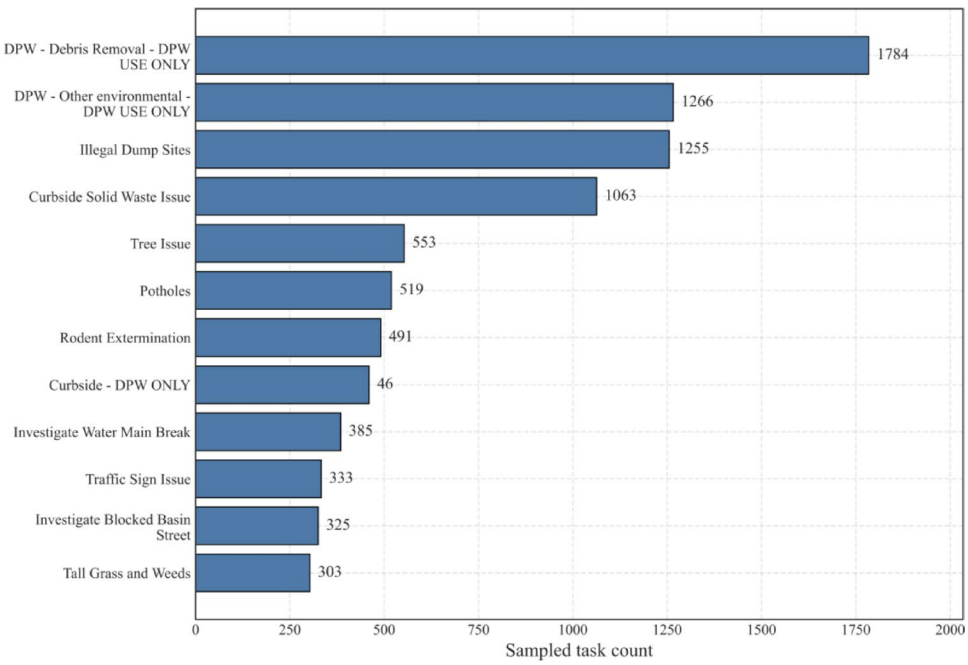


Fig. 8 Distribution of major task types in task-side logs

Table 18 Perturbation configurations

Level	Miss SEM.[a]	False SEM	Miss attr	False attr	Loc. drift	Unavail	Rep. drift	Fail
Clean	0%	0%	0%	0%	0 km	0%	0%	0%
Light	5%	3%	4%	2%	0.25 km	6%	4%	2%
Med	10%	6%	8%	5%	0.45 km	12%	8%	4%
Heavy	15%	10%	12%	8%	0.75 km	18%	12%	6%

[a]SEM. = semantic; attr. = attribute; Loc. = location; Rep. = reputation; Med. = Medium

Percentage columns denote perturbation rates; location drift is measured in kilometers

Table 19 LLM-DVGCL performance under different perturbation strengths (%)

Strength	Accuracy	Precision	Recall	F1-score	Latency (ms)	Description
Clean	87.63	51.01	98.34	67.18	18.0	No perturbation
Light	85.14	47.02	88.44	61.40	17.7	Light perturbation
Medium	83.45	41.20	70.07	51.89	14.8	Medium perturbation
Heavy	81.83	39.67	85.21	54.14	14.0	Heavy perturbation

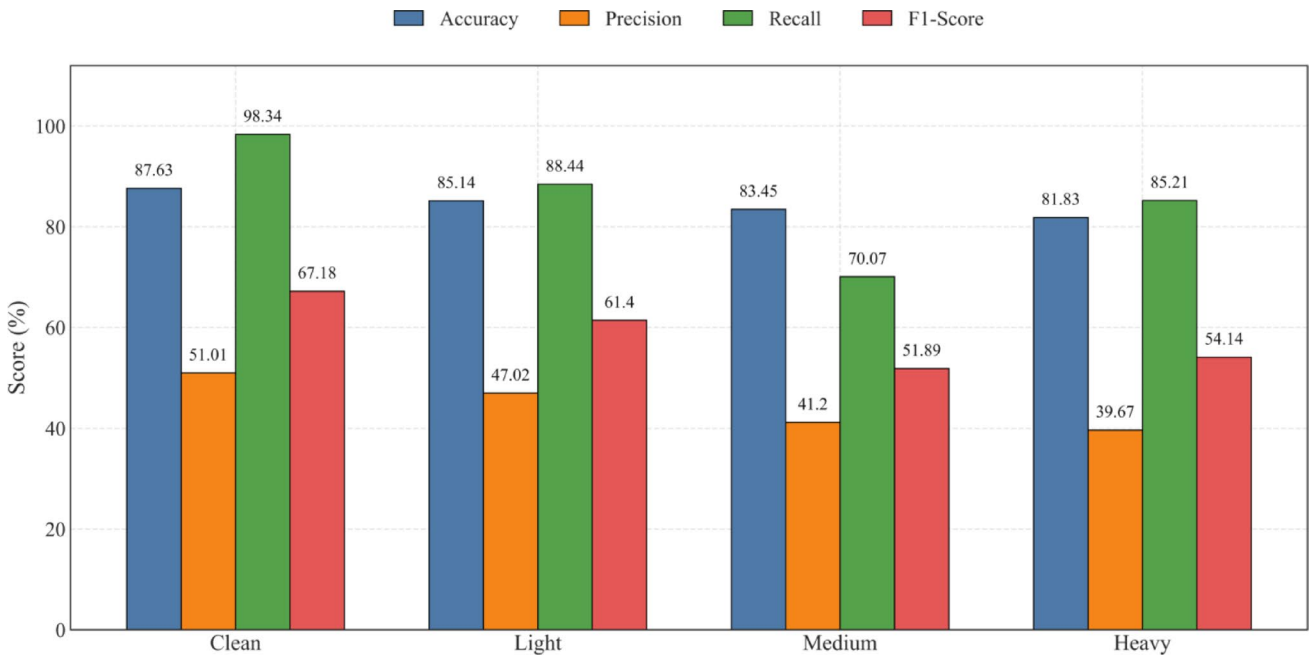


Fig. 9 Performance under different perturbation levels

5.3.6 Legacy ablation diagnostics

The original 1:5 cold-start ablations are retained to document how the legacy proxy benchmark responds to semantic edges, view organization, and InfoNCE. They are treated as mechanism diagnostics and are not used to infer the accuracy of the independently audited Helsinki semantic front end.

Table 15 shows that removing the legacy semantic inputs sharply reduces proxy-benchmark F1, whereas removing InfoNCE does not reduce the single-run F1. The large w/o-LLM difference reflects dependence on the benchmark's constructed semantic variables and must not be interpreted as independent validation of LLM extraction quality.

Table 16 reports the original five-seed sensitivity analysis and does not show a statistically significant difference between the default InfoNCE setting and the no-InfoNCE variant. The revised ten-seed candidate-level mechanism study, reported in the Supplementary Material, likewise does not support a stable additional gain. InfoNCE is therefore retained only as an optional regularizer. The seed-level results and paired plot are provided in Table S4 and Fig. S1.

Table 17 shows that separating the two views is more reliable than either single view in this legacy diagnostic. Although the InfoNCE row is higher in this particular run, the broader evidence does not support describing contrastive regularization as a consistently beneficial or stability-improving component.

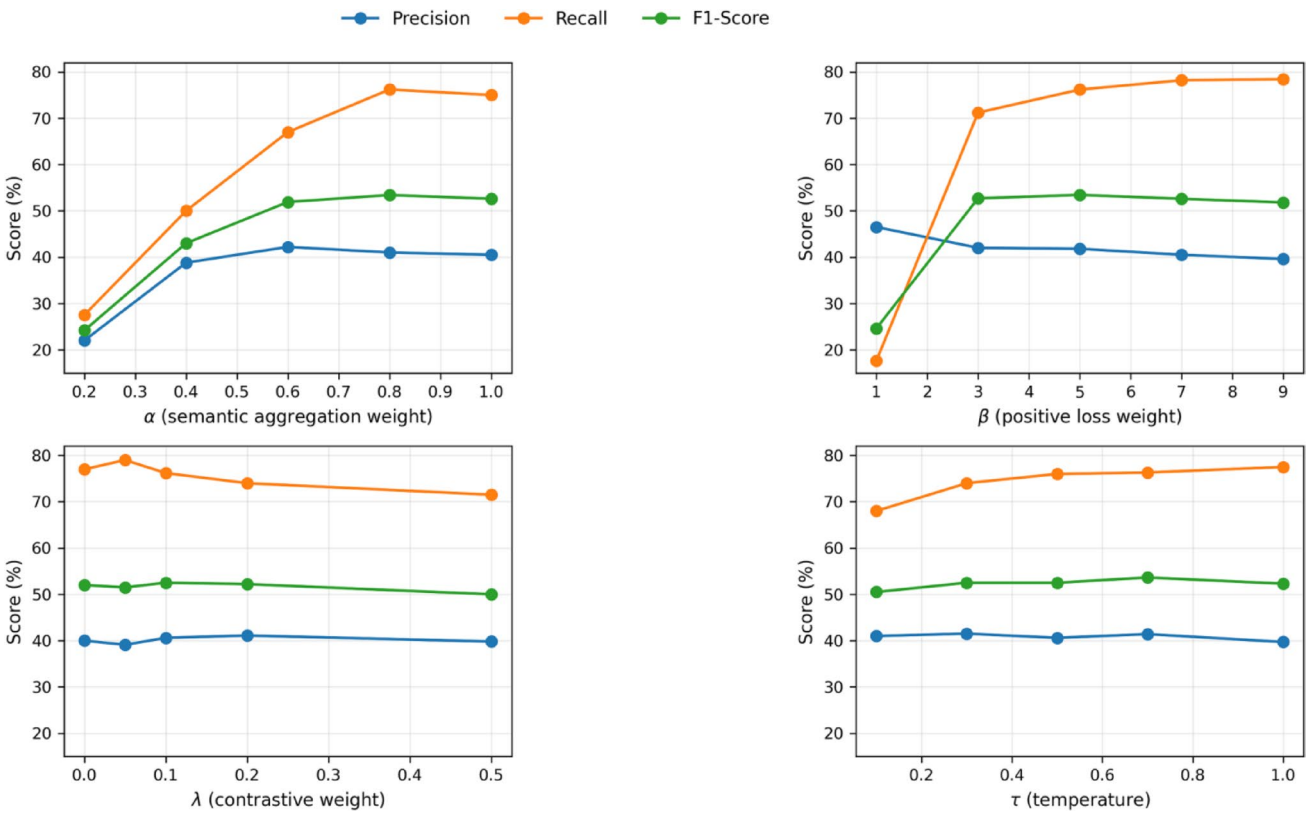


Fig. 10 Hyperparameter sensitivity analysis

5.3.7 Legacy task-type distribution

To examine the composition of task-side logs, we count the major task types in the Improve Detroit/SeeClickFix sample, as shown in Fig. 8.

Figure 8 shows a long-tailed distribution of request types in the Detroit sample. This supports retaining heterogeneous task categories in the controlled benchmark, but the distribution alone does not verify any particular equipment or skill requirement inferred from text.

5.3.8 Controlled semantic-edge and contextual perturbation sensitivity

To examine structural sensitivity within the legacy benchmark, we retain the original controlled perturbations of semantic tags, candidate attributes, location, availability, reputation, and execution-failure indicators. These perturbations are designed stress levels; they are not calibrated to the observed error distribution of multilingual LLM and must not be presented as a realistic end-to-end LLM error simulation. A separate controlled semantic-edge sensitivity analysis is reported in Table S8.

Table 18 and Table 19 and Fig. 9 show that proxy-benchmark precision and F1 generally decline as controlled perturbations increase, although recall is non-monotonic at the heaviest level. The result demonstrates structural sensitivity to corrupted inputs; it does not estimate the operational failure rate of the audited semantic front end.

Table 20 Comparison of model size and inference latency

Method	Parameters (K)	Inference latency (ms)	F1-score (%)	F1/latency[a]
MAB-DRL	790.47	4.2	15.45	3.68
Meta-Transfer	790.59	5.0	13.46	2.69
HIN	799.55	8.9	22.47	2.52
LightGCL	794.62	3.0	6.88	2.29
LLMRec	49.92	412.9	35.48	0.09
LLM-DVGCL	824.38	24.5	53.19	2.17

[a]F1/latency is computed as F1-score divided by inference latency

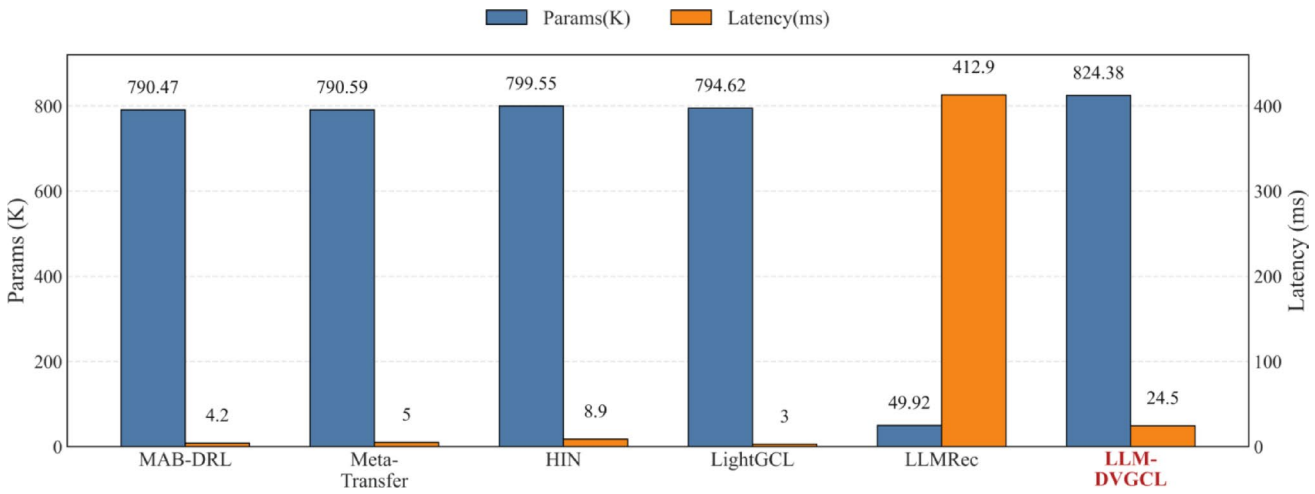


Fig. 11 Comparison of model size and inference latency

Table 21 Complementary semantic and dynamic-assignment cases

Panel A. Helsinki semantic-front-end cases

Public source text (shortened Finnish)	Gold	Lexical	Multilingual LLM relation + evidence span	Gate decision
H001: Asemapäällikönkatu 10 edustalla rajattu jalkakäytävä pois käytöstä nauhalla. Ei löydy kartalta…	Modality: location_or_map Domain: traffic_road | infrastructure_maintenance	Modality: none_explicit Domain: traffic_road | public_facility_service	location_or_map; traffic_road | infrastructure_maintenance Evidence: "Asemapäällikönkatu 10… Ei löydy kartalta"	Accept modality/domain; abstain on requirements
H005: Vid Nordsjö lkågstadieskola finns en skylt med texten "Kiinteistölle kulku sallittu"…	Modality: location_or_map Domain: traffic_road | public_facility_service Equipment: supported	Modality/domain: none_explicit Equipment: insufficient_evidence	location_or_map; public_facility_service Evidence: "Vid Nordsjö lkågstadieskola finns en skylt" Equipment omitted	Human review: equipment mismatch

Panel B. LaDe-D dynamic-assignment case (task_index = 25,141)

Candidate role	Distance	History	Backlog	Recent accepts	HGB score	Dual-view score
Observed courier	2.522 km	1788	6	6/118 (1 d/7 d)	-1.4295	-0.9715
Alternative favored by HGB	1.603 km	2578	13	19/188 (1 d/7 d)	-1.3614	-1.3976

5.3.9 Hyperparameter sensitivity analysis

This legacy sensitivity analysis examines the attribute-fusion weight α, positive-sample weight β, contrastive coefficient λ, and InfoNCE temperature τ in the 1:5 cold-start proxy setting. The analysis is retained for reproducibility, while conclusions about InfoNCE are governed by the revised multi-seed evidence.

Figure 10 shows that α has a clear effect on cold-start representation. When $\alpha = 0.2$, the model depends more on ID representation, and F1-score is only 24.17%. As α increases, semantic attributes become more important, and F1-score reaches 53.42% when $\alpha = 0.8$. Increasing α to 1.0 slightly reduces performance, indicating that relying entirely on attribute aggregation weakens the contribution of learnable representations.

The λ curve in Fig. 10 describes one legacy diagnostic run and should not be interpreted as evidence that moderate contrastive regularization improves stability. Across the revised city-level and ten-seed analyses, the incremental effect of InfoNCE is small and inconsistent. Accordingly, λ=0 remains a valid primary configuration and λ>0 represents an optional regularized variant.

The β results retain their original interpretation: insufficient positive weighting produces low recall, whereas excessive weighting reduces precision. The α results support combining attribute aggregation with learnable identity information rather than relying exclusively on either source. The temperature analysis is relevant only when InfoNCE is enabled.

5.3.10 Ranking-stage model size and inference efficiency

Table 20 and Fig. 11 retain the original comparison of trainable parameters and forward latency for the legacy candidate-ranking models. The reported latency measures the graph-based ranking stage on the test set; it does not include offline LLM inference, ontology grounding, validation, retries, or human review.

Within the ranking-stage comparison, LLM-DVGCL remains substantially faster than the online LLMRec configuration while using more parameters than several lightweight baselines. This result supports separating the semantic front end from latency-sensitive ranking, but it is not a complete end-to-end cost comparison. The Helsinki audit separately reports LLM latency, valid-output rate, and token usage.

5.3.11 Evidence-bearing case analysis

The Helsinki panel includes both a successful domain-relation case and a missed supported-equipment case, so the audit does not present only favorable outputs. The LaDe-D panel illustrates how pre-decision history and operational context change the ranking relative to HGB. The two panels come from different datasets and do not represent a single joined task-to-courier record (see Table 21).

5.4 Discussion and limitations

The revised evidence supports two bounded conclusions. First, the dual-view residual architecture improves pairwise ranking over HGB in two real-interaction city datasets when pre-decision operational context is available. Second, the offline multilingual LLM front end adds clear value for modality and domain relations relative to a transparent lexical baseline. These findings are complementary, but neither dataset independently validates the complete LLM-to-assignment pipeline.

Several limitations remain. LaDe-D provides observed task–courier interactions but no open task text, while Helsinki provides public multilingual text but no observed worker identities, assignment decisions, or completion chain. Sampled alternative couriers are implicit negatives and cannot be interpreted as physically infeasible or unwilling candidates. Road-network travel time, complete concurrent-task conflicts, and all real-time device states are also unavailable in the current evaluation.

The semantic audit shows that equipment and skill relations contain rare positive classes and are not reliably improved by the LLM. These fields therefore require conservative gates, abstention, or manual review. The multilingual LLM also introduces non-negligible latency and token cost. In addition, the ten-seed study does not show a stable extra benefit from InfoNCE, so contrastive learning should be viewed as an optional regularizer rather than the primary explanation for the assignment gains.

Finally, the Detroit construction remains a legacy proxy benchmark whose labels overlap with several model inputs, whose service cells are rule-derived rather than observed workers, and whose full-history aggregation is not time-safe. Its 12,000-record sample and minimum-support threshold of 12 are empirical choices rather than optimal values. The retained legacy tables and figures are provided for reproducibility and controlled diagnosis, not as the central evidence for real deployment performance. The fixed-seed reproducibility record and post hoc sample-size sensitivity are reported in Table S9.

6 Conclusion

We proposed LLM-DVGCL, an LLM-enhanced dual-view graph learning framework for context-aware task assignment in mobile crowdsensing. The framework uses an LLM as an offline, evidence-bearing semantic-relation proposer, grounds retained relations in a predefined ontology, initializes cold-start participants through attribute aggregation, and separates semantic compatibility from contextual execution feasibility. Cross-view InfoNCE is retained as an optional regularizer rather than being assumed to provide a stable gain.

The revised experiments provide two complementary forms of evidence. On LaDe-D, the dual-view residual model improves pairwise ranking over HGB in Jilin and Yantai using observed task–courier interactions and pre-decision operational context. On the independently adjudicated Helsinki benchmark, multilingual LLM improves modality and domain relation extraction over a lexical baseline, but it does not show stable advantages for equipment or skill requirement states and incurs additional latency. These results support selective LLM-assisted semantic enhancement and dynamic dual-view assignment while defining their current limitations.

Future work will construct end-to-end datasets that jointly contain open task text, verified worker attributes, dynamic user states, assignment decisions, and completion evidence. We will also investigate road-network-aware feasibility, concurrent-task conflicts, calibrated semantic confidence, multilingual transfer, privacy-preserving learning, fairness, and human-in-the-loop review for high-risk sensing constraints.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s42486-026-00272-w.

Author contributions Yujie Yan contributed to conceptualization, methodology, software implementation, data processing, experiments, formal analysis, visualization, and writing of the original draft. Haifeng Jiang contributed to conceptualization, methodology, supervision, project administration, validation, and Writing – review & editing. Chaogang Tang contributed to methodology, supervision, and manuscript review. Shuo Xiao contributed to data curation, experimental validation, and visualization. Huaming Wu contributed to academic guidance, high-level methodological discussion, and manuscript review. Ruidong Li contributed to academic guidance, high-level research discussion, and manuscript review. All authors read and approved the final manuscript.

Funding This work was supported by the National Natural Science Foundation of China under Grant 62476276 and Grant 62271486.

Data availability The processed data used in this study are available from the corresponding author upon reasonable request, subject to applicable data-use restrictions.

Code availability The code used in this study is available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors declare that they have no competing interests.

Ethical approval This study used public or anonymized secondary datasets and did not conduct new interventions or collect identifiable personal data from human participants.

References

Abdeddine, A., Iraqi, Y., Mekouar, L.: Mobile crowdsensing model: a survey. J. Syst. Archit. **162**, 103384 (2025). https://doi.org/10.1016/j.sysarc.2025.103384

Anand, V., Maurya, A.K.: A survey on recommender systems using graph neural networks. ACM Trans. Inf. Syst. **43**(1), 1–49 (2025). https://doi.org/10.1145/3694784

Cai, X., Huang, C., Xia, L., Ren, X.: LightGCL: simple yet effective graph contrastive learning for recommendation. In: The Eleventh International Conference on Learning Representations (ICLR) (2023)

Capponi, A., Fiandrino, C., Kantarci, B., Foschini, L., Kliazovich, D., Bouvry, P.: A survey on mobile crowdsensing systems: challenges, solutions, and opportunities. IEEE Commun. Surv. Tutor. **21**(3), 2419–2465 (2019)

Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: Proceedings of the 37th International Conference on Machine Learning (ICML), Proceedings of Machine Learning Research, vol. 119, pp. 1597–1607. PMLR (2020)

Cohen, J.: A coefficient of agreement for nominal scales. Educ. Psychol. Meas. **20**(1), 37–46 (1960). https://doi.org/10.1177/001316446002000104

Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001). https://doi.org/10.1214/aos/1013203451

Ganti, R.K., Ye, F., Lei, H.: Mobile crowdsensing: current state and future challenges. IEEE Commun. Mag. **49**(11), 32–39 (2011). https://doi.org/10.1109/MCOM.2011.6069707

Guo, B., Wang, Z., Yu, Z., Wang, Y., Yen, N.Y., Huang, R., Zhou, X.: Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. ACM Comput. Surveys (CSUR) **48**(1), 1–31 (2015)

Hamilton, W.L., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems (NIPS), pp. 1024–1034 (2017)

Han, L., Yu, Z., Yu, Z., Wang, L., Yin, H., Guo, B.: Online organizing large-scale heterogeneous tasks and multi-skilled participants in mobile crowdsensing. IEEE Trans. Mob. Comput. **22**(5), 2892–2909 (2023). https://doi.org/10.1109/TMC.2021.3132616

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th International Conference on World Wide Web (WWW), pp. 173–182 (2017).

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: LightGCN: simplifying and powering graph convolution network for recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pp. 639–648 (2020)

Hou, Y., Zhang, J., Lin, Z., Lu, H., Xie, R., McAuley, J., Zhao, W.X.: Large language models are zero-shot rankers for recommender systems. In: European Conference on Information Retrieval (ECIR), pp. 364–381. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-56060-6_24

Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. **20**(4), 422–446 (2002). https://doi.org/10.1145/582415.582418

Kazemi, L., Shahabi, C.: GeoCrowd: enabling query answering with spatial crowdsourcing. In: Proceedings of the 20th International Conference on Advances in Geographic Information Systems (SIGSPATIAL GIS), pp. 189–198 (2012). https://doi.org/10.1145/2424321.2424346

Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations (ICLR) (2015)

Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (ICLR) (2017)

Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009). https://doi.org/10.1109/MC.2009.263

Lee, H., Im, J., Jang, S., Cho, H., Chung, S.: MeLU: meta-learned user preference estimator for cold-start recommendation. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD), pp. 1073–1082 (2019)

Li, Z., Li, Z., Zhang, W.: Quality-aware task allocation for mobile crowd sensing based on edge computing. Electronics **12**(4), 960 (2023). https://doi.org/10.3390/electronics12040960

Li, H., Yu, Z., Luo, Y., Cui, H., Guo, B.: ContinuousSensing: a task allocation algorithm for human-robot collaborative mobile crowdsensing with task migration. CCF Trans. Pervasive Comput. Interact. **6**, 228–243 (2024). https://doi.org/10.1007/s42486-024-00154-z

Lin, J., Dai, X., Xi, Y., Liu, W., Chen, B., Zhang, H., Liu, Y., Wu, C., Li, X., Zhu, C., Guo, H., Yu, Y., Tang, R., Zhang, W.: How can recommender systems benefit from large language models: a survey. ACM Trans. Inf. Syst. **43**(2), 1–47 (2025). https://doi.org/10.1145/3678004

Liu, Y., Li, Y., Cheng, W., Wang, W., Yang, J.: UAV-assisted cluster-based task allocation for mobile crowdsensing in a space–air–ground–sea integrated network. Sensors **24**(1), 208 (2024)

Liu, Y., Chen, H., Liu, X., Wei, W., Ma, G., Liu, X., Ye, D.: Quality-aware multi-task allocation based on location importance in mobile crowdsensing. J. Netw. Comput. Appl. **236**, 104113 (2025). https://doi.org/10.1016/j.jnca.2025.104113

Luo, Y., Yu, Z., Yin, H., Cui, H., Guo, B.: Multi-agent mobile crowdsensing by pervasive machines: a robust task allocation approach. CCF Trans. Pervasive Comput. Interact. **5**, 13–30 (2023). https://doi.org/10.1007/s42486-022-00104-7

Shang, J., Jiang, H., Tang, C., Wu, H., Wang, S., Zhang, S.: Multi-stage optimization of incentive mechanisms for mobile crowd sensing based on top-trading cycles. In: Algorithms and Architectures for Parallel Processing. ICA3PP 2023. Lecture Notes in Computer Science, vol. 14487, pp. 170–186. Springer, Singapore (2024). https://doi.org/10.1007/978-981-97-0834-5_11

Shi, C., Hu, B., Zhao, W.X., Yu, P.S.: Heterogeneous information network embedding for recommendation. IEEE Trans. Knowl. Data Eng. **31**(2), 357–370 (2019). https://doi.org/10.1109/TKDE.2018.2833443

Sinnott, R.W.: Virtues of the Haversine. Sky Telesc. **68**(2), 158–159 (1984)

Tong, Y., She, J., Ding, B., Wang, L., Chen, L.: Online mobile micro-task allocation in spatial crowdsourcing. In: 2016 IEEE 32nd International Conference on Data Engineering (ICDE), pp. 49–60. IEEE (2016). https://doi.org/10.1109/ICDE.2016.7498228

van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)

Velickovic, P., Fedus, W., Hamilton, W.L., Lio, P., Bengio, Y., Hjelm, R.D.: Deep Graph Infomax. In: International Conference on Learning Representations (ICLR) (2019)

Wang, L., Yu, Z., Zhang, D., Guo, B., Liu, C.H.: Heterogeneous multi-task assignment in mobile crowdsensing using spatiotemporal correlation. IEEE Trans. Mob. Comput. **18**(1), 84–97 (2019). https://doi.org/10.1109/TMC.2018.2827375

Wang, X., Jin, H., Zhang, A., He, X., Xu, T., Chua, T.S.: Disentangled graph collaborative filtering. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pp. 1001–1010 (2020).

Wei, W., Ren, X., Tang, J., Wang, Q., Su, L., Cheng, S., Wang, J., Yin, D., Huang, C.: LLMRec: large language models with graph augmentation for recommendation. In: Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM), pp. 806–815 (2024). https://doi.org/10.1145/3616855.3635853

Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., Xie, X.: Self-supervised graph learning for recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pp. 726–735 (2021)

Zhang, W., Bei, Y., Yang, L., Zou, H.P., Zhou, P., Liu, A., Li, Y., Chen, H., Wang, J., Wang, Y., Huang, F., Zhou, S., Bu, J., Lin, A., Caverlee, J., Karray, F., King, I., Yu, P.S.: Cold-start recommendation towards the era of large language models (LLMs): a comprehensive survey and roadmap. arXiv preprint arXiv:2501.01945 (2025)

Zhao, B., Dong, H., Yang, D.: A spatio-temporal task allocation model in mobile crowdsensing based on knowledge graph. Smart Cities **6**(4), 1937–1957 (2023). https://doi.org/10.3390/smartcities6040090

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



Yujie Yan Yujie Yan is currently pursuing a master"s degree in Big Data Technology and Engineering at the School of Computer Science and Technology/School of Artificial Intelligence, China University of Mining and Technology, China. His research interests include mobile crowdsensing, with a particular focus on intelligent task assignment and user behavior modeling in mobile sensing systems.



Haifeng Jiang Haifeng Jiang is an Associate Professor with the School of Computer Science and Technology, China University of Mining and Technology, China. He received his Ph.D. degree from China University of Mining and Technology. His research interests include intelligent Internet of Things, crowdsensing computing, federated learning, reinforcement learning, and industrial Internet.



Chaogang Tang Chaogang Tang received his B.S. degree from Nanjing University of Aeronautics and Astronautics and his Ph.D. degree through a joint doctoral program of the University of Science and Technology of China and City University of Hong Kong. He is currently with the School of Computer Science and Technology, China University of Mining and Technology. His research interests include edge and fog computing, vehicular edge computing, the Internet of Things, and deep reinforcement learning.



Shuo Xiao Shuo Xiao is a Professor with the School of Computer Science and Technology, China University of Mining and Technology, China. His research interests include the Internet of Things, natural language processing, artificial intelligence, and big data analytics.



Huaming Wu Huaming Wu is a Professor with the Center for Applied Mathematics, Tianjin University, China. He received his Ph.D. degree in mathematics and computer science from Freie Universität Berlin, Germany, in 2015. His research interests include mobile cloud computing, the Internet of Things, stochastic networks, and queueing theory.



Ruidong Li Ruidong Li is an Associate Professor with the Institute of Science and Engineering, Kanazawa University, Japan. He received his Ph.D. degree from the University of Tsukuba, Japan, in 2008. His research interests include future networks, the Internet of Things, B5G/6G networks, network intelligence, machine learning, and network security.