Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

RCDP: Receiver-Conditioned Disclosure Planning for LLM-Generated Messages

Kai He
Wuhan Textile University
Xueru Zhao
Wuhan Textile University
Jiaoli Shi
shijiaoli@whu.edu.cn

Jiujiang University

Research Article

Keywords:

Posted Date: November 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11684399/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

RCDP: Receiver-Conditioned Disclosure Planning for LLM-Generated Messages

Abstract

Large language models can draft outgoing messages from user-provided context. Yet not all context is suitable for disclosure, and appropriate disclosure levels for the same information can vary across receivers. Existing approaches primarily select task-relevant information or reduce exposure through omission and generalization. However, task-relevant information may contain private details inappropriate for the receiver, while excessive restriction can impede task completion. We propose Receiver-Conditioned Disclosure Planning (RCDP), which formulates disclosure control as unit-level planning in a shared four-level disclosure space. Task-Sufficiency Planning assesses candidate expressions at different levels against the communicative goal's information and specificity requirements, selecting necessary disclosure units and their minimum sufficient levels. Receiver-Boundary Planning sets default upper bounds based on information attributes and the user–receiver relationship, then adjusts them according to the precision required for the current transaction and applicable privacy constraints. The two plans are combined into a final disclosure plan, which is converted into retention, generalization, and omission constraints to guide message generation. We compare RCDP with five baselines on a structured dataset of 1,000 message-generation tasks and 3,872 disclosure units. Compared with Role-aware Direct Prompting, RCDP reduces the over-disclosure rate from 25.70% to 15.78% while increasing the required-information preservation rate from 96.64% to 98.35%. Paired tests that vary user–receiver relationships while holding user context and communicative goals fixed further support its ability to adapt disclosure plans to relationship changes.

Keywords: Large language models, LLM-assisted writing, Contextual privacy, Information disclosure control, Receiver-conditioned disclosure planning

1 Introduction

Large language models can draft complete messages for interpersonal communication tasks, including email writing and scheduling, based on a user's communicative intent and background information (Endacott 2024; Fu et al. 2023). Using intelligent systems to modify or generate messages on a user's behalf is known as AI-mediated communication (Hancock et al. 2020). Compared with sentence-level suggestions, full-message

suggestions have been shown to reduce the time needed to write email replies and improve their helpfulness (Fu et al. 2023).

Users may provide detailed accounts of events, personal circumstances, and reasons for requests to help a model understand their communicative intent. Yet information that helps the model understand a task may be inappropriate to disclose to the receiver. Even when the user's background and communicative goal remain unchanged, the appropriate level of detail for the same information may depend on the receiver's role and relationship with the user.

As Figure 1 illustrates, a user has a job interview at another company that conflicts with a planned lunch. The user wants to explain the conflict and reschedule lunch for the following week. Disclosing the interview to a colleague at the user's current company could reveal an undisclosed job search, but omitting the reason entirely would fail to provide the explanation the user explicitly requested. For a colleague the user is not close to, the reason can be generalized to "a personal commitment"; for a close friend, the specific interview arrangements can be retained. Both messages must retain the rescheduling request, but the appropriate level of detail for the reason varies with the receiver.

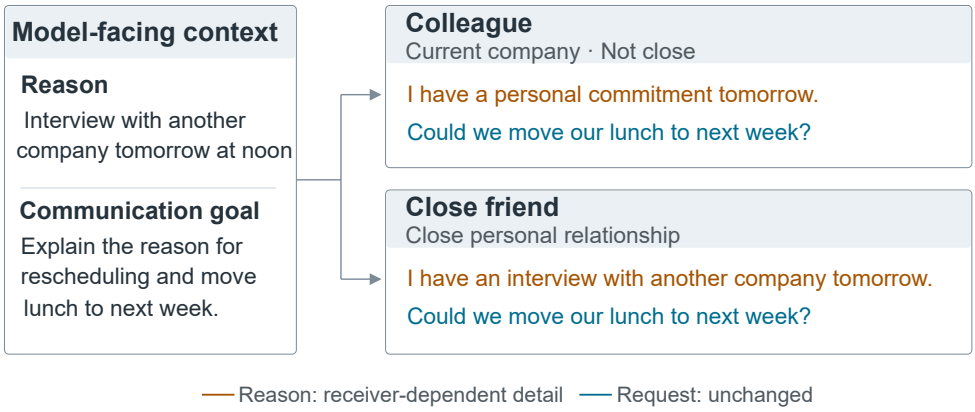


Fig. 1 An example of how disclosure varies with the relationship between the user and the receiver

Existing studies assess the appropriateness of information sharing based on task requirements and interaction context, using contextual integrity reasoning to reduce inappropriate information flows (Ghalebikesabi et al. 2025; Lan et al. 2025). Other work explores omission, generalization, free-text pseudonymization, and task-conditioned abstraction to reduce information exposure while preserving communication utility (Abdelnabi et al. 2026; Xiao et al. 2026). ConVerse further distinguishes information's role in the task from its privacy attributes and evaluates whether information that needs to be shared can be appropriately abstracted (Gomaa et al. 2026). Task relevance alone, however, does not establish which details are appropriate to disclose to the current receiver, and a generalized expression may fail to convey the content or specificity required by the task. A remaining challenge is to control how

specifically the same information is disclosed to different receivers while accounting for task requirements and receiver boundaries.

We propose Receiver-Conditioned Disclosure Planning (RCDP), which formulates disclosure control in outgoing messages as a problem of planning disclosure levels for individual information units. RCDP represents the communicative goal as task slots with requirements for both content and specificity. It extracts disclosure units from user input and constructs candidate expressions at four disclosure levels for each unit.

Task-Sufficiency Planning (TSP) checks whether candidate expressions satisfy the required slots, selects units that cover these slots, and determines their minimum sufficient disclosure levels. Receiver-Boundary Planning (RBP) groups units by their information attributes and roles in the current transaction, then sets default upper bounds based on the user–receiver relationship. It then adjusts these bounds according to the precision required for the current transaction and applicable privacy constraints. For each unit, RCDP takes the lower of the task-required level and the receiver-specific upper bound to form the final disclosure plan. This plan is converted into retention, generalization, and omission constraints that guide message generation.

Our main contributions are as follows:

1. We define receiver-conditioned disclosure control for messages generated by large language models and introduce a unit-level representation with four disclosure levels. This places decisions about whether to include information and how much detail to disclose in a shared decision space.
2. We propose RCDP, which uses TSP and RBP to determine each unit's task-required disclosure level and permitted upper bound, respectively. It combines these into an explicit disclosure plan and converts the plan into generation constraints, linking disclosure planning to message generation.
3. We construct a structured evaluation dataset of 1,000 message-generation tasks and 3,872 disclosure units and evaluate RCDP through comparisons with five baselines, ablation studies, tests that vary user–receiver relationships, and generation experiments with different models. The results show that RCDP reduces over-disclosure while maintaining a high required-information preservation rate and adjusts disclosure plans across receivers.

2 Related Work

This section reviews how large language models assess external information flows in context and control disclosure levels in outgoing text. It also covers supporting techniques for processing private information and generating text under content constraints.

2.1 Context-Sensitive Information Sharing and External Information Flow Control

Contextual integrity views privacy in terms of information flows that conform to contextual norms. The appropriateness of a flow depends on the actors involved, the

type of information, and the conditions of transmission, rather than information sensitivity alone (Nissenbaum 2004). ConfAIde uses this framework to evaluate whether large language models can assess the appropriateness of information sharing in context (Mireshghallah et al. 2024). PrivacyLens extends evaluation to actions such as sending emails, posting on social media, and making tool calls, showing that understanding privacy norms does not reliably translate into control over actual information flows (Shao et al. 2024).

Studies of task execution examine whether models can use information needed for a task without disclosing it inappropriately. Following the principle of data minimization, AgentDAM evaluates whether web agents use information only when it is necessary for the target task (Zharmagambetov et al. 2025). CIMemories places the same user attribute in different task contexts to test whether models adapt their disclosure of stored information to the task, focusing on persistent memory (Mireshghallah et al. 2026). CI-Work examines this issue in retrieval-intensive environments for enterprise agents, requiring models to convey information needed for the task without revealing sensitive context (Fu et al. 2026).

To control information sharing, AirGapAgent separates user-data minimization from third-party interactions, giving the conversational agent access only to task-necessary information (Bagdasarian et al. 2024). Ghalebikesabi et al. (2025) apply contextual integrity to information-sharing decisions in form filling, while Lan et al. (2025) combine explicit contextual reasoning with reinforcement learning to reduce inappropriate disclosure during task execution. However, for information that needs to be shared but should not be revealed in full, the appropriate disclosure level remains to be determined.

2.2 Controlling Disclosure Levels in Outgoing Text

Text abstraction reduces disclosure by expressing information in broader terms. Dou et al. (2024) identify self-disclosures in social media posts and generalize them to reduce privacy risks while preserving their usefulness in context. Say Something Else frames privacy protection in messages generated on users' behalf as a problem of information sufficiency. It compares omission, generalization, and free-text pseudonymization through multi-round follow-up interactions to assess whether these strategies preserve privacy and communication utility (Xiao et al. 2026). ConVerse distinguishes three types of information in multi-turn interactions between agents and external services: irrelevant, relevant but private, and relevant and useful. It evaluates whether models appropriately abstract task-useful information rather than disclose its private details (Gomaa et al. 2026).

The Firewalls framework constrains inter-agent communication with task-specific rules and uses a data abstraction firewall to express user information at the level of detail permitted by the current task (Abdelnabi et al. 2026). Wang and Liu (2026) propose STRATA, which separates disclosure decisions from text generation and uses risk-aware memory and hierarchical planning to determine how individual attributes are disclosed in recommendation explanations. Whereas STRATA aims to prevent the omission of risk-relevant item information, RCDP controls the disclosure of user information to different receivers while preserving task-required content.

2.3 Private Information Processing and Constrained Generation

Input-side approaches limit the private information that user prompts expose to model services. Rescriber uses a small language model running locally to detect and sanitize personal information in user prompts, with user participation in decisions about privacy and utility (Zhou et al. 2025). Huang et al. (2025) introduce NAP^2, a benchmark for privacy-preserving text rewriting, and study deletion and abstraction strategies that preserve naturalness. ALSA retains, replaces, encrypts, or deletes prompt content based on privacy risk, contextual importance, and task relevance (Ma et al. 2025). Pr$\varepsilon\varepsilon$mpt selects protection mechanisms according to how sensitive information contributes to the task response, sanitizing prompts while preserving model utility (Roy Chowdhury et al. 2026).

Constrained generation aims to produce text that meets predefined content requirements. NeuroLogic Decoding incorporates predicate logic constraints to meet complex lexical requirements (Lu et al. 2021). Fang et al. (2023) combine declarative rules with constrained decoding to generate fluent, faithful dialogue responses. Such methods generally assume that content constraints are already specified. RCDP addresses the preceding planning stage: it determines disclosure levels based on task requirements and the user–receiver relationship, then translates the resulting plan into constraints for message generation.

3 Problem Formulation

This section formalizes receiver-conditioned disclosure control by defining task inputs and output, disclosure units and levels, and requirements for task sufficiency and appropriate disclosure to the receiver.

3.1 Outgoing Message Generation

We study how large language models generate complete messages for a specified receiver from the user's full input. This input may include task instructions, facts needed for communication, and background information that helps the model understand the user's intent and situation.

We represent a task instance as

$$x = (u, r, q, b, s, g) \tag{1}$$

where u denotes the user's original input, r the receiver's role, and q the social relationship between the user and the receiver. The base task b may be a request, notification, explanation, or negotiation, and s describes the scenario in which it occurs. The communicative goal g specifies what the receiver needs to understand or do in response to the message.

Given x, the system generates a message y intended to achieve g. It determines which information from u to include and at what disclosure level, based on task requirements and the receiver.

3.2 Disclosure Units and Levels

A disclosure unit is a span of user input u that can be independently retained, generalized, or omitted. Units may contain private background, such as personal experiences or health conditions, or task-related facts, such as times and locations. We denote the set of disclosure units by

$$D = \{d_1, d_2, \ldots, d_n\} \tag{2}$$

where d_i is the ith disclosure unit and n is the number of units. Each unit is assigned a disclosure level from

$$\mathcal{L} = \{\text{None}, \text{Domain}, \text{Category}, \text{Exact}\} \tag{3}$$

The levels are ordered by increasing specificity:

$$\text{None} < \text{Domain} < \text{Category} < \text{Exact}. \tag{4}$$

None omits the unit from the final message; Domain conveys only a broad domain or general circumstance; Category identifies the information category while omitting finer details; and Exact preserves the original fact or an equally specific paraphrase. For example, "attending a job interview" can be generalized to "a personal commitment" at Domain or "a work-related commitment" at Category.

3.3 Disclosure Control Objectives

Disclosure control has two requirements: task sufficiency and appropriateness for the receiver.

Task sufficiency requires the final message to convey the information needed to achieve g in sufficient detail for the receiver to understand or respond as intended. Essential information must therefore be neither omitted nor overly generalized. Appropriateness for the receiver requires each unit's disclosure level to reflect the receiver's role and relationship with the user. Even task-relevant information may warrant different disclosure levels across receivers.

For each disclosure unit d_i, let m_i^{final} denote its target disclosure level in the final message y. When the two requirements are compatible, the target level should satisfy both. If they conflict, it must remain within the receiver's disclosure boundary. Section 4 describes how the task-required lower bounds and receiver-specific upper bounds are determined and combined.

4 Method

4.1 RCDP Overview

RCDP has three stages (Fig. 2): task and disclosure unit modeling, disclosure-level planning, and constrained message generation. It first identifies task slots, extracts disclosure units, and constructs candidate expressions. For each unit, TSP determines the minimum disclosure level needed for the task, and RBP sets the maximum permitted for the receiver. These bounds are combined into a final disclosure plan that guides message generation through explicit constraints.

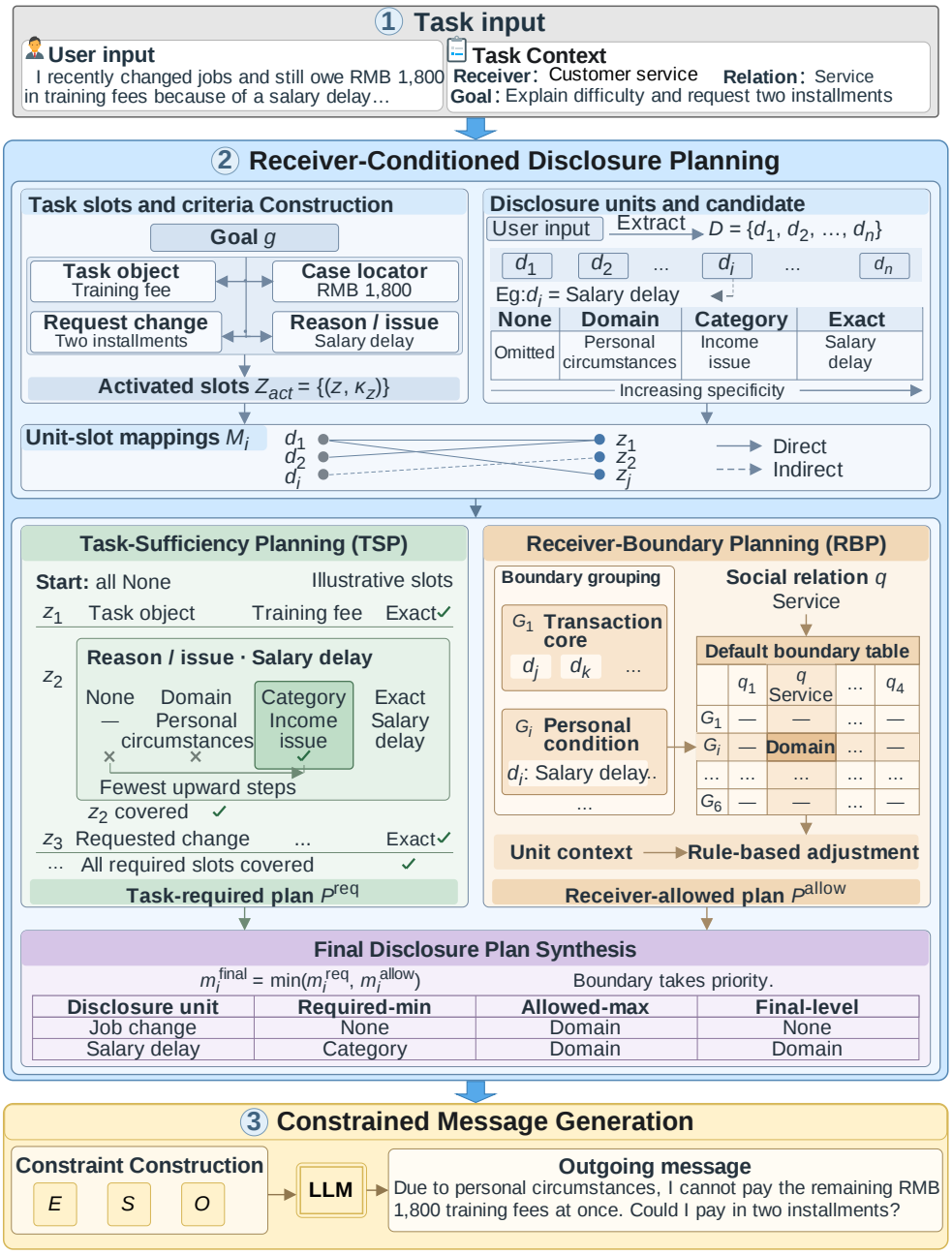


Fig. 2 Overview of the RCDP framework

4.2 Modeling Task Conditions and Disclosure Units

RCDP links the communicative goal to user information through task slots, level-specific candidate expressions, and unit–slot mappings, providing a common basis for TSP and RBP.

4.2.1 Task Slots and Criterion Construction

Task-slot activation. RCDP defines four task-slot types for evaluating task sufficiency: task object, case locator, requested change detail, and reason or issue. Given the communicative goal g, predefined rules activate the slots required by the current task.

Criterion construction. For each activated slot z, RCDP constructs a criterion κ_z specifying the content and level of detail required by the task. For example, "make other arrangements" is insufficient when the task requires rescheduling a meal for the following week. The activated slots and their criteria are represented as

$$Z_{\text{act}} = \{(z, \kappa_z) \mid z \text{ is required by } g\}. \tag{5}$$

RCDP also constructs communicative constraints:

$$C_{\text{comm}} = (c_{\text{act}}, c_{\text{resp}}, c_{\text{tone}}, c_{\text{fact}}). \tag{6}$$

where c_{act} and c_{resp} ,derived from g,specify the communicative act and expected receiver response. The component c_{tone} captures the user's explicit writing requirements, while c_{fact} requires fidelity to the input facts. These constraints guide message generation but do not affect disclosure-level decisions.

4.2.2 Disclosure Unit Extraction and Candidate Construction

Disclosure unit extraction. RCDP uses predefined lexicons, patterns, and task-role cues to identify candidate spans in u, including task-related information, private background, and reasons. A contiguous span is split when its components can be independently retained, generalized, or omitted; components that must remain together to express a single fact are kept as one unit. RCDP then normalizes span boundaries and removes duplicates to obtain

$$D = \text{Dedup}(\text{Normalize}(\text{Split}(\text{Detect}(u)))) = \{d_1, \ldots, d_n\}. \tag{7}$$

For each unit, RCDP also records its information type, information subject, and whether it contains third-party private information.

Level-specific candidate construction. For each unit d_i, RCDP constructs a candidate expression $c_i(\ell)$ at each disclosure level ℓ according to the unit's information type:

$$c_i(\ell) = \begin{cases} \varepsilon, & \ell = \text{None}, \\ \text{a broad domain expression}, & \ell = \text{Domain}, \\ \text{a category expression}, & \ell = \text{Category}, \\ \text{the original span in } d_i, & \ell = \text{Exact}. \end{cases} \tag{8}$$

Here, ε denotes an empty expression. Domain and Category candidates follow predefined mappings from information types to expressions. Each unit has exactly one normalized candidate per level. Representative Domain and Category expressions are listed in Table B.1.

4.2.3 Disclosure Unit–Task Slot Mapping

Task-slot matching. RCDP uses the information type of d_i to identify potentially supported activated slots. It then resolves ambiguous matches using κ_z and semantic and causal relationships in u, retaining only the slots that d_i supports in the current task.

Support-type labeling. Support is labeled direct if the unit can independently satisfy κ_z at some candidate disclosure level, and indirect if it provides relevant background or explanation without independently satisfying the criterion. Let σ_{iz} denote the support relation from d_i to z. The mapping set for d_i is

$$M_i = \{(z, \sigma_{iz}) \mid (z, \kappa_z) \in Z_{\text{act}},\ \sigma_{iz} \in \{\text{direct}, \text{indirect}\}\}. \tag{9}$$

A unit may support multiple activated slots. If it supports none, $M_i = \varnothing$. Both TSP and RBP use M_i as input.

4.3 Task-Sufficiency Planning

TSP determines task-required disclosure levels using the activated slots and their criteria Z_{act}, candidate expressions $c_i(\ell)$, and unit–slot mappings M_i. A candidate $c_i(\ell)$ satisfies an activated slot z if d_i directly supports z, the expression is nonempty, and it meets κ_z:

$$h_z(d_i, \ell) = \mathbb{I}[(z, \text{direct}) \in M_i \ \wedge\ c_i(\ell) \neq \varepsilon \ \wedge\ c_i(\ell) \models \kappa_z]. \tag{10}$$

Here, $\mathbb{I}[\cdot]$ equals 1 when its condition holds and 0 otherwise, and $c_i(\ell) \models \kappa_z$ means that the expression conveys the content and level of detail required by the slot. Let $m^{req}=(m_1^{req},\ldots,m_n^{req})$ denote the current disclosure-level assignment. Slot z is covered if at least one unit satisfies it at its current level:

$$H_z(m^{\text{req}}) = \mathbb{I}[\exists i \in \{1, \ldots, n\} : h_z(d_i, m_i^{\text{req}}) = 1]. \tag{11}$$

Algorithm 1 initializes all units at None and increases disclosure levels only to cover unmet required slots. For each uncovered slot in a fixed order, it selects a candidate requiring the fewest upward steps in the disclosure-level ordering. After each update, coverage is re-evaluated for all activated slots. The resulting assignments form the task-required disclosure plan P^{req}.

LexicographicSelect prioritizes the fewest upward steps in the ordering in Eq. (4), then the most currently uncovered slots satisfied, and finally a fixed source order. If $A_z=\varnothing$, no update is made for slot z.

Algorithm 1 Task-Sufficiency Planning

Require: Disclosure units D; activated slots and criteria Z_{act}; candidates $c_i(\ell)$; unit–slot mappings M_i; ordered levels $\mathcal{L}$.
Ensure: Task-required disclosure plan P^{req}.

Set $m_i^{\text{req}} \leftarrow$ None for every $d_i \in D$
for each $(z, \kappa_z) \in Z_{\text{act}}$ in a fixed order **do**
  if $H_z(m^{\text{req}}) = 0$ **then**
    $A_z \leftarrow \{(i, \ell) \mid \ell \in \mathcal{L},\ h_z(d_i, \ell) = 1,\ \ell > m_i^{\text{req}}\}$
    if $A_z \neq \varnothing$ **then**
      $(i^*, \ell^*) \leftarrow \text{LexicographicSelect}(A_z)$
      $m_{i^*}^{\text{req}} \leftarrow \ell^*$
      Re-evaluate coverage for all activated slots
    end if
  end if
end for
return $P^{\text{req}} = \{(d_i, m_i^{\text{req}}) \mid i = 1, \ldots, n\}$

4.4 Receiver-Boundary Planning

RBP determines each unit's maximum permitted disclosure level for the current receiver by assigning boundary groups, setting default upper bounds, and applying unit-level adjustments.

Step 1: Boundary Group Assignment. RBP assigns $\mathrm{d_i}$ to a boundary group based on its information attributes and $\mathrm{M_i}$. Units that directly specify task objects, requested changes, handling outcomes, or disputed facts to be addressed belong to transaction_core. Units primarily used to identify specific cases, records, or documents belong to transaction_locator. Other units are assigned to personal_condition, identity_affiliation, private_schedule_location, or relationship_context according to their content. When a unit serves multiple purposes, its group is determined by its primary function in the current communication. Appendix B.2 provides the detailed assignment criteria. The assigned group is

$$G_i = B(d_i, M_i). \tag{12}$$

Step 2: Default Upper-Bound Assignment. Given $\mathrm{G_i}$ and the social relationship q between the user and the receiver, RBP assigns a default upper bound:

$$m_{i,0}^{\text{allow}} = A_0(G_i, q). \tag{13}$$

where $\mathrm{A_0}$ maps each boundary group and social relationship to a default upper bound, as specified in Table 1.

Table 1 Default upper bounds by social relation and boundary group

Boundary group G_i	**Service**	**Institutional**	**Weak tie**	**Close**
transaction_core	Exact	Exact	Category	Exact
transaction_locator	Exact	Category	Domain	Category
personal_condition	Domain	Category	Domain	Category
identity_affiliation	Domain	Category	Domain	Category
private_schedule_location	Domain	Category	Domain	Exact
relationship_context	Domain	Domain	Domain	Category

Step 3: Unit-Level Adjustment. RBP adjusts the default upper bounds based on relevance to the current communication, the receiver's responsibilities, and privacy attributes, using the following three rules.

- **Third-party privacy limits.** For units containing third-party private facts, the disclosure cap is None if they are clearly unrelated to the current communication, and Domain if they are directly or indirectly relevant or their relevance is uncertain. Applying this cap does not increase a lower default upper bound.
- **Limits on irrelevant private background.** Private background information is capped at None when it is unrelated to both the current matter and its explanatory context and has no reasonable purpose for sharing given the relationship and communicative goal. Information that provides indirect explanatory context or serves a reasonable sharing purpose is not downshifted merely because it lacks a direct slot mapping.
- **Transaction-precision adjustment.** A unit in transaction_core or transaction_locator qualifies for an Exact upper bound when its content must be accurately identified, verified, modified, or used as a basis for action within the receiver's current responsibilities.

Personal reasons are subject to the default bounds and disclosure restrictions applicable to their information attributes. Let $\mathrm{p_i}$ be Exact when the transaction-precision condition is met and None otherwise. Let $\mathrm{m_i^{cap}}$ denote the lowest cap imposed by all applicable disclosure restrictions, or Exact if none applies. The final permitted upper bound is

$$m_i^{\text{allow}} = \min\{\max(m_{i,0}^{\text{allow}}, p_i), m_i^{\text{cap}}\}. \tag{14}$$

These bounds form the upper-bound plan:

$$P_{\text{allow}} = \{(d_i, m_i^{\text{allow}}) \mid i = 1, \ldots, n\}. \tag{15}$$

4.5 Final Disclosure Plan Synthesis

RCDP combines the outputs of TSP and RBP to determine the final disclosure level for each unit:

$$m_i^{\text{final}} = \begin{cases} m_i^{\text{req}}, & m_i^{\text{req}} \le m_i^{\text{allow}}, \\ m_i^{\text{allow}}, & m_i^{\text{req}} > m_i^{\text{allow}}. \end{cases} \tag{16}$$

The task-required level is retained when it does not exceed the receiver's upper bound; otherwise, the upper bound takes priority. These assignments form the final disclosure plan:

$$P = \{(d_i, m_i^{\text{req}}, m_i^{\text{allow}}, m_i^{\text{final}}) \mid i = 1, \ldots, n\}. \tag{17}$$

4.6 Constraint-Guided Message Generation

RCDP converts the final disclosure plan P into semantic constraints for retention, generalization, and omission:

$$\begin{aligned} E &= \{d_i \mid m_i^{\text{final}} = \text{Exact}\}, \\ S &= \{(d_i, c_i(m_i^{\text{final}})) \mid m_i^{\text{final}} \in \{\text{Domain}, \text{Category}\}\}, \\ O &= \{d_i \mid m_i^{\text{final}} = \text{None}\}. \end{aligned} \tag{18}$$

For units in E, the model must preserve the original facts and their specificity, including identifiers, amounts, and times where applicable. For entries in S, the model may paraphrase the candidate expression $c_i(m_i^{\text{final}})$, but must preserve its meaning without adding more specific details. The message must omit units in O and any paraphrases of their content.

The prompt supplies these constraints, the task input x, and the communicative constraints C_{comm} to the generation model G:

$$y = G(x; E, S, O, C_{\text{comm}}). \tag{19}$$

Box B.1 in Appendix B.3 outlines the generation prompt structure; the complete template is provided in Section S1.3 of Online Resource 1.

5 Dataset and Experimental Setup

This section describes dataset construction and annotation, the experimental setup, and the evaluation protocol and metrics.

5.1 Dataset and Scenario Construction

We constructed a structured dataset of 1,000 single-turn message-generation tasks in Chinese to evaluate disclosure control and the preservation of task-required information. Each example contains user input, receiver information, and a communicative goal. The dataset covers multiple task types, social relationships, and writing scenarios, with distributions shown in Table 2. Table A.1 in Appendix A.1 maps receiver roles to social relationships.

Dataset construction involved three stages.

(1) Task and scenario extraction. We extracted service task types and interaction scenarios from the Action-Based Conversations Dataset (ABCD; Chen et al. 2021) and the Schema-Guided Dialogue Dataset (SGD; Rastogi et al. 2020). We also identified complaint and appeal scenarios from the CFPB Consumer Complaint Database (Consumer Financial Protection Bureau n.d.). These resources informed the

task structures and scenarios; their original texts were not used directly as dataset examples.

(2) Candidate example generation. We used an LLM to generate candidate user inputs based on these task types and communication scenarios.

(3) Human review. Researchers reviewed and revised each candidate to ensure that the communicative goal, receiver relationship, and task scenario were aligned, that the input provided sufficient task information, and that the example was internally consistent. Examples containing explicit privacy-control instructions were excluded or rewritten.

Table 2 Distribution of the 1,000 evaluation samples

Dimension	Category	Count (%)
Task type	Requests and negotiations	631 (63.1%)
	Transaction handling	228 (22.8%)
	Complaints and appeals	141 (14.1%)
Social relation	Institutional	326 (32.6%)
	Service	321 (32.1%)
	Weak tie	308 (30.8%)
	Close	45 (4.5%)
Writing scenario	Applications and negotiations	306 (30.6%)
	Social communication	253 (25.3%)
	Workplace communication	168 (16.8%)
	Complaints and appeals	141 (14.1%)
	Customer service communication	132 (13.2%)

Note: Each dimension summarizes all 1,000 evaluation examples.

5.2 Annotation Scheme and Quality Control

We created structured gold annotations for each example, covering 3,872 disclosure units in total. The annotations include:

- ***Task requirements.*** We record the required task slots, their criteria, and the message's communicative constraints.
- ***Disclosure units and their task roles.*** For each unit, we record its source span, information type and subject, whether it involves third-party private information, candidate expressions at all four levels, and direct or indirect support for activated slots.
- ***Gold disclosure plan.*** For each unit, required-min and allowed-max are annotated, and final-level is derived using Eq. (16). These labels support evaluation of disclosure plans and generated messages.

Researchers manually annotated an initial set of examples and developed guidelines, then used an LLM to produce preliminary annotations for the remaining examples. All preliminary annotations were manually reviewed and revised item by item. Sections A.2–A.4 of Appendix A provide the annotation rules and a complete example.

To assess inter-annotator agreement, a second annotator independently annotated a stratified sample of 150 examples. Before comparing disclosure-level labels, we aligned semantically corresponding units while allowing minor differences in span boundaries, yielding 550 aligned units. The F_1 score for disclosure-unit extraction was 0.790. On the aligned units, weighted Cohen's κ values for required-min, allowed-max, and final-level were 0.655, 0.705, and 0.740, respectively.

All agreement statistics were computed before adjudication. The annotators then resolved disagreements in this subset through discussion. The gold annotations were finalized and fixed before the experimental runs and evaluation. Appendix A.5 details the alignment procedure, agreement results, and adjudication process.

5.3 Experimental Setup

Baselines. We compare RCDP with five baselines grouped into three categories.

(1) Unprotected generation. No Protection generates messages directly from the full user input without disclosure control.

(2) Text transformations. Redaction removes private information (Pilán et al. 2022). Generalization rewrites specific details in broader terms while preserving their meaning (Pilán et al. 2025). Pseudonymization replaces private facts with fictitious information (Xiao et al. 2026; Yermilov et al. 2023).

(3) Role-aware Direct Prompting. Inspired by the privacy-enhancing prompts evaluated by Shao et al. (2024), this baseline includes the communicative goal, receiver role, user–receiver relationship, and task scenario in the prompt. It instructs the model to avoid disclosing information the receiver does not need, without explicitly planning disclosure levels.

All five baselines use fixed prompts to implement their respective strategies. Section S1.2 of Online Resource 1 provides the complete templates.

Ablation settings. We evaluate two ablations to examine the roles of TSP and RBP:

- **w/o TSP.** We disable the task-sufficiency functions listed in Appendix B.4 and use the upper bounds produced by the remaining boundary rules as the final disclosure levels.
- **w/o RBP.** We disable receiver-conditioned upper bounds and their adjustment rules and use the task-required levels as the final disclosure levels.

Both variants retain constraint conversion and message generation. Appendix B.4 provides implementation details.

Execution settings. All methods receive the same task inputs: user input, receiver role, social relationship, base task, scenario, and communicative goal. Gold annotations are used only for evaluation. The baselines generate messages directly, whereas RCDP and its ablations first construct disclosure plans and convert them into generation constraints. All methods are instructed to remain faithful to the user input, except that Pseudonymization may introduce fictitious information as required by its replacement strategy.

RCDP's rules and implementation were fixed before the main experiment. All methods used Qwen3.6-Plus via Model Studio with temperature set to 0 (Alibaba

Cloud 2026a,b). Sections S1.1 and S2 of Online Resource 1 provide the common prompting protocol and complete runtime parameters, respectively.

Supplementary experiments. To assess robustness to changes in the generation model, we regenerated messages for RCDP and all five baselines on a stratified sample of 300 examples using DeepSeek-V4-Pro Preview. We retained the method implementations from the main experiment, and RCDP reused its original disclosure plans and generation constraints. Model-specific runtime settings are reported in Section S4.1 of Online Resource 1.

We also conducted two receiver-boundary diagnostic analyses. First, we varied receiver relationships in 38 pairs while holding user context and communicative goals fixed, testing whether planned disclosure levels for the same units changed with the receiver. Second, we compared RCDP with w/o RBP at both the planning and output levels on 285 boundary-sensitive examples from the main evaluation set. These examples contain private or third-party background information whose disclosure must be limited according to the receiver. Fig. 3 summarizes the method execution and evaluation workflow.

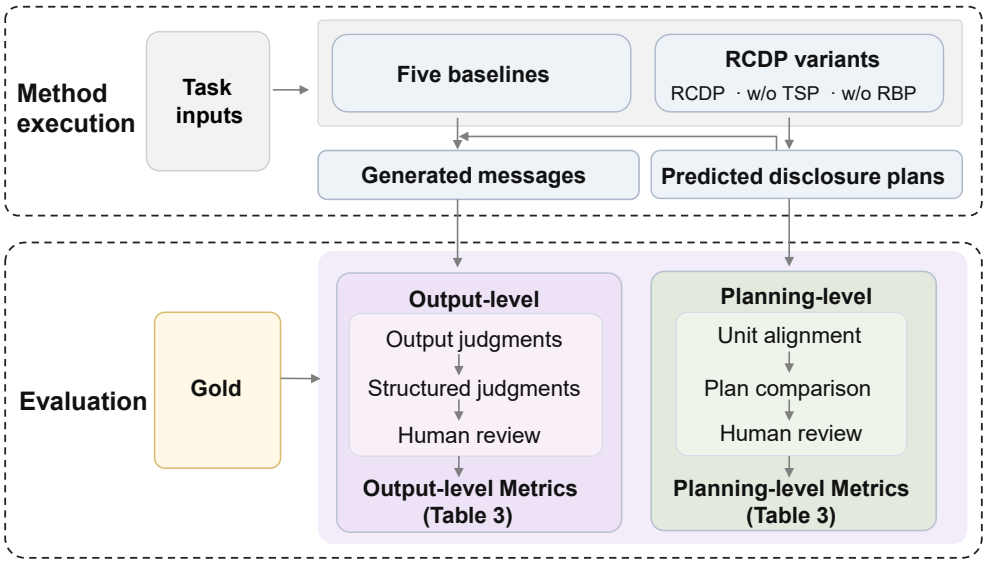


Fig. 3 Method execution and evaluation workflow

5.4 Evaluation Protocol and Metrics

Evaluation scope and metrics. Our primary evaluation assesses generated messages using five output-level metrics. Three planning-level metrics support ablation analysis and receiver-boundary diagnostics. Table 3 defines the metrics and their denominators.

Evaluation procedures. We evaluate outputs and disclosure plans as follows:

- Output-level evaluation. Using the task inputs and gold annotations, we assess actual disclosure levels in generated messages, coverage of required task slots, and unsupported

additions. Following structured LLM-based evaluation (Liu et al. 2023), Qwen3.6-Plus produces preliminary judgments under a predefined schema. Researchers review and revise each judgment before metric calculation (Gu et al. 2026). Section S1.4 of Online Resource 1 provides the complete judgment template, and Appendix C.1 describes the structured fields and human review procedure.

- **Planning-level evaluation.** We match predicted units to gold units using exact source-span matching. Metrics are computed only for matched gold units with comparable disclosure-level labels, assessing whether planned levels match gold final-level, fall below task requirements, or exceed receiver boundaries.

Table 3 Evaluation metric definitions

Panel A. Output-level Metrics

Metric	Definition / Denominator
Over-disclosure Rate (ODR) ↓	Proportion of evaluated disclosure units whose realized disclosure level in the generated message exceeds gold final-level.
Boundary Violation Rate (BVR) ↓	Proportion of evaluated disclosure units whose realized disclosure level in the generated message exceeds gold allowed-max.
Goal Success Slot Coverage (GSC) ↑	Proportion of required disclosure-relevant task slots covered by the generated message.
Required-information Preservation Rate (RPR) ↑	Among disclosure units with gold required-min $\neq$ None and gold required-min $\leq$ gold allowed-max, the proportion whose realized disclosure level reaches gold required-min.
Unsupported Addition Rate (UAR) ↓	Proportion of generated messages containing at least one unsupported fact, reason, commitment, request, or follow-up action.

Panel B. Planning-level Metrics

Metric	Definition / Denominator
Final-level Accuracy ↑	Among gold disclosure units with corresponding predicted labels, the proportion whose predicted final-level matches gold final-level.
Under-required Plan Rate ↓	Among gold disclosure units with corresponding predicted labels, gold required-min $\neq$ None and gold required-min $\leq$ gold allowed-max, the proportion whose predicted final-level is lower than gold required-min.
Over-boundary Plan Rate ↓	Proportion of evaluated disclosure units whose predicted final-level exceeds gold allowed-max.

Note: For output-level unit metrics, evaluated disclosure units are gold units for which judgments have been completed based on the generated message. For planning-level metrics, they are gold units with a comparable predicted final-level. ↑ indicates that higher values are better; ↓ indicates that lower values are better.

Evaluation agreement. A second annotator independently reviewed 900 messages generated by RCDP and the five baselines for a stratified sample of 150 tasks. With

method identities hidden and messages randomly ordered, the annotator assessed actual disclosure levels, slot coverage, and unsupported additions. We compared these assessments with the human-reviewed judgments used to calculate the metrics. Section S3.1 of Online Resource 1 and Appendix C.2 provide the review scope and agreement statistics, respectively.

Confidence intervals. We used task-level bootstrap resampling to compute 95% confidence intervals for all five output-level metrics in the main experiment and for paired differences between RCDP and each baseline. Appendix C.3 provides the calculation details.

6 Results and Analysis

This section analyzes RCDP's disclosure control and preservation of task-required information. Ablation studies, paired receiver tests, and error analysis examine the role of disclosure planning and whether generated messages follow the resulting plans.

6.1 Overall Performance

Table 4 compares RCDP with the five baselines. RCDP achieves the highest RPR and the lowest UAR among the evaluated methods. It also reduces over-disclosure relative to Role-aware Direct Prompting.

Table 4 Output-level results for RCDP and the baselines.

Method	ODR ↓	BVR ↓	GSC ↑	RPR ↑	UAR ↓
No Protection	0.3461	0.3461	0.9970	0.9807	0.0080
Redaction	0.1296	0.1291	0.9250	0.9044	0.0120
Generalization	0.3050	0.3024	0.9960	0.9545	0.0130
Pseudonymization	0.1916	0.1875	0.9977	0.9415	0.6360
Role-aware Direct Prompting	0.2570	0.2554	0.9935	0.9664	0.0120
RCDP	0.1578	0.1547	0.9889	0.9835	0.0060

Note: Metric abbreviations follow Table 3. All methods were evaluated on the same 1,000 examples using human-reviewed structured judgments. Unit-level metrics use the 3,872 gold disclosure units, subject to the eligibility conditions in Table 3.

Disclosure control. Compared with Role-aware Direct Prompting, RCDP reduces ODR from 25.70% to 15.78% and BVR from 25.54% to 15.47%, while increasing RPR from 96.64% to 98.35%.

Task information preservation. Redaction achieves the lowest ODR (12.96%), but its RPR and GSC are 90.44% and 92.50%, respectively. RCDP has a higher ODR but achieves an RPR of 98.35% and a GSC of 98.89%. This contrast shows why disclosure metrics should be considered alongside task performance: deletion can reduce disclosure at the cost of information needed for communication.

Fig. 4 shows paired differences and 95% confidence intervals for ODR and RPR. For both metrics, the intervals lie entirely above zero in comparisons with Role-aware Direct Prompting, Generalization, and Pseudonymization.

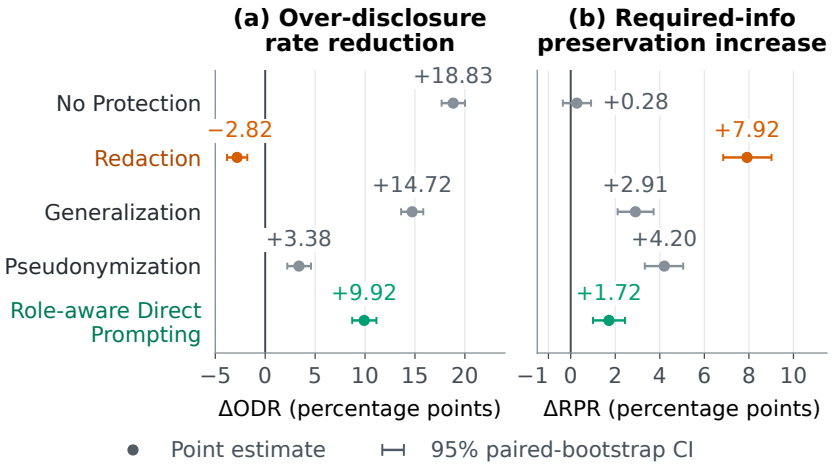


Fig. 4 Paired performance differences between RCDP and the baselines. (a) Over-disclosure rate reduction, calculated as baseline minus RCDP; (b) required-information preservation rate increase, calculated as RCDP minus baseline. Points show differences, and error bars show 95% paired bootstrap confidence intervals obtained by resampling the 1,000 tasks. Values are in percentage points; positive values favor RCDP. ODR, over-disclosure rate; RPR, required-information preservation rate; CI, confidence interval

Performance across generation models. On the 300 examples evaluated with DeepSeek-V4-Pro Preview, RCDP again achieves the highest RPR (98.66%). Its ODR and BVR are lower than those of Role-aware Direct Prompting and slightly higher than those of Redaction. This pattern is consistent with the main experiment, suggesting that RCDP's disclosure plans remain effective with a different generation model. Full results are provided in Section S4.2 of Online Resource 1.

6.2 Ablation Study of Disclosure Planning Components

Table 5 reports planning- and output-level results for RCDP and its ablation variants to assess the roles of TSP and RBP.

TSP's role in preserving task information. Without TSP, the under-required plan rate rises from 0.25% to 35.22%. In generated messages, RPR falls from 98.35% to 85.92%, while ODR rises from 15.78% to 19.83%. These results suggest that TSP helps preserve task-required information while limiting unnecessary disclosure.

RBP's role in controlling boundary violations. Without RBP, the over-boundary plan rate rises from 0.26% to 7.50%, and BVR increases from 15.47% to 20.20%. RPR remains nearly unchanged at 98.32%. These results indicate that task sufficiency alone does not ensure compliance with receiver boundaries.

Table 5 Results for RCDP and its ablation variants

Panel A: Planning-level ablation

(a) Removing TSP

Variant	Final-level Acc. ↑	Under-required ↓
RCDP	0.9951	0.0025
w/o TSP	0.6163	0.3522

(b) Removing RBP

Variant	Final-level Acc. ↑	Over-boundary ↓
RCDP	0.9951	0.0026
w/o RBP	0.9234	0.0750

Panel B: Output-level ablation

Variant	ODR ↓	BVR ↓	GSC ↑	RPR ↑	UAR ↓
RCDP	0.1578	0.1547	0.9889	0.9835	0.0060
w/o TSP	0.1983	0.1937	0.9650	0.8592	0.0150
w/o RBP	0.2051	0.2020	0.9907	0.9832	0.0220

Note: Panel A reports planning-level results. Panel B reports output-level results. Metric abbreviations follow Table 3.

6.3 Receiver-Boundary Sensitivity Analysis

Table 6 reports diagnostic results from the paired receiver tests and the boundary-sensitive subset.

Table 6 Diagnostics for receiver pairs and boundary-sensitive cases

Subset	Metric / statistic	Result
Receiver-variant pairs	Pairs with same-span final-level change	17 / 38
	Changed same-span unit comparisons	23
	Final-level accuracy on aligned changed paired unit instances	42 / 44 = 0.9545
Boundary-sensitive cases	Planning over-boundary rate	0.2905 → 0.0000
	BVR	0.4201 → 0.2363

Note: The paired receiver tests comprise 38 pairs (76 examples) and 238 disclosure units. In 17 pairs, at least one identical source span has different gold final-level labels across receivers, yielding 23 paired unit comparisons (46 unit instances). Accuracy is computed for the 44 instances aligned with predicted units; the two unmatched instances are excluded. The boundary-sensitive subset, selected using existing labels in the main evaluation set, contains 285 examples and 876 disclosure units. Arrows indicate the change from w/o RBP to RCDP.

Disclosure adjustments across receivers. With user context and communicative goals held fixed, 17 of the 38 pairs contain units whose gold final-level varies with the receiver. RCDP correctly predicts the final disclosure level for 42 of the 44 aligned unit

instances (95.45%). This result supports RCDP's ability to adjust planned disclosure levels to receiver differences on the evaluated units.

RBP's control of boundary-sensitive information. On the 285 boundary-sensitive examples, RCDP reduces the over-boundary plan rate from 29.05% without RBP to 0%, and BVR from 42.01% to 23.63%. RBP thus improves boundary control at both levels, although violations remain in generated messages.

6.4 Error Analysis

Table 7 summarizes errors in RCDP outputs against the task inputs and gold annotations. We further analyze boundary violations in relation to the predicted disclosure plans. Representative error cases are provided in Section S3.2 of Online Resource 1 (Table S3).

Table 7 Error types and frequencies in RCDP outputs.

Error type	Counting basis	Count	Error criterion
Over-specific realization	Disclosure unit	508	Actual level > gold final-level; gold final-level ≠ None
Residual leakage of suppressed information	Disclosure unit	103	Gold final-level = None; actual level > None
Required-info under-preservation	Required disclosure unit	47	Actual level < gold required-min; gold required-min ≠ None and gold required-min ≤ gold allowed-max
Required slot miss	Required slot	44	Required disclosure-relevant slot is not covered
Unsupported addition	Message	6	Message contains unsupported facts, reasons, commitments, requests, or follow-up actions

Note: The evaluation comprises 3,872 disclosure units, including 2,855 required disclosure units; 3,973 required task slots; and 1,000 messages. Counts use the counting basis specified for each error type.

Major error types. The main forms of over-disclosure are overly specific wording (508 units) and residual disclosure of information that should have been omitted (103 units). Disclosure control must therefore govern both whether information is included and how specifically it is expressed.

Boundary violations. As shown in Fig. 5, 585 of the 599 units with output boundary violations (97.66%) have planned levels at or below gold allowed-max, but their actual disclosure levels exceed both the plan and that upper bound. Most output boundary violations therefore reflect failures to follow the disclosure plan during generation. Among these 585 units, 71.79% exceed the plan by one level, while 37 are fully disclosed at Exact despite being planned for omission (None).

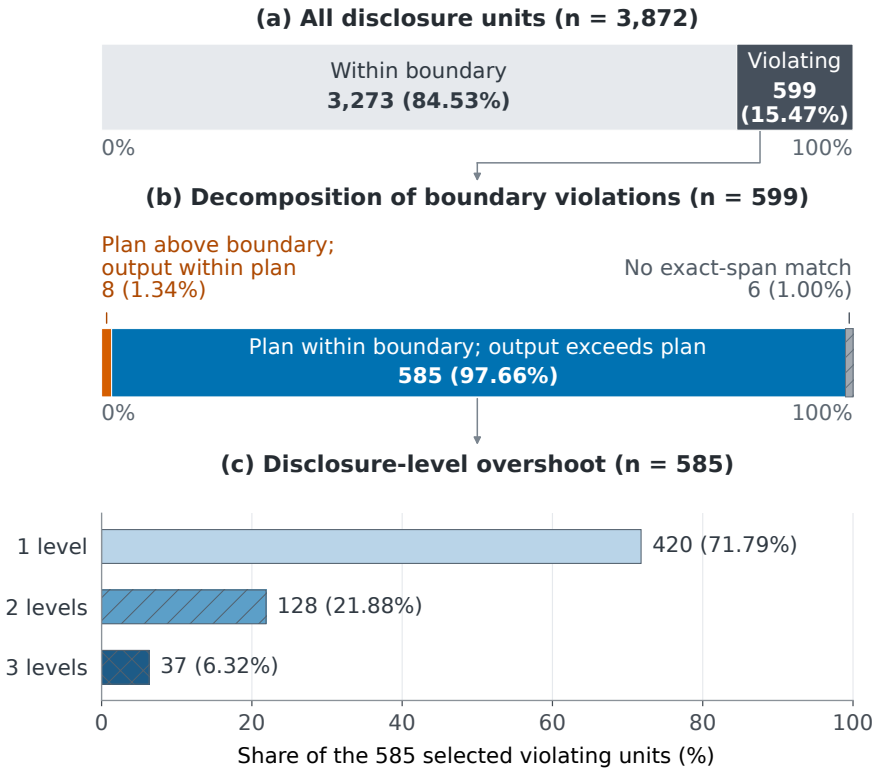


Fig. 5 Boundary violations and disclosure-level deviations in RCDP outputs. (a) Boundary violation rate across all 3,872 gold disclosure units. (b) Classification of the 599 violating units by the relationships among planned levels, actual disclosure levels, and receiver upper bounds. (c) Number of levels by which outputs exceed plans for the 585 violating units whose plans remain within the boundary. In (b), six units lack an exact source-span match to a predicted unit; no matched unit has both a plan above the boundary and an output above the plan. Plans and receiver upper bounds refer to predicted final-level and gold allowed-max, respectively. Level differences use None = 0, Domain = 1, Category = 2, and Exact = 3. All panels summarize disclosure units

7 Conclusion

We presented RCDP for disclosure control in LLM-generated outgoing messages. It combines task-required lower bounds from TSP with receiver-specific upper bounds from RBP to produce a final disclosure plan that guides message generation.

In comparisons with five baselines on the 1,000-task evaluation set, RCDP achieved the highest RPR (98.35%) and reduced the ODR from 25.70% to 15.78% relative to Role-aware Direct Prompting. Ablation studies support TSP's role in preserving task-required information and RBP's role in controlling boundary violations. Paired receiver tests support RCDP's ability to adapt disclosure plans to different receiver relationships. A similar overall pattern was observed with a different generation model.

These findings support jointly considering task sufficiency and receiver boundaries before message generation.

The evaluation primarily used constructed single-turn tasks in Chinese, so RCDP's applicability to real-world communication remains to be established. Receiver-boundary annotations may not fully capture individual preferences or cultural differences. Predefined task slots, boundary rules, and discrete disclosure levels may also limit the handling of implicit or overlapping relationships among disclosure units. Generated messages may exceed planned disclosure levels or include information intended for omission. Future work will investigate personalized disclosure boundaries using real user data, extend evaluation to multilingual and multi-turn settings, and improve adherence to disclosure plans during generation.

References

Abdelnabi S, Gomaa A, Bagdasarian E, Kristensson PO, Shokri R (2026) Firewalls to secure dynamic LLM agentic networks. Transactions on Machine Learning Research. https://openreview.net/forum?id=w02FW1dMoY

Alibaba Cloud (2026a) Model calls in a sub-workspace. https://www.alibabacloud.com/help/en/model-studio/model-calling-in-sub-workspace. Accessed 18 Sep 2026

Alibaba Cloud (2026b) qwen3.6-plus. https://www.alibabacloud.com/help/en/model-studio/qwen3-6-plus. Accessed 18 Sep 2026

Bagdasarian E, Yi R, Ghalebikesabi S, Kairouz P, Gruteser M, Oh S, Balle B, Ramage D (2024) AirGapAgent: Protecting privacy-conscious conversational agents. In: Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security pp 3868–3882. Association for Computing Machinery. https://doi.org/10.1145/3658644.3690350

Chen D, Chen H, Yang Y, Lin A, Yu Z (2021) Action-based conversations dataset: A corpus for building more in-depth task-oriented dialogue systems. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies pp 3002–3017. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.239

Consumer Financial Protection Bureau (n.d.) Consumer Complaint Database [dataset]. https://www.consumerfinance.gov/data-research/consumer-complaints/. Accessed 15 Aug 2026

Dou Y, Krsek I, Naous T, Kabra A, Das S, Ritter A, Xu W (2024) Reducing privacy risks in online self-disclosures with language models. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) pp 13732–13754. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.741

Endacott CG (2024) Enacting machine agency when AI makes one's day: Understanding how users relate to AI communication technologies for scheduling. Journal of Computer-Mediated Communication 29(4):zmae011. https://doi.org/10.1093/jcmc/zmae011

Fang H, Balakrishnan A, Jhamtani H, Bufe J, Crawford J, Krishnamurthy J, Pauls A, Eisner J, Andreas J, Klein D (2023) The whole truth and nothing but the truth: Faithful and controllable dialogue response generation with dataflow transduction and constrained decoding. In: Findings of the Association for Computational Linguistics: ACL 2023 pp 5682–5700. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-acl.351

Fu L, Newman B, Jakesch M, Kreps S (2023) Comparing sentence-level suggestions to message-level suggestions in AI-mediated communication. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems article 103, pp 1–13. Association for Computing Machinery. https://doi.org/10.1145/3544548.3581351

Fu W, Qin X, Zhang J, Lin Q, Wutschitz L, Sim R, Rajmohan S, Zhang D (2026) CI-Work: Benchmarking contextual integrity in enterprise LLM agents. In: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track) pp 1483–1508. Association for Computational Linguistics. https://doi.org/10.18653/v1/2026.acl-industry.103

Ghalebikesabi S, Bagdasarian E, Yi R, Yona I, Shumailov I, Pappu A, Shi C, Weidinger L, Stanforth R, Berrada L, Kohli P, Huang PS, Balle B (2025) Privacy awareness for information-sharing assistants: A case-study on form-filling with contextual integrity. Transactions on Machine Learning Research. https://openreview.net/forum?id=l9rATNBB8Y

Gomaa A, Salem A, Abdelnabi S (2026) ConVerse: Benchmarking contextual safety in agent-to-agent conversations. In: Findings of the Association for Computational Linguistics: EACL 2026 pp 3246–3268. Association for Computational Linguistics. https://doi.org/10.18653/v1/2026.findings-eacl.170

Gu F, Li Z, Colon CR, Evans B, Mondal I, Boyd-Graber J (2026) Large language models are effective human annotation assistants, but not good independent annotators. In: Findings of the Association for Computational Linguistics: ACL 2026 pp 71–89. Association for Computational Linguistics. https://doi.org/10.18653/v1/2026.findings-acl.4

Hancock JT, Naaman M, Levy K (2020) AI-mediated communication: Definition, research agenda, and ethical considerations. Journal of Computer-Mediated Communication 25(1):89–100. https://doi.org/10.1093/jcmc/zmz022

Huang S, MacLean W, Kang X, Xu Q, Li Z, Yuan X, Haffari G, Qu L (2025) NAP^2: A benchmark for naturalness and privacy-preserving text rewriting by learning from human. In: Findings of the Association for Computational Linguistics: EMNLP

2025 pp 8954–8970. Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.findings-emnlp.476

Lan G, Inan HA, Abdelnabi S, Kulkarni J, Wutschitz L, Shokri R, Brinton CG, Sim R (2025) Contextual integrity in LLMs via reasoning and reinforcement learning. In: Advances in Neural Information Processing Systems vol 38, pp 115503–115539. Curran Associates, Inc. https://doi.org/10.52202/085713-3485

Liu Y, Iter D, Xu Y, Wang S, Xu R, Zhu C (2023) G-Eval: NLG evaluation using GPT-4 with better human alignment. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing pp 2511–2522. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.153

Lu X, West P, Zellers R, Le Bras R, Bhagavatula C, Choi Y (2021) NeuroLogic decoding: (Un)supervised neural text generation with predicate logic constraints. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies pp 4288–4299. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.339

Ma H, Lu W, Liang Y, Wang T, Zhang Q, Zhu Y, Si J (2025) ALSA: Context-sensitive prompt privacy preservation in large language models. In: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 pp 2042–2053. Association for Computing Machinery. https://doi.org/10.1145/3711896.3736840

Mireshghallah N, Kim H, Zhou X, Tsvetkov Y, Sap M, Shokri R, Choi Y (2024) Can LLMs keep a secret? Testing privacy implications of language models via contextual integrity theory. In: The Twelfth International Conference on Learning Representations. https://openreview.net/forum?id=gmg7t8b4s0

Mireshghallah N, Mangaokar N, Kokhlikyan N, Zharmagambetov A, Zaheer M, Mahloujifar S, Chaudhuri K (2026) CIMemories: A compositional benchmark for contextual integrity in LLMs. In: The Fourteenth International Conference on Learning Representations. https://proceedings.iclr.cc/paper_files/paper/2026/hash/9a2bcfaf383638e166162a25b6dff125-Abstract-Conference.html

Nissenbaum H (2004) Privacy as contextual integrity. Washington Law Review 79(1):119–158. https://digitalcommons.law.uw.edu/wlr/vol79/iss1/10/

Pilán I, Lison P, Øvrelid L, Papadopoulou A, Sánchez D, Batet M (2022) The Text Anonymization Benchmark (TAB): A dedicated corpus and evaluation framework for text anonymization. Computational Linguistics 48(4):1053–1101. https://doi.org/10.1162/coli_a_00458

Pilán I, Manzanares-Salor B, Sánchez D, Lison P (2025) Truthful text sanitization guided by inference attacks. Applied Soft Computing 185:114013. https://doi.org/

10.1016/j.asoc.2025.114013

Rastogi A, Zang X, Sunkara S, Gupta R, Khaitan P (2020) Towards scalable multi-domain conversational agents: The Schema-Guided Dialogue dataset. Proceedings of the AAAI Conference on Artificial Intelligence 34(5):8689–8696. https://doi.org/10.1609/aaai.v34i05.6394

Roy Chowdhury A, Glukhov D, Anshumaan D, Chalasani P, Papernot N, Jha S, Bellare M (2026) Prɛɛmpt: Sanitizing sensitive prompts for LLMs. In: Proceedings of the Network and Distributed System Security Symposium 2026. Internet Society. https://doi.org/10.14722/ndss.2026.231277

Shao Y, Li T, Shi W, Liu Y, Yang D (2024) PrivacyLens: Evaluating privacy norm awareness of language models in action. In: Advances in Neural Information Processing Systems vol 37, pp 89373–89407. Curran Associates, Inc. https://doi.org/10.52202/079017-2837

Wang C, Liu B (2026) Mitigating selective disclosure in recommender explanations via risk-aware memory and hierarchical planning. Journal of King Saud University Computer and Information Sciences 38:452. https://doi.org/10.1007/s44443-026-00826-z

Xiao Y, Li W, Wu X, Ma N, Song Y, Xuan W (2026) Say something else: Rethinking contextual privacy as information sufficiency. arXiv preprint arXiv:2604.06409. https://doi.org/10.48550/arXiv.2604.06409

Yermilov O, Raheja V, Chernodub A (2023) Privacy- and utility-preserving NLP with anonymized data: A case study of pseudonymization. In: Proceedings of the Third Workshop on Trustworthy Natural Language Processing pp 232–241. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.trustnlp-1.20

Zharmagambetov A, Guo C, Evtimov I, Pavlova M, Salakhutdinov R, Chaudhuri K (2025) AgentDAM: Privacy leakage evaluation for autonomous web agents. In: Advances in Neural Information Processing Systems vol 38, pp 153145–153170. Curran Associates, Inc. https://doi.org/10.52202/085713-4609

Zhou J, Xu E, Wu Y, Li T (2025) Rescriber: Smaller-LLM-powered user-led data minimization for LLM-based chatbots. In: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems article 246, pp 1–28. Association for Computing Machinery. https://doi.org/10.1145/3706598.3713701

Appendix A Dataset and Gold Annotation Details

A.1 Receiver Roles and Social Relations

Table A.1 Receiver roles grouped by social relation.

Social relation	Receiver roles
Service	Platform customer service; platform complaint handler; merchant customer service; service-provider customer service; billing representative; training provider customer service; account specialist; loan servicing representative; refund specialist; bank customer service; credit card customer service; e-commerce platform customer service
Institutional	Course instructor; course administrator; direct supervisor; teacher; teaching assistant; reviewer; HR representative; manager; academic advisor; colleague
Weak tie	Casual colleague; casual friend; club member; classmate; team member; project member; neighbor
Close	Partner; friend; family member

A.2 Task Slot, Unit Extraction, and Mapping Rules

Table A.2 Task-slot definitions and unit–slot mapping criteria.

Slot	Definition and criterion	Non-mapping cases
task_object	Identifies the primary object addressed by the message, such as an order, leave request, charge record, or project file.	A background source, institutional setting, or personal experience alone does not identify the task object.
case_locator	Enables the receiver to identify a specific case, record, material, or event through information such as an order number, transaction ID, meeting time, or document link.	General background that does not help identify the specific case.
requested_change_detail	Specifies the concrete change, arrangement, or handling outcome sought by the user, such as a revised deadline, installment payment, or account correction.	Routine requests such as “please take a look” or “please understand” that do not specify a concrete change or outcome.
reason_or_issue	Provides a reason, problem fact, error state, or supporting basis for the current request, complaint, appeal, or explanation.	Private background that describes the user’s situation without supporting the current task.

Disclosure units are annotated at the smallest span that can independently support a disclosure decision. The annotation covers task-relevant facts, private background information, and third-party information. Writing instructions, tone requirements, connectives, and routine communicative acts are not annotated as separate units. A continuous span is split when its components can receive different disclosure decisions; components that jointly express a single complete fact are retained as one unit.

Table A.3 Disclosure unit types used in annotation.

Unit type	**Annotation scope and examples**
Organization/ institution information	Names of companies, schools, platforms, banks, or hospitals, such as Meituan or Peking Union Medical College Hospital.
Employment/ identity status	Resignation, dismissal, a recent job change, an internship, or student status.
Financial condition	Limited funds, interrupted income, cash-flow difficulty, or credit-card repayment pressure.
Health information	Follow-up examination, gastroscopy, emergency treatment, unstable physical condition, or unstable emotional condition.
Family/ relationship information	Family members, roommates, partners, friends, family matters, or relationship conflicts.
Location/ itinerary information	Hospitals, cities, business travel, moving, rental-related arrangements, or private itineraries.
Temporal information	9:00 a.m., Saturday evening, next Friday, or the morning of May 26.
Work-context information	Weekly project meetings, departmental meetings, requirements reviews, or work discussions.
Amount/fee information	Rent amounts, training fees, or duplicate-charge amounts.
Transaction identifier information	Order numbers, transaction IDs, account numbers, telephone numbers, or document links.
Task object	The order, leave request, charge record, or project file to be handled.
Event/reason information	A failed submission, missing refund, inaccessible account, or incorrect record.

Unit–slot mappings follow the criteria in Table A.2. A mapping is labeled direct if the unit can independently satisfy the slot criterion at some candidate disclosure level, and indirect if it provides supporting background without independently satisfying that criterion. Units that support no activated slot receive an empty mapping.

A.3 Planning Label Annotation Rules

The required-min label follows the minimum-sufficiency principle: it selects the lowest candidate level that satisfies the corresponding task slot. Identifiers, amounts, times, and other information needed to locate a specific case, as well as concrete changes

that the receiver must make, are labeled Exact when the task cannot be completed without their specific values. Reasons or issue facts generally need to be retained only at the Domain or Category level, whereas additional private background that does not directly support task completion is labeled None.

The allowed-max label is determined from the receiver role, social relation, information content, information subject, and the unit's role in the task. Task-core facts and necessary locator information generally permit greater specificity, whereas health, financial, employment, family-relationship, private-location, and third-party information receive more conservative upper bounds. A close relationship may permit a higher disclosure level, but does not automatically warrant an Exact label.

A.4 Gold Annotation Example

This section presents a complete gold annotation example, showing how the task context, activated slots, disclosure units, unit-slot mappings, and planning labels correspond within a single sample.

Table A.4 Sample context for the gold annotation example.

Field	Value
Receiver role	Training provider customer service
Social relation	Service
User input	I recently changed jobs, and my salary payment has been delayed. I still owe RMB 1,800 for this training fee and would like to ask whether I can pay it in two installments. Please write a sincere negotiation message to the training provider's customer service representative.
Communicative goal	Explain the user's payment difficulty and ask the training provider's customer service representative to allow the user to pay the remaining RMB 1,800 training fee in two installments.

Table A.5 Goal-success slots for the gold annotation example.

Slot ID	Slot name	Required	Description
D1	task_object	Yes	Identify the training fee as the object of the negotiation.
D2	requested_change_detail	Yes	Specify the request to pay in two installments.
D3	case_locator	Yes	Specify the outstanding amount of RMB 1,800.
D4	reason_or_issue	Yes	Indicate delayed income or payment difficulty supporting the installment request.

Table A.6 Unit-level gold annotation for the example.

Unit	Span	Type	Slot	Support	Req. min	Allow. max	Final
U1	recently changed jobs	Employment/ identity status	D4	indirect	None	Domain	None
U2	salary payment has been delayed	Event/reason information	D4	direct	Category	Domain	Domain
U3	this training fee	Task object	D1	direct	Exact	Exact	Exact
U4	RMB 1,800	Amount/fee information	D3	direct	Exact	Exact	Exact
U5	pay it in two installments	Event/reason information	D2	direct	Exact	Exact	Exact

Note: Slot and Support record the task slot supported by each unit and the form of support, respectively. required-min = None indicates that the unit is not selected for the minimally sufficient disclosure plan. Slot IDs refer to Table A.5. Req. min, Allow. max, and Final denote required-min, allowed-max, and final-level, respectively.

A.5 Inter-Annotator Agreement

A second annotator independently annotated a stratified sample of 150 instances. Because disclosure units were annotated as spans, the two sets of units were aligned one-to-one using exact-span matching, containment matching, and character-overlap matching, in that order, with a character-overlap threshold of 0.55. The primary annotation contained 707 disclosure units, and the second annotator identified 685 units. The procedure aligned 550 units, with a mean alignment score of 0.827.

Precision, recall, and F1 for disclosure-unit extraction were computed over all annotated units. Agreement for unit type, unit-slot mapping, required-min, allowed-max, and final-level was computed over the 550 aligned units. For unit-slot mapping, each (slot_name, support_relation) pair was treated as the comparison unit. Table A.7 reports agreement before adjudication.

Table A.7 Inter-annotator agreement for gold annotations.

Component	Metric	Value
Unit extraction	Precision	0.803
	Recall	0.778
	F1	0.790
Unit type	Exact agreement	0.655
	Cohen's kappa	0.603
Unit-slot mapping	Precision	0.524
	Recall	0.619
	F1	0.568
Required-min	Exact agreement	0.613
	Weighted Cohen's kappa	0.655
Allowed-max	Exact agreement	0.753
	Weighted Cohen's kappa	0.705
Final-level	Exact agreement	0.693
	Weighted Cohen's kappa	0.740

After completing the independent annotations, the two annotators reviewed and adjudicated disagreements to produce the final gold annotations.

Appendix B Implementation Details

B.1 Disclosure-Level Candidate Construction

Table B.1 Unit-type mappings for Domain and Category candidates.

Unit type	Domain expression	Category expression
Organization/ institution information	relevant organization; relevant background	organization information; employer information; organizational affiliation
Employment/ identity status	personal circumstances	employment change; job-search arrangement; job risk; work pressure
Financial condition	personal circumstances	financial pressure; income change; payment difficulty; delayed income receipt
Health information	personal circumstances	physical discomfort; health condition; medical examination; treatment arrangement
Family/ relationship information	personal circumstances	family matter; relationship change; family circumstance; caregiving arrangement
Location/ itinerary information	personal arrangement; relevant background	location arrangement; itinerary; medical location
Temporal information	relevant time; relevant matter	date; time period; deadline; meeting or event time
Work-context information	work arrangement; relevant circumstances	departmental meeting; project review; work activity
Amount/fee information	relevant expense; relevant matter	amount; rent; service fee; charge details
Transaction identifier information	relevant record; relevant matter	order number; transaction ID; account number; document link
Task object	relevant item; relevant matter	order; application; meeting; report; social arrangement
Event/reason information	relevant circumstances; a related matter	unexpected event; scheduling conflict; submission issue; accompanying someone

Note: These mappings are used to construct normalized candidates and do not constitute an exhaustive lexicon.

B.2 Receiver-Boundary Planning Rules

Boundary-group assignment prioritizes the task role of each disclosure unit: a direct mapping takes precedence over an indirect mapping. Among mappings with the same support relation, the primary role is determined in the order of task_object or requested_change_detail, case_locator, and reason_or_issue. For units supporting reason_or_issue, unit type and subject scope are additionally used to distinguish task facts, personal conditions, and relationship context. Units without slot support are grouped by information type and information subject.

Table B.2 Boundary-group assignment criteria.

Boundary group	**Assignment criterion**
transaction_core	The disclosure unit represents a task object, requested change, concrete handling outcome, or a task fact such as a disputed fact, error state, or basis for appeal.
transaction_locator	The disclosure unit is primarily used to identify a specific case, record, material, time, or identifier in the current task.
personal_condition	The disclosure unit describes the information subject's financial, health, physical, or psychological condition.
identity_affiliation	The disclosure unit describes the information subject's identity, employer, school, or organizational affiliation.
private_schedule_location	The disclosure unit describes the information subject's private location, time, itinerary, or activity schedule.
relationship_context	The disclosure unit describes family, partner, friend, colleague, roommate, or other relationship context.

Note: third_party_private is not a separate boundary group; it is used as a signal for subsequent allowed-max adjustment.

B.3 Constrained Generation Prompt

Box B.1 presents the prompt structure used to generate messages from the disclosure plan. The complete template is provided in Section S1.3 of Online Resource 1.

Box B.1 Representative prompt skeleton for RCDP constrained generation.

Input information
Receiver role: {receiver_role}
Social relation: {social_relation}
Base task: {base_task}
Scenario: {scenario}
Communicative goal: {communicative_goal}
User input: {user_input}

Communicative constraints:
- communicative act: {communicative_act}
- expected response: {response_expectation}
- tone requirement: {tone_requirement}
- factuality constraint: {factuality_constraint}

Disclosure constraints
exact_preserve_units: {exact_preserve_units}
semantic_constraints: {semantic_constraints}
omitted_units: {omitted_units}

Instructions:
1. Faithfully express every unit listed in exact_preserve_units, preserving all task-critical values.
2. Express semantic_constraints at their specified abstraction levels; semantic labels need not be copied verbatim.
3. Do not reveal omitted_units or their semantic equivalents.
4. Do not exceed the predicted final-level of any disclosure unit.
5. Do not add facts, promises, requests, or commitments unsupported by the input.
6. Output only the final receiver-facing message.

B.4 Ablation Implementation

Table B.3 Implementation settings of planning-function ablations.

Ablation	Disabled functionality	Final-level construction
w/o TSP	Required-min planning, including candidate-sufficiency and slot-coverage decisions; limits on irrelevant private background; transaction-precision adjustment.	Retain boundary grouping, relation-conditioned default bounds, and third-party privacy limits; use the resulting allowed-max as final-level.
w/o RBP	Receiver-boundary planning, including default upper-bound assignment and all unit-level adjustments.	Use required-min from the complete TSP as final-level.

Appendix C Evaluation and Statistics

C.1 Output Judgment and Human Review

Table C.1 summarizes the structured judgment fields used with the template in Section S1.4 of Online Resource 1.

Human reviewers verify and correct the LLM-generated judgments against the task input, gold annotations, and generated message before metric calculation. The same procedure is used for the main and cross-generator experiments.

Table C.1 Main fields in output-level structured judgments.

Field group	Description	Example	Metrics / use
actual_disclosure_levels	Actual disclosure level of each gold unit in the generated message.	None, Domain, Category, Exact	ODR; BVR; RPR
slot_coverage	Whether each required disclosure-relevant task slot is covered.	covered=true/false	GSC
unsupported_addition	Whether the message introduces facts, reasons, commitments, requests, or follow-up actions unsupported by the user input.	true/false; unsupported span + reason	UAR
review_evidence	Output spans and brief rationales supporting human review.	output span; brief note	Review / adjudication

Note: Metric abbreviations follow Table 3.

C.2 Second-Annotator Agreement

The second annotator independently assessed actual disclosure levels, slot coverage, and unsupported additions, with method identities hidden and messages presented in random order. Table C.2 reports agreement with the human-reviewed judgments used for metric calculation. The review scope is specified in Section S3.1 of Online Resource 1.

Table C.2 Second-annotator agreement for output-level judgments.

Judgment field	Agreement	Kappa type	Kappa
actual disclosure level	0.940	Weighted Cohen's kappa	0.934
slot coverage	0.994	Cohen's kappa	0.776
unsupported addition	0.904	Cohen's kappa	0.626

C.3 Bootstrap Confidence Intervals for the Main Experiment

Table C.3 reports point estimates and 95% confidence intervals for the five output-level metrics, using 5,000 task-level bootstrap resamples with random seed 20260628. For

the paired intervals in Figure 4, methods were resampled using identical task indices, and metrics and differences were recalculated in each resample. Interval bounds are the 2.5th and 97.5th percentiles of the corresponding bootstrap distributions.

Table C.3 Bootstrap confidence intervals for the main experiment.

Panel A: Disclosure control

Method	**ODR ↓**	**BVR ↓**	**UAR ↓**
No Protection	0.3461 [0.3317, 0.3604]	0.3461 [0.3317, 0.3604]	0.0080 [0.0030, 0.0140]
Redaction	0.1296 [0.1189, 0.1408]	0.1291 [0.1185, 0.1402]	0.0120 [0.0060, 0.0190]
Generalization	0.3050 [0.2920, 0.3182]	0.3024 [0.2893, 0.3155]	0.0130 [0.0060, 0.0210]
Pseudonymization	0.1916 [0.1795, 0.2040]	0.1875 [0.1753, 0.1997]	0.6360 [0.6050, 0.6650]
Role-aware Direct Prompting	0.2570 [0.2426, 0.2710]	0.2554 [0.2412, 0.2694]	0.0120 [0.0060, 0.0190]
RCDP	0.1578 [0.1462, 0.1690]	0.1547 [0.1431, 0.1660]	0.0060 [0.0020, 0.0110]

Panel B: Task-information preservation

Method	**GSC ↑**	**RPR ↑**
No Protection	0.9970 [0.9952, 0.9985]	0.9807 [0.9753, 0.9858]
Redaction	0.9250 [0.9175, 0.9325]	0.9044 [0.8938, 0.9150]
Generalization	0.9960 [0.9940, 0.9977]	0.9545 [0.9466, 0.9621]
Pseudonymization	0.9977 [0.9962, 0.9990]	0.9415 [0.9333, 0.9496]
Role-aware Direct Prompting	0.9935 [0.9909, 0.9957]	0.9664 [0.9594, 0.9730]
RCDP	0.9889 [0.9856, 0.9919]	0.9835 [0.9786, 0.9881]

Note: Entries are point estimates [95% confidence interval]. Metric abbreviations follow Table 3.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- ESM1.pdf