Web Appendices for

Engaging Customers with AI in Online Chats: Evidence from a Randomized Field Experiment

Appendix A: Randomization Checks of the Field Experiment

We verify the randomness of the timing regarding enabling access to L.ai's AI assistance tool among Company X's agents. Company X selected agents for AI treatment assignment in a staggered manner; thus, adoption occurred on many different dates during the sample period. It would be difficult to conduct pairwise tests among all adoption dates. Instead, we follow Seamans and Zhu (2014) and Proserpio et al. (2021): as shown in Equation (A.1), we regressed the number of days into the study when agents were assigned to AI on agent characteristics:

$$AdoptionTiming_i = Constant + \rho \cdot Agent\_Variables\_preAI_i + \varepsilon_i, \quad \text{(A.1)}$$

where $AdoptionTiming_i$ indicates the number of days between the adoption date and the start date of data collection (i.e., December 1, 2020). $\rho \cdot Agent\_Variables\_preAI_i$ captures the agent characteristics of interest (measured prior to AI assignment): agent tenure, number of messages per conversation, number of conversations per day, average response time, and channel shift (whether an agent worked in the chat channel only, i.e., "chat only," or worked chat *and* in other channels, i.e., "mix"). Thus, the estimated coefficient ρ empirically tests to what extent agent characteristics predict the timing of the AI treatment.

We perform a set of five regressions and report the results in Table S1. The coefficients are statistically insignificant for all characteristics. The results suggest that agent characteristics do not predict the timing of AI assignment; that is, when an agent received treatment, it was not dependent on their characteristics.

Table S1. Randomization Check: Timing of AI Treatment

VARIABLES	ESTIMATES (Std. Err.)

Agent tenure	–0.0446
	(0.0687)
Number of messages from agent per conversation	–6.187
	(5.575)
Number of conversations per day	–0.723
	(0.989)
Average response time (seconds)	–0.890
	(0.588)
Channel shift (“chat only”; using “mix” as baseline)	–28.79
	(17.10)

Note. The dependent variable is the timing of enabling AI to an agent, and the independent variables are agent-related characteristics that were measured as the mean of all pre-treatment periods. Five regressions are performed separately. Robust standard errors are in parentheses. * $p < 0.05$ ** $p < 0.01$ *** $p < 0.001$

Appendix B: Using an HTE-Robust Estimator to Address Staggered Treatment Concerns

B.1. Staggered Adoption Design and the IFEct (Liu et al. 2024) Estimator

We conducted a robustness analysis in which we used an alternative approach to estimate the impact of AI on agent-customer conversations. Note that, in our randomized experiment, AI was not made available to all chat agents simultaneously; instead, the company implemented a staggered adoption design, as discussed in Athey and Imbens (2022) and Liu et al. (2024). Recently, there has been a growing examination of the robustness of the TWFE estimator, which is our main specification (see Section 2.5), in scenarios of staggered adoption.

As explained by de Chaisemartin and D'Haultfoeuille (2020), the TWFE estimator compiles a weighted average of all individual treatment effects across each unit and time point. More specifically, when treatment is staggered and binary (as in our case), the TWFE estimator is a weighted average of individual ATTs from 2×2 difference-in-differences cells by exploiting variance in treatment status in the cells (Goodman-Bacon 2021). However, even with the parallel trend assumption, the TWFE estimand might not accurately reflect a convex combination of individual treatment effects in cases where treatments start at different times *and* their effects change over time. The underlying reason for this issue is that TWFE models utilize data from units that adopted the treatment early as a control for units that adopted it later. In these cases, "already-treated" agents—those treated before time *t*—also serve as controls. When treatment effects vary over time (heterogeneous treatment effects, or HTE), some treated units receive negative weights, leading to potential bias; because of negative weighting, the aggregated ATT could be misleadingly negative (or positive) even if all individual treatment effects are positive (or negative).

To address the possible bias in the TWFE model, estimators robust to HTE have been proposed for treatments under staggered adoption, for example, Arkhangelsky et al. (2021; Synthetic DiD (difference-in-differences)), Callaway and Sant'Anna (2021), Sun and Abraham (2021; interaction-weighted estimator), Wooldridge (2021; staggered DiD), and Liu et al. (2024; interactive fixed effects counterfactual or IFEct). In particular, Liu et al. (2024) proposed the IFEct estimator and addressed the potential negative weighting

issue with the counterfactual estimation approach. This method estimates the counterfactuals for each treated observation by using an interactive fixed effects model of outcomes on observations that are untreated (or not yet treated). Subsequently, it calculates the individual treatment effect for each treated unit by subtracting the predicted counterfactual outcome from the actual observed outcome. From there, both the treatment effect for each period and average treatment effect are determined.

Among the HTE-robust estimators, we chose IFEct as our robustness approach because of its numerous advantages. First, unlike methods such as synthetic DiD, which do not allow for the inclusion of time-varying covariates, IFEct can flexibly incorporate time-varying covariates. This is advantageous for our study because we aim to include agent tenure and conversation characteristics to assess their impacts on outcome variables. Second, our panel data are imbalanced because we do not observe every agent every day because some agents join or leave the company, take leave, or work on nonchat channels on certain days. The IFEct method can accommodate this, whereas methods such as those by Arkhangelsky et al. (2021) and Goodman-Bacon (2021) require a balanced panel. Third, by the end of our data window, all agents had been given access to AI, meaning that we could use only the "not-yet-treated" but not the "never-treated" as the control group. This is perfectly aligned with how the IFEct method constructs the counterfactual outcome, whereas the interaction-weighted methods, such as the one proposed by Sun and Abraham (2021), use "never-treated" or "last-treated" as the control group. Fourth, the IFEct model is well suited for our large dataset.[1] It is more computationally efficient because it estimates the outcome model only once. In contrast, methods such as those of Strezhnev (2018), Sun and Abraham (2021), and Callaway and Sant'Anna (2021) might not handle our large data well because they attempt to estimate all cohort average treatment effects on the treated given cohort or relative time and then aggregate them. Finally, the IFEct estimator allows for time-varying unobserved confounders, which most other methods do not accommodate.

[1] We thank Yiqing Xu (author of Liu et al. 2024) for pointing this out.

The IFEct method specifies the following outcome model for unit *i* in time *t* as in Equation (B.1):

$$Y_{it}(0) = g(X_{it}) + u(U_{it}) + \epsilon_{it}, \tag{B.1}$$

where $g(X_{it}) = X_{it}\beta$ illustrates how the outcome depends on the observed characteristics, while $u(U_{it}) = \alpha_i + \tau_t + \lambda_i f_t$ explains the influence of unobserved factors on the outcome. Note that α_i represents the time-invariant unobserved individual unit fixed effect and τ_t is the time fixed effect, both of which are additive fixed effects. Additionally, $\lambda_i f_t$ are the interactive fixed effects capturing time-varying unobserved confounders by breaking down these confounders into a vector of unobserved common or latent factors $f_t = [f_{1t}, f_{2t}, \dots, f_{rt}]'$ and a vector of unknown factor loadings $\lambda_i = [\lambda_{i1}, \lambda_{i2}, \dots., \lambda_{ir}]$. Generally, any unobserved random variable that can be expressed in a multiplicative form can be incorporated through $\lambda_i f_t$ (the interaction terms might vary from unit-specific linear or quadratic time trends to autoregressive components; refer to Gobillon and Magnac 2016). The dimension *r* must be smaller than the dimensions of data *N* (number of units) and *T* (number of time periods). The derivation of λ_i and f_t follows the assumption of a lower rank representation of U with r<<N, T: We rewrite $\boldsymbol{u}(\boldsymbol{U})$=$\boldsymbol{\Lambda F}$, where $\boldsymbol{\Lambda} \in \mathbb{R}^{N\times r}$ is a matrix of factor loadings and $\mathbf{F} \in \mathbb{R}^{r\times T}$ is a matrix of factors. Note that α_i and τ_t are just special cases of $\lambda_i f_t$, where the interaction terms become a set of unit fixed effects when f_t is constant and a set of time fixed effects when λ_i is constant.

As seen from Equation (B.1), the IFEct method makes the following assumptions:

(1) the outcome can be modeled parametrically and possesses the property of additive separability,

(2) strict exogeneity is satisfied, which ensures a parallel treatment condition, and

(3) there exists a low-dimensional decomposition of $u(U)$ into a matrix of factors and a matrix of factor loadings.

The IFEct method models the response surface of potential outcomes for (*i*, *t*) if untreated:

$$Y_{it}(0) = X_{it}'\beta + \alpha_i + \tau_t + \lambda_i' f_t + \epsilon_{it}, \tag{B.2}$$

Liu et al. (2024) demonstrated that, under these assumptions, the IFEct estimator remains unbiased in situations of staggered adoption and temporal changes in the treatment effect.

B.2. Estimation Algorithm of the IFEct Model

Regarding the estimation of the IFEct model, recall that we can compute the average treatment effect on the treated (ATT) as $\mathrm{E}[\delta_{it}|Z_{it}=1]$, where $Z_{it}=1$ denotes the dummy variable representing being treated and 0 otherwise. The core of the IFEct method lies in imputing counterfactual outcomes for the treated observations. Liu et al. (2024) showed that the ATT can be written as a weighting estimator where each treated observation is matched with its predicted counterfactual: $\hat{Y}_{it}(0)$ and $\hat{\delta}_{it}=Y_{it}-\hat{Y}_{it}(0)$. The comparison within each set of matched observations eliminates the potential biases that can arise from improper weighting, which may affect TWFE estimates (see Web Section B of Liu et al. 2024 for proof).

To simplify the notation, let $\mathcal{D}_0=\{\{i,t\}|Z_{it}=0\}$ and $\mathcal{D}_1=\{\{i,t\}|Z_{it}=1\}$ denote the set of observations under the control condition (i.e., when AI is not enabled in our study) and treatment condition, respectively. Conceptually, the estimation strategy involves iteratively fitting the model on the two sets of observations, $\mathcal{D}_0$ and $\mathcal{D}_1$, in four steps:

- In step 1: we fit a model of the response surface Y_{it} on $\mathcal{D}_0$ (untreated observations), obtaining $\hat{g}$ and $\hat{u}$. Here, we rely on the linear function form assumption for *g(.)* and *u(.)* and the low-rank assumption of $\boldsymbol{U}$.
- In step 2: we predict the counterfactual outcome $Y_{it}(0)$ on $\mathcal{D}_1$ (treated observations) using $\hat{g}(X_{it})$ and $\hat{u}(U_{it})$ that we obtained from step 1: $\hat{Y}_{it}(0)=\hat{g}(X_{it})+\hat{u}(U_{it})$ for $(i,t)\in\mathcal{D}_1$.
- In step 3: based on steps 1 and 2, for each treated unit $(i,t)\in\mathcal{D}_1$, we compute the treatment effect for individual unit time $\hat{\delta}_{it}=Y_{it}-\hat{Y}_{it}(0)$.
- In step 4: the ATT is computed as the average of $\hat{\delta}_{it}$: $\widehat{ATT}=\frac{1}{|\mathcal{D}_1|}\sum_{\mathcal{D}_1}\hat{\delta}_{it}$.

Note that the above four steps are implemented through iterations. In iteration 0, for initialization, we let the two unobservables $\widehat{\lambda_\iota}^{(0)} = \widehat{f_t}^{(0)} = 0$ and fit a TWFE model to obtain initialized μ, α_i, τ_t , where $\alpha_i, \tau_t$are the unit and time fix effects and μ is the grand mean. Then, suppose in step 1 of iteration h we have the estimates $\widehat{\alpha_\iota}^{(h)}, \widehat{\tau_t}^{(h)}, \widehat{\lambda_\iota}^{(h)}, \widehat{f_t}^{(h)}, \hat{\beta}^{(h)}$. We can define the intermediate outcome $\dot{Y}_{it}^{(h)} = Y_{it} - \hat{\mu}^{(h)} - \widehat{\alpha_\iota}^{(h)} - \widehat{\tau_t}^{(h)} - (\widehat{\lambda_\iota}^{(h)})'\widehat{f_t}^{(h)}$ for $(i,t) \in \mathcal{D}_0$.

Next, in step 2, we update the coefficients of covariates $\hat{\beta}^{(h+1)} = \left(\sum_{(i,t)\in\mathcal{D}_0} X_{it}X_{it}'\right)^{-1} \sum_{(i,t)\in\mathcal{D}_0} X_{it}\dot{Y}_{it}^{(h)}$. Given the updated $\hat{\beta}$ in iteration $h+1$, we define this as follows:

$$W_{it}^{(h+1)} = \begin{cases} Y_{it} - X_{it}'\hat{\beta}^{(h+1)}, & (i,t) \in \mathcal{D}_0 \\ \hat{\mu}^{(h)} + \widehat{\alpha_\iota}^{(h)} + \widehat{\tau_t}^{(h)} - \left(\widehat{\lambda_\iota}^{(h)}\right)'\widehat{f_t}^{(h)}, & (i,t) \in \mathcal{D}_1 \end{cases}$$

On untreated observations $(i,t) \in \mathcal{D}_0$, $W_{it}^{(h+1)}$ can be directly computed. On treated observations, we instead compute its conditional expectation $E(W_{it}^{(h+1)}|\widehat{\lambda_\iota}^{(h)}, \widehat{f_t}^{(h)})$. We denote $W_{\cdot\cdot}^{(h+1)} = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T} W_{it}^{(h+1)}}{NT}$, $W_{i\cdot}^{(h+1)} = \frac{\sum_{i=1}^{N} W_{it}^{(h+1)}}{NT}$, and $W_{\cdot t}^{(h+1)} = \frac{\sum_{t=1}^{T} W_{it}^{(h+1)}}{NT}$ as averages over data (i,t) and define $\widetilde{W}_{it}^{(h+1)} = W_{it}^{(h+1)} - W_{i\cdot}^{(h+1)} - W_{\cdot t}^{(h+1)} + W_{\cdot\cdot}^{(h+1)}$.

The estimated unobservables $\widehat{\lambda_\iota}^{(h)}, \widehat{f_t}^{(h)}$ are updated by minimizing the following least squares objective function on data $W^{(h+1)} = \left[\widetilde{W}_{it}^{(h+1)}\right] \forall i,t$:

$$\left(\widetilde{\boldsymbol{\Lambda}}^{(\boldsymbol{h+1})}, \tilde{\mathbf{F}}^{(\boldsymbol{h+1})}\right) = \underset{(\widetilde{\boldsymbol{\Lambda}}, \tilde{\mathbf{F}})}{\operatorname{argmin}} \ \mathbf{tr}\left[(\boldsymbol{W}^{(\boldsymbol{h+1})} - \tilde{\mathbf{F}}\widetilde{\boldsymbol{\Lambda}}')'(\boldsymbol{W}^{(\boldsymbol{h+1})} - \tilde{\mathbf{F}}\widetilde{\boldsymbol{\Lambda}}')\right], \text{ s.t., } \frac{\tilde{\mathbf{F}}'\tilde{\mathbf{F}}}{T} = \mathbf{I}_r, \ \widetilde{\boldsymbol{\Lambda}}'\widetilde{\boldsymbol{\Lambda}} \text{ is diagonal.}$$

Here, tr[] indicates trace, where, for any matrix A, we write its norm as $\|A\| = (\mathrm{tr}(A'A))^{1/2}$. The identification of $\boldsymbol{\Lambda}, \mathbf{F}$ relies on a few restrictions imposed on the unobservables: $\sum_{i=1}^{N} \alpha_i = 0$, $\sum_{t=1}^{T} \tau_t = 0$, $\sum_{i=1}^{N} \lambda_i = 0$, and $\sum_{t=1}^{T} f_t = 0$.

Subsequently, in step 3 of iteration *h+1*, we update the estimates of the grand mean and the fixed effects: $\hat{\mu}^{(h+1)} = W_{\cdot\cdot}^{(h+1)}$, $\widehat{\alpha_i}^{(h+1)} = W_{i\cdot}^{(h+1)} - W_{\cdot\cdot}^{(h+1)}$, $\widehat{\tau_t}^{(h)} = W_{\cdot t}^{(h+1)} - W_{\cdot\cdot}^{(h+1)}$. We estimate the counterfactual for treated observations using the updated estimates: $\widehat{Y_{it}}(0) = X_{it}'\beta + \widehat{\alpha_i} + \widehat{\tau_t} + \lambda_i' f_t + \epsilon_{it}$ for $(i,t) \in \mathcal{D}_1$. Finally, in step 4 the estimated ATT is computed using the updated treated counterfactual outcome estimates.

B.3. IFEct Estimation Results on Aggregated Agent-Daily-Level Data

B.3.1. Main Effects of AI Assistance

To estimate the IFEct model, we first aggregate our data from the individual conversation level up to the agent-daily level. This means that, for each agent *i* and each day *t*, we construct aggregate variables by averaging across all conversations handled by *i* on day *t*. This aggregation ensures that there is no more than one observation per unit per period because, to the best of our knowledge, none of the existing HTE-robust estimators, including IFEct, can handle settings with multiple observations within the same unit in the same period. It is important to note that aggregating the data results in the loss of some information at the individual conversation level, which constrains the depth of our analysis. For instance, we were unable to include variables such as customer intent and types of chatbot involvement and to examine how they vary the effect of AI across different conversations. Nevertheless, the aggregated data enabled us to perform a robustness analysis using the HTE-robust estimator to examine the main treatment effect.

In the agent-daily-level data, we present the results of estimating the two main dependent variables.[2] The control variables that were included in the TWFE models, for example, agent tenure, were also controlled for in the IFEct models. In Figure S1, we plot the estimated treatment effect of AI suggestions on agent response time (Panel a) and customer sentiment improvement (Panel b). To examine potential temporal changes in the treatment effect and pre-treatment trends, we plot the estimated treatment effect for a 60-day period, 30 days before and 30 days after treatment assignment. The left vertical axis displays

[2] We used the *fect* software to implement the IFEct estimation and cross-validation to select the optimal number of loading factors λ.

the average treatment effect, as marked by small dots and 95% confidence intervals, while the right vertical axis shows the volume of observations for each period, represented by gray bars at the bottom.

First, we observe that the confidence intervals for the estimated effect across all pre-treatment periods intersect with zero, indicating no significant differences in trends between the treatment and control groups for the dependent variables. That is, the parallel trend assumption is verified. Second, the plots for the post-treatment periods show results consistent with our main findings obtained under a TWFE model. Specifically, there is a negative and significant impact of AI suggestions on agent response time (Panel [a]) and a positive and significant effect on customer sentiment improvement (Panel [b]). Notably, a visual examination of the estimated treatment effect, period by period, in Figure S1, also suggests that, upon receiving treatment, the agent immediately experienced a visible treatment effect, which remained consistent thereafter. That is, we did not find evidence for HTE over time.

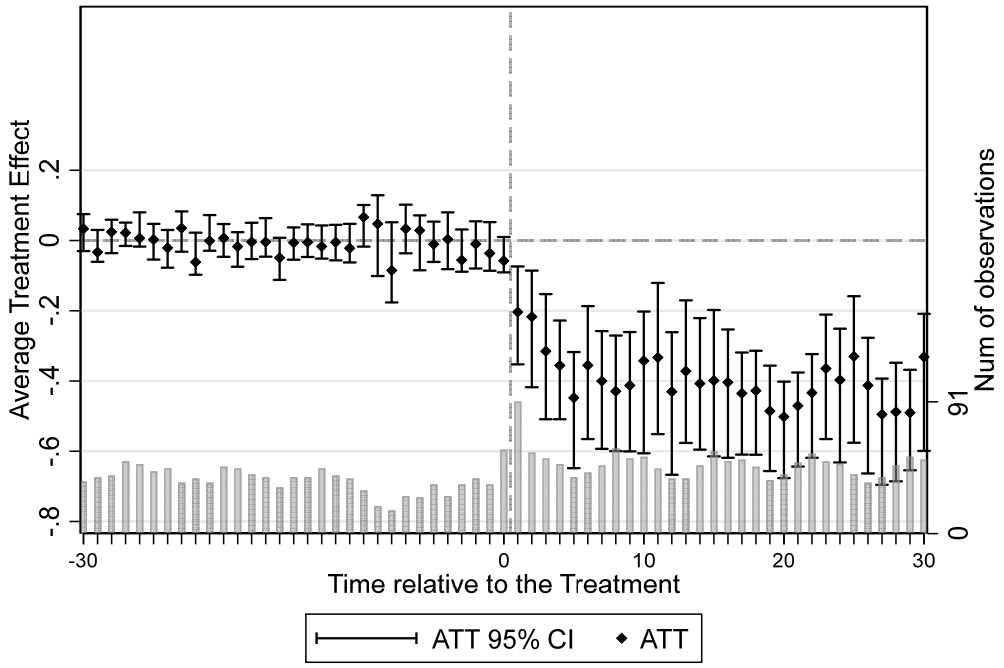
The results from estimating the IFEct model confirm that our main findings with the TWFE model are robust, despite the staggered setting. One explanation for this is that, under the strict exogeneity assumption, there were no significant temporal changes in the treatment effect, thus addressing the issue of negative weighting, which is caused by temporal heterogeneity in effects (Goodman-Bacon 2021). Although the TWFE estimator faces the risk of bias stemming from the negative weighting issue in the presence of staggered treatment, this does not automatically imply that it is biased in these cases. For instance, Zhang et al. (2023) analyzed longitudinal livestreaming data and found that the estimated effects of livestreaming on sales were very similar across different estimation methods (TWFE, synthetic DiD, and staggered DiD). In a thorough review, Chiu et al. (2023) replicated over 30 studies using observational panel data with binary treatments and compared six newly proposed HTE-robust estimators with the TWFE estimator; they concluded that these HTE-robust estimators generally do not change the main findings derived from TWFE estimation, even though there may be variations in precision of the results.

Although we lack full proof, it is possible that the following factors have contributed to the temporally invariant treatment effects: (1) The AI tool was simple to understand and use, (2) L.ai did not

update their AI model during our study, and (3) Company X did not alter their method of assigning conversations to agents throughout our study. Factor (1) implies that agents were unlikely to undergo a learning curve, where one would need time to gain experience and practice using the AI tool before fully leveraging its benefits. Factor (2) indicates that the model's performance remained constant; hence, the benefits of using AI suggestions did not vary over different time periods (this was confirmed with Company X and L.ai). Factor (3) means that Company X did not assign different conversations to agents based on the agents' access to AI, thus not affecting the degree to which one benefited from using AI (this was confirmed with Company X and is empirically verified in Web Appendix C).

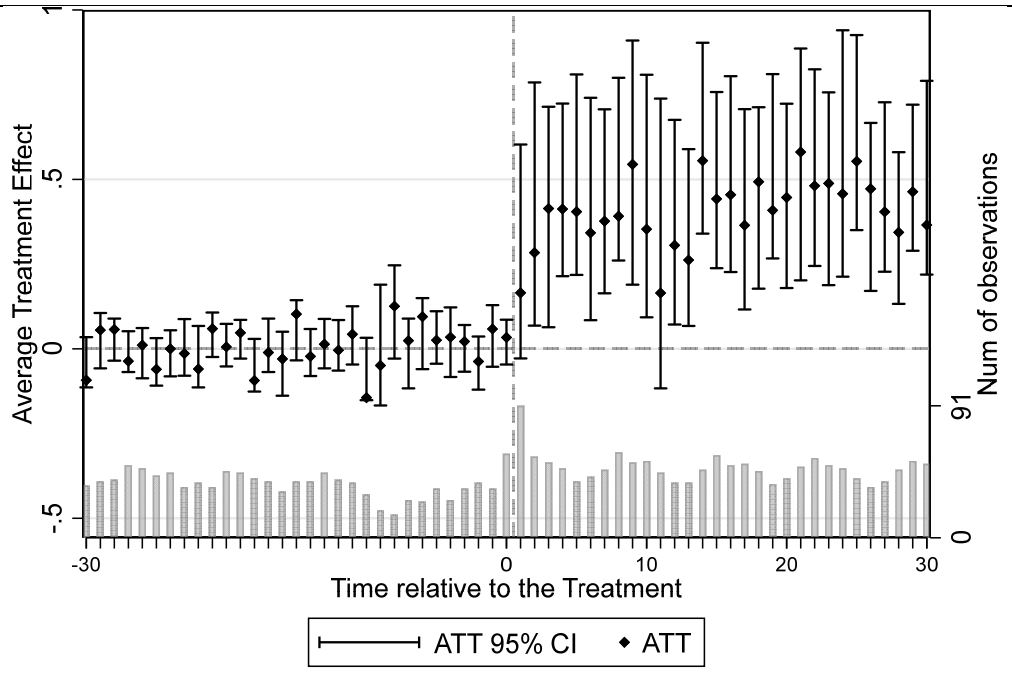
Figure S1. Robustness Analysis with the IFEct Estimator: Plotting the Treatment Effect of AI on the Key Outcome Variables

Panel (a). Plot of the effect of AI on *Agent Response Time* by time



The average effect of AI on agent response time: $b = -0.40$, $p < 0.001$.

Panel (b). Plot of the effect of AI on *Customer Sentiment* over time



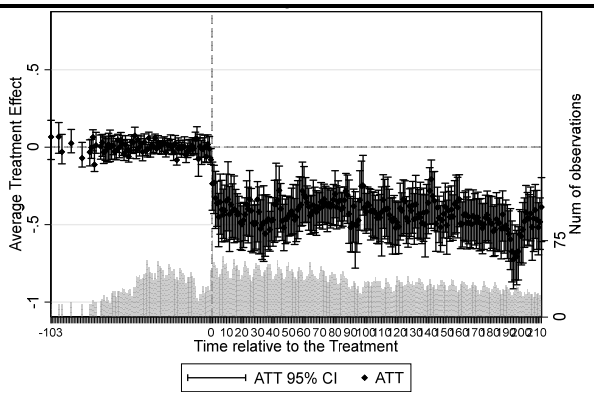
The average effect of AI on customer sentiment: $b = 0.42$, $p < 0.01$.

Notes. Estimated average treatment effect on treated of having access to AI on agent's response time to customer messages (Panel [a]) and on the improvement in customer sentiment (Panel [b]). The estimated treatment effects are plotted for each individual period *t*, which spans 60 days in total, from up to 30 days prior to treatment to up to 30 days following treatment. The gray bar denotes the number of units at the *t* period before and after treatment.

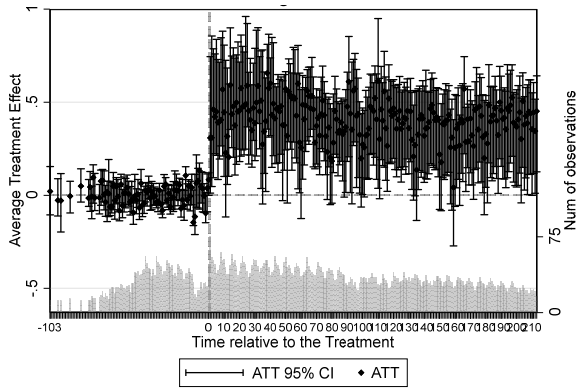
Furthermore, in Figure S2, we plotted the full time periods, instead of a 60-day window as in Figure S1, for agent response time (Panel a) and customer sentiment (Panel b). Throughout these periods, the estimated treatment effects remain relatively stable. However, the estimated effect of AI over longer periods is somewhat "noisy" because of the reduced number of data points (the sample size for each time period is indicated on the vertical axis on the right). This is because the number of observations in each extended period significantly decreases, weakening the statistical power. For instance, the first adoption cohort was on January 18, 2021. Thus, the number of units treated for 315 days (from the first adoption to the end of the data on November 30, 2021) would be limited to just this cohort. Therefore, to visualize the IFEct estimation results, we will focus on a 30-day period moving forward.

Figure S2 Plotting the Treatment Effect of AI on Outcome Measures: Full Time Panel

Panel (a). Plot of the effect of AI on *Agent Response Time* over time



Panel (b). Plot of the effect of AI on *Customer Sentiment* over time



Notes. Estimated average treatment effect on treated agents of having access to AI on agent's average response time to customer messages (Panel [a]) and on the customer sentiment (Panel [b]). The estimated treatment effects are plotted for each individual period *t*. The gray bar denotes the number of units at the *t* period before and after treatment.

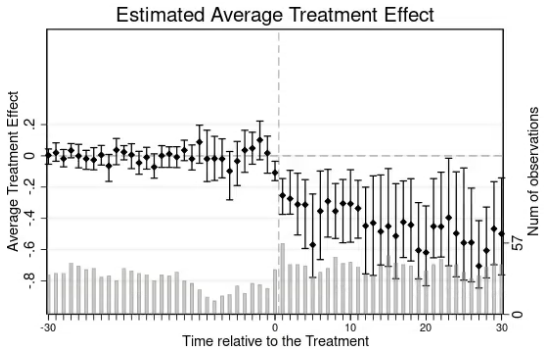
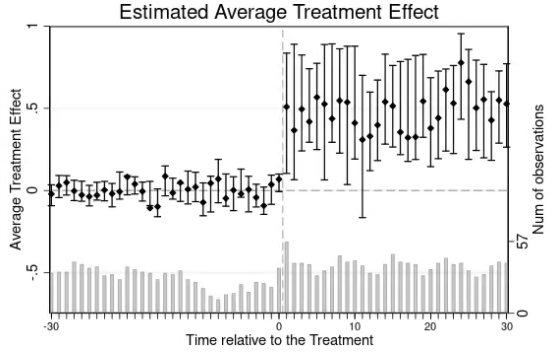
B.3.2. Heterogeneous Effects of AI Assistance

In addition to the main effects, the IFEct analysis also helps us assess the robustness of the heterogeneity results (see Section 5). Since the IFEct model does not have straightforward methods for interactions, we split the data and performed the IFEct analyses separately. For the treatment interaction with agent tenure, a continuous variable, we divided the data by the sample median (median agent tenure = 190 days) into "longer-tenured agents" and "shorter-tenured agents" subsets. Figure S3 and Figure S4 plot the treatment effects of AI on agents with tenure shorter than the median and on those with tenure longer than the median, respectively. The two figures confirm the robustness of the TWFE heterogeneity results: the IFEct estimation showed a larger effect of AI on reducing average response time for shorter-tenured agents ($b = -0.44$, $p < 0.001$; Panel a of Figure S3) than for longer-tenured agents ($b = -0.33$, $p < 0.05$; Panel a of Figure

S4) and a larger effect of AI on improving customer sentiment for shorter-tenured agents (b = 0.48, p < 0.001; Panel b of Figure S3) than for longer-tenured agents (b = 0.29, p > 0.1; Panel b of Figure S4).

We do not perform IFEct analyses for the TWFE interactions with customer intent and chatbot error. This is because IFEct is conducted on aggregated samples, with all conversations combined at the agent-day level. Aggregation is not an issue for agent tenure, because the variable is measured at the day–agent level. In contrast, customer intent and chatbot error were measured at the conversation level, posing a challenge for customer intent and chatbot error.

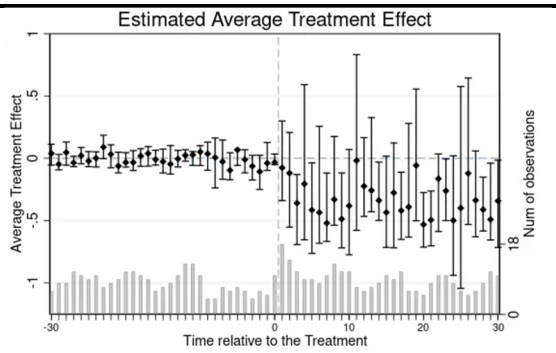
Figure S3. Plotting the Treatment Effect of AI: Impact on Agents with Shorter Tenures

Panel (a). *Agent Response Time*:	**Panel (b). *Customer Sentiment***
	
The average effect of AI on agent response time: $b = -0.44$, $p < 0.001$.	The average effect of AI on customer sentiment: $b = 0.48$, $p < 0.001$.

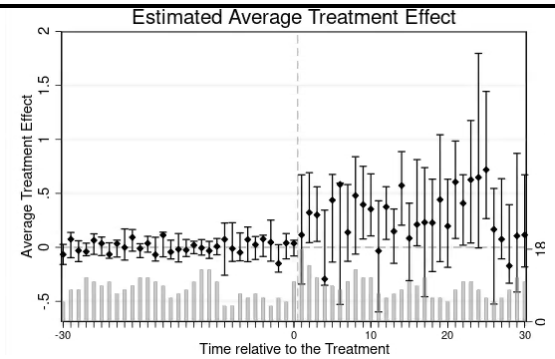
Notes. Estimated average treatment effect on treated of having access to AI on agent's response time to customer messages (Panel [a]) and on the improvement in customer sentiment (Panel [b]). The left and right columns plot the IFEct estimation results on a subsample: agents with tenure shorter than the median (which is 190 days in data). These results illustrate how the estimated effects of AI on key outcome variables interact with agent tenure. The estimated treatment effects are plotted for each individual period *t*, which spans 60 days in total, from up to 30 days prior to treatment to up to 30 days following the treatment. The gray bar denotes the number of units at the *t* period before and after treatment.

Figure S4. Plotting the Treatment Effect of AI: Impact on Agents with Longer Tenures

Panel (a). *Agent Response Time*:	**Panel (b). *Customer Sentiment***



The average effect of AI on agent response time: $b = -0.33$, $p < 0.05$.



The average effect of AI on customer sentiment: $b = 0.29$, $p > 0.1$.

Notes. Estimated average treatment effect on treated of having access to AI on agent's response time to customer messages (Panel [a]) and on the improvement in customer sentiment (Panel [b]). The left and right columns plot the IFEct estimation results on a subsample: agents with tenure longer than the median (which is 190 days in data). These results illustrate how the estimated effects of AI on key outcome variables interact with agent tenure. The estimated treatment effects are plotted for each individual period *t*, which spans 60 days in total, from up to 30 days prior to treatment to up to 30 days following the treatment. The gray bar denotes the number of units at the *t* period before and after treatment.

Appendix C: Assignment of Conversations to Agents in the Presence of AI Treatment

Our main findings indicate a notable improvement in the effectiveness of customer–agent interactions for agents equipped with AI. However, a concern arises that this improvement might be influenced by changes in the types of conversations assigned to agents with AI support rather than AI's effectiveness alone. For example, if Company X systematically assigned fewer "harder conversations" to agents once they had access to AI, these changes could inflate the perceived impact of AI. To address this concern, we first confirmed with Company X that they did not alter the distribution of online conversation tasks among agents regarding AI assignment. Next, we performed a robustness analysis to provide empirical support. We examined task assignment variations for agents pre- and post-AI adoption, which helped isolate AI's true impact on productivity by ensuring that our findings would not be confounded by changes in task complexity or type.

In Table S2, we present the results of estimating five models. We regressed a set of conversation characteristics on the control variables, including agent tenure, agent and day fixed effects, and whether agents had access to AI. In Panel (a), Model (1) used *Start Sentiment* as the dependent variable and tested whether agents were assigned conversations with different levels of initial customer sentiment, which indicated the ease of a conversation, after they had AI. Model (2) used *Bot Not Understand* as the dependent variable and tested whether agents, after they had AI, received more or fewer conversations in which the customer experienced an instance of the chatbot comprehension issue. As suggested by the insignificant coefficients of *AI Enabled* in columns 1 and 2 of Panel (a), having access to AI suggestions did not alter the difficulty or chatbot-related issues of conversations assigned to agents.

Moving to Panel (b), the models used *Customer Intent* as the dependent variable and tested whether agents were assigned customers with different intents after they had AI. In predicting three types of customer intents—*Cancel Subscription*, *Repeat Complaint*, and *Non-Complaint*—we found that the coefficients of *AI Enabled* were statistically insignificant across all three models, suggesting that the company did not vary the types of customer intents distributed to agents before and after AI assignment.

Taken together, the results from Table S2 rule out the alternative explanation of the company assigning different types of conversations to agents based on whether their AI treatment status. This strengthens the robustness of our primary finding: The improvement in agent's work performance was not confounded by systematic changes in task assignments. Instead, it was attributable to AI-enabled suggestions, which assisted agents in managing customer conversations with increased efficiency and effectiveness.

Table S2. Assessing the Types of Conversations Assigned to Agents Before and After AI Treatment

Panel (a): Customer start sentiment and chatbot issues of assigned conversations

VARIABLES	**ESTIMATES (Robust Std. Err.)**			
	(1) Start Sentiment		(2) Bot Not Understand	
AI Enabled	–0.0168	(0.0175)	0.00639	(0.00432)
Fixed Effect	Agent, Daily		Agent, Daily	
Observations	256934		256934	

Panel (b): Types of customer intents of assigned conversations

VARIABLES	**ESTIMATES (Robust Std. Err.)**					
	(1) Cancel Subscription		(2) Repeat Complaint		(3) Non-Complaint	
AI Enabled	0.00119	(0.0470)	–0.0275	(0.0367)	0.0334	(0.0349)
Fixed Effect	Agent, Daily		Agent, Daily		Agent, Daily	
Observations	162419		162419		162419	

Notes. The unit of observation was per conversation. The models in Panel (a) tested whether having access to AI suggestions (*AI Enabled* = 1) affected two key characteristics of conversations that an agent was assigned: the start sentiment of a customer message (column 1) and whether the customer experienced a chatbot comprehension issue (column 2). The models in Panel (b) tested whether having access to AI suggestions (*AI Enabled* = 1) affected the types of customer intents assigned to a chat agent: canceling subscription, repeat complaint, or non-complaint inquiries. Agent tenure, agent fixed effects, and day fixed effects were controlled for across all five models. Robust standard errors are clustered at the agent level and reported in parentheses. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Appendix D: When Were Agents More Likely to Adopt AI-Suggested Replies?

We conducted additional analyses to enrich our understanding of how AI might have shaped the behavior of chat agents. Specifically, we investigated when agents complied more, in terms of using an AI-suggested reply message. We computed two variables that capture the compliance level of agents: *Adoption_Rate* and *Num_Adopted_Sentences*. *Adoption_Rate* is computed as the ratio of AI suggestions used by an agent, and *Num_Adopted_Sentences* was computed as the total number of sentences of AI-generated suggestions that were adopted by an agent. Note that we followed a strict definition of "adoption": An AI-suggested reply is "adopted" if and only if the suggestion was *exactly* included in the agent's response message. For instance, if AI suggested "How can I help you?" and if the agent's next reply to the customer included exactly "How can I help you?" then this would be considered an adoption. However, if there were revisions or changes, such as "What can I do for you?" or "Is there anything I can help you with?" in an agent's next reply, these would not count as an adoption, despite having similar meanings to the AI-suggested message.

As can be seen from Table S3, the coefficient of *Num. of Conversation in the Day* (indicating the total number of conversations that agent *i* had completed on day *i* by the time of conversation *j*) is positive and significant across columns 1 and 2, suggesting that, as agents engage in more conversations within a day, their likelihood to follow an AI suggestion increased. This may be because agents feel increasingly fatigued as the day progresses, leading them to rely more on AI to lighten their workload.

Next, consistent with our anticipation, agents were less likely to follow AI suggestions when the customer had just experienced an issue with the chatbot being unable to comprehend human messages, as suggested by the negative and significant coefficients of *Bot Not Understand* across the models. Two reasons can explain this effect. First, seeing the presence of the *Bot Not Understand* issue makes an agent less confident in the performance of the AI model assisting them and, thus, less likely to rely on AI suggestions to interact with the customer. Second, agents know that the customer may be averse to machines after they experience the *Bot Not Understand* issue and, hence, may strategically choose not to use AI suggestions. The two explanations, despite leading to the same observation, have different managerial

implications. The first explanation suggests that companies should address algorithmic aversion problems not just for customers but also for agents. For instance, when these issues arise, companies can display a message reminding agents of AI's superior performance, for example, by showing the agent's improvement with AI over the past few days.

The second explanation, however, suggests that companies should focus on addressing the algorithmic aversion problem for their customers; for example, they could strategically increase the response time of AI-generated suggestions (see Section 5.2). Alternatively, when the "*Bot Not Understand*" issue arises, companies can present a message to the customer to ensure that they understand that, in the customer–agent conversation, AI does not automate and replace a human agent; instead, AI is used only to offer suggestions and augment the human agent. Upon addressing a customer's algorithmic aversion problem, to increase an agent's compliance with AI, companies can show the agent that the AI suggestions have been adjusted to cater to the needs of the focal customer, hence minimizing the customer's aversion to AI. Subject to data limitations, we were unable to determine which explanation was driving the results. Yet this should be a question that companies keep in mind and examine if they want to maximize the effectiveness of AI and enhance the adoption of AI among targeted users.

Table S3. Chat Agent's Compliance with AI Suggestions

VARIABLES	**ESTIMATES (Robust Std. Err.)**			
	(1) Adoption Rate		(2) Number of Adopted Sentences	
End Customer Sentiment of Last Conversation	0.00161*	(0.000723)	0.000629	(0.00532)
log Response Time of Last Conversation	0.000388	(0.00236)	–0.00599	(0.0116)
log Tenure	–0.217	(0.240)	–0.778	(0.919)
Num. of Conversation in the Day	0.000538***	(0.000130)	0.00369***	(0.000926)
Bot Not Understand	–0.0183***	(0.00253)	–0.0597***	(0.0109)
Customer Intent: Cancel Subscription	0.0273***	(0.00391)	0.125***	(0.0254)
Customer Intent: Non-Complaint	–0.0466***	(0.00358)	–0.0898***	(0.0172)
Customer Intent: Repeat Complaint (baseline: Others)	–0.0160	(0.00880)	–0.0756	(0.0596)

Constant	1.306	(1.172)	4.583	(4.485)
Fixed Effect	Agent, Daily		Agent, Daily	
Observations	138860		138860	
Within R-squared	0.096		0.023	

Notes. The unit of observation was per conversation. Models (1) and (2) assessed the factors that predicted the extent to which chat agents complied with AI suggestions. The compliance level was captured by the percentage of AI suggestions that were used (Model (1)) and the number of sentences of AI suggestions that were used (Model (2)) in an agent's subsequent responses to the customer. The outcome variables of the two models are computed against AI-enabled suggestions. Robust standard errors are clustered at the agent level and reported in parentheses.
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Appendix E: Robustness Checks and Additional Analyses

E.1. Using the Difference in Customer Sentiment as D.V.: Impact of AI on Improving Customer Sentiment

In our main analysis, when analyzing customer sentiment, we use the sentiment of the last message as an outcome variable and control for the sentiment of the first message. Doing so allows us to assess how AI affected customer sentiment at the end of the conversation compared with the beginning of the conversation. Here, we conducted a robustness check, where we directly analyzed the *change* in customer sentiment. That is, the dependent variable now is *Change_in_Sentiment = End_Sentiment – Start_Sentiment*. We have replicated the analysis found in Section 4.1 and report the results in Table S4. As can be seen, the positive coefficient of *AI Enabled* indicates that, by the end of the conversation, AI led to more improvement in customer sentiment than human effort alone ($b = 0.464$, $p < 0.001$). Thus, using the difference in customer sentiment as D.V., we obtained results that are consistent with the main results.

Table S4 Robustness Test: Main Effects of AI on Customer Sentiment

VARIABLES	**ESTIMATES (Robust Std. Err.)**	
	(1)	
	Change in Customer Sentiment	
AI Enabled	0.464***	(0.0456)
log Tenure	0.276***	(0.0547)
Constant	–0.432*	(0.200)
Fixed Effect	Agent, Daily	
Observations	256934	
Within R-squared	0.024	

Notes. The unit of observation was per conversation. The D.V. was the difference in customer sentiment between the last and first message. The model examined the effect of AI on changing customer sentiment within a conversation. Robust standard errors are clustered at the agent level and reported in parentheses. * p < 0.05, ** p < 0.01, *** p < 0.001

E.2. Testing Violation of SUTVA via Shared Workload: Did Effects of AI Change as More Agents Had Access to It?

In the main paper, we clarified how customer inquiries are routed in chat agents. Specifically, for Company X's online chat customer support, the routing algorithm operates by assigning incoming customer inquiries to any available chat agent. This approach ensures that all customers receive timely assistance and that customer queries are evenly distributed among the customer support team to maintain efficient service levels across the board. That is, the routine algorithm should not be affected by the assignment status of AI, which affected agent performance. The results from Section C of this appendix further confirm that the status of AI assignment for one agent should not affect the performance of other agents who did not have AI assistance, via the routing of different types of conversations among agents, that is, no spillover across treated agents versus nontreated agents.

However, one may still wonder whether there could be spillover effects through the conversation-routing algorithm that assigns different workloads to agents. For instance, because chat agents use the same routing algorithm, treated agents who complete tasks more quickly may be more likely to become available to handle a new customer. This could influence the number of customers routed to nontreated agents, potentially affecting both treated and control units, hence violating the stable unit treatment value assumption (SUTVA).

While SUTVA cannot be directly tested empirically, we indirectly assessed the potential for SUTVA violation through workload analysis. Treated agents, who used AI and responded to customer messages more quickly, may have "shared" the workload that would have otherwise been distributed to nontreated agents. Specifically, we estimated the following model in Equation (E.1):

$$Y_{ijt} = Constant + \delta \cdot AI\_Enabled_{ijt} + \beta \cdot Num.Treated_t + \eta \cdot \left(AI\_Enabled_{ijt} \cdot Num.Treated_t\right) + \lambda \cdot X_{ijt} + \rho_i + \gamma_t + \varepsilon_{ijt,} \quad \text{(E.1)}$$

where Y_{ijt} indicates an outcome variable of interest for conversation *j* that was handled by agent *i* at day *t*, and $Num.Treated_t$ refers to the number of agents who were already treated by day *t* and who worked on the chat channel on that day. X_{ijt} captures conversation characteristics, and γ_t and ρ_i are the day and agent fixed effects, respectively. Note that β captures whether agent *i*'s outcome is affected as more and more agents were treated, and ρ captures whether such an effect is different for the treated versus the nontreated agents.

The logic of this analysis is as follows: If SUTVA were violated, we would expect the spillover from treated to nontreated agents to be relatively limited at the start when there were few treated agents. For instance, if only one agent had received the AI treatment and completed more tasks, the additional workload handled by this agent would have a minimal impact on each nontreated agent when distributed among the remaining nontreated agents. Conversely, we would expect the spillover effect to be more significant with more treated agents. Therefore, if SUTVA were not violated, we would observe that the number of treated agents did not affect an agent's outcome; that is, β and η would be insignificant.

In Table S5, we report the estimated results from regressing our main outcome variables, *Average Response Time* and *Customer Sentiment* in columns 1 and 2, respectively. As can be seen from column 1, estimated β and η are statistically insignificant, suggesting that the response speeds of both treated and nontreated agents were unaffected by increased access to AI among agents. Similarly in column 2, we find that estimated β and η are statistically insignificant again, suggesting that the improvement in customer sentiment, here conditional on AI adoption and other factors, was not influenced by the number of treated agents. Although not definitive, Table S5 does not provide support that SUTVA was violated through treated agents taking on more work and affecting both treated and nontreated agents, thereby reinforcing our confidence in the causal impacts of AI on chat agents.

Table S5 Testing SUTVA: Impact of AI on Response Time and Customer Sentiment with Varying Number of Treated Chat Agents

VARIABLES	ESTIMATES (Robust Std. Err.)

	(1) Average Response Time		(2) Customer Sentiment	
AI Enabled	–0.258**	(0.0920)	0.526***	(0.145)
Start Sentiment			0.242***	(0.00798)
Num.Treated	–0.0000875	(0.00135)	–0.00243	(0.00161)
log Tenure	–0.245***	(0.0355)	0.258***	(0.0476)
AI Enabled X Num.Treated	0.000143	(0.00154)	–0.00144	(0.00272)
Constant	4.214***	(0.154)	–0.844***	(0.164)
Fixed Effect	Agent, Daily		Agent, Daily	
Observations	256934		256934	
Within R-squared	0.186		0.074	

Notes. The unit of observation was per conversation. We used two measures of chat agent productivity: *Average Response Time*, which reflects the agent's responsiveness to each customer message in a conversation (Column 1; log-transformed), and *Customer Sentiment*, which assesses the effect of AI on customer sentiment at the end of the conversation, controlling for sentiment at the beginning of the conversation (Column 2). *Num.Treated* refers to the number of chat agents that were active on the chat channel and who had obtained access to AI by the time, which allows us to examine how AI effects on the two outcome measures changed as there were more treated agents who might have affected the outcome measures for the untreated agents via shared workload. Standard errors are clustered at the agent level and reported in parentheses. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

E.3. Testing Violation of the No Anticipation Assumption: Did Effects of AI Vary as Agents Possibly Spread Word about The Treatment Over Time?

One of the key identification assumptions, as described in Section 2.5.1 of the main manuscript, is that the exact future treatment adoption date (the adoption is not realized yet) for an agent does not affect the potential outcome in the current period. This assumption—the *No Anticipation* assumption—is violated when the treatment path is prior knowledge and, hence, the units "anticipate" a treatment.

As we discussed, this assumption is likely to hold in our setting because Company X did not broadcast the rollout of AI in the chat channel and instead communicated privately with the to-be-treated agents only the day before the exact adoption date. Communication among agents was minimal because they worked remotely and did not share a physical workspace. As a result, agents would not have prior knowledge about the treatment path and, thus, would not anticipate treatment adoption. In this section, we provide suggestive empirical evidence that possible spillover, if any, was likely to be very limited; that is, the *No Anticipation* assumption was likely not violated.

We conducted a robustness test by assessing how the impact of AI in assisting chat agents changed over time. The logic of this analysis is similar to the SUTVA test in Equation (E.1): If agents talked to each

other and as a result agents "expected" treatment in the chat channel, then word must have been *initially* spread from treated agents to nontreated agents and then possibly from anyone to anyone. There are two consequences of such "word-of-mouth": (1) the agents who were in the first batch or cohort to be treated should not be affected by the spillover, and (2) though we do not know the speed of people spreading word, the impact of the spillover on the remaining nontreated agents likely would increase as time goes by because the likelihood that any nontreated agent heard about the treatment would increase as people started talking about it. Equation (E.2) tests possible spillovers:

$$Y_{ijt} = Constant + \delta \cdot AI\_Enabled_{ijt} + \beta \cdot FirstTreatWeek_t + \eta \cdot \left(AI\_Enabled_{ijt} \cdot FirstTreatWeek_t\right) + \lambda \cdot X_{ijt} + \rho_i + \gamma_t + \varepsilon_{ijt,} \quad \text{(E.2)}$$

where we interacted *AI_Enabled* with a time indicator $FirstTreatWeek_t$ to test whether the effects of AI on *Y* changed as time went by. Specifically, $FirstTreatWeek_t$ equals 1 if time *t* was in the first week following the first adoption, that is January 18, 2021, to January 24, 2021. The idea follows the logic that the spillover, within a short period of time after the first batch of agents were notified of AI, has not yet happened quickly. Thus, η captures whether agents behaved differently later than initially in the field experiment. For spillover to be negligible, we expect η to be insignificant.

In columns 1 and 2 of Table S6, we present the results from estimating *Average Response Time*, and *Customer Sentiment*, respectively. The results suggest that, across columns 1 and 2, there are similar effects of AI with the main results: AI reduced agent response time (the coefficient of *AI Enabled* is –0.251, $p < 0.001$) and improved customer sentiment (the coefficient of *AI Enabled* is 0.452, $p < 0.001$). Importantly, the estimated coefficients of *AI Enabled X FirstTreatWeek* are statistically insignificant at the 95% significance level across two columns, suggesting that, between the first week and the later weeks, there was no difference in the AI effect on agent response time or customer sentiment.

Recall that, if there were spillover, we would expect the anticipation assumption (*Assumption 2*) to be more likely violated—or violated to a greater extent—as the experiment progressed and as more agents

experienced AI. Therefore, the observed effects of AI, as influenced by agents' adjusted behavior in anticipation of treatment, would significantly differ between the later and earlier stages of the experiment. However, the results in Table S6 highlight that the impact of AI on agent behavior was immediate, following the first adoption on January 18, and remained consistent over time. While not full proof, the robustness test does not imply the presence of a spillover effect via agents spreading words about the AI treatment.

Table S6 Testing Spillover: Impact of AI Treatment by Time

VARIABLES	ESTIMATES (Robust Std. Err.)			
	(1)		(2)	
	Average Response Time		Customer Sentiment	
AI Enabled	–0.251***	(0.0412)	0.452***	(0.0397)
Start Sentiment			0.242***	(0.00797)
log Tenure	–0.245***	(0.0355)	0.257***	(0.0476)
AI Enabled X FirstTreatWeek	0.0115	(0.0386)	–0.00901	(0.0719)
Constant	4.213***	(0.154)	–0.842***	(0.164)
Fixed Effect	Agent, Daily		Agent, Daily	
Observations	256934		256934	
Within R-squared	0.186		0.074	

Notes. The unit of observation was per conversation. We used two measures of chat agent productivity: *Average Response Time*, which reflects the agent's responsiveness to each customer message in a conversation (Column 1; log-transformed), and *Customer Sentiment*, which assesses the effect of AI on customer sentiment at the end of the conversation, here controlling for sentiment at the beginning of the conversation (Column 2). *FirstTreatWeek* equals 1 if time *t* was in the first week following the first adoption, that is, January 18, 2021, to January 24, 2021, and examines how AI played a role in assisting chat agents in the first week of AI assignment, when the possible "word-of-mouth" among the agents was limited. Standard errors are clustered at the agent level and reported in parentheses. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

E.4. Further Discussion of the TWFE Results

We present supporting evidence, in addition to the IFEct results in Section B, that verify the TWFE results. First, we follow de Chaisemartin and D'Haultfœuille (2020) to compute a ratio metric. In "Two-Way Fixed Effects Estimators with Heterogeneous Treatment Effects," de Chaisemartin and D'Haultfœuille (2020) showed that the TWFE estimate is a weighted average of treatment effects across different cohorts and periods. They pointed out that some weights can be negative if the treatment effect varies across cohorts or overtime, which can result in an estimate with an incorrect sign, that is, the negative weighting issue that

might arise in a staggered setting, as discussed in Section 2.5. They further proposed a metric to help researchers identify and assess the severity of negative weighting in a specific dataset/context: the ratio of the expected coefficient, $\hat{\beta}_{fe}$, divided by the standard deviation of the weights, $\sigma(\omega)$. They also showed that this ratio, $\underline{\sigma}_{fe}$, equals the minimal value of $\sigma(\Delta)$—the standard deviation of the average treatment effects (ATEs) across cohort-period cells, where the average treatment effect on the treated (ATT) might have an opposite sign to the coefficient. Specifically, the ratio $\underline{\sigma}_{fe}$ is expressed below (Corollary 1, de Chaisemartin & D'Haultfœuille, 2020):

$$\underline{\sigma}_{fe} = \frac{|\hat{\beta}_{fe}|}{\sigma(\omega)},$$

There is no rule of thumb for what value of $\underline{\sigma}_{fe}$ is "good enough." However, generally speaking, if $\underline{\sigma}_{fe}$ is larger, it is safer to conclude that the TWFE estimation results are robust (against the heterogeneity treatment effect and negative weighting issues). This is because it implies that the heterogeneity across cohorts and times, as captured by $\sigma(\Delta)$, must be more substantial to cause the TWFE estimate $\hat{\beta}_{fe}$ to have an incorrect sign (or to be nullified) than the ATT. In contrast, if this ratio is very small, the TWFE estimate and ATT can have opposite signs, even with a small and plausible amount of treatment effect heterogeneity, indicating a significant validity concern for the TWFE estimate $\hat{\beta}_{fe}$.

We followed the approach described by de Chaisemartin and D'Haultfœuille (2020)[3] and below report two pieces of evidence that support the empirical validity of our TWFE results.

First, we consistently obtained large values of $\underline{\sigma}_{fe}$. For one main dependent variable, average response time, we obtained $\underline{\sigma}_{fe} = 0.18$. In our context, this suggests that ATT and $\hat{\beta}_{fe}$ may be of opposite signs if the standard deviation of treatment effects across agent-day observations exceeds 0.18 (de Chaisemartin and D'Haultfœuille 2020). Given that the estimated TWFE coefficient on average response time was $\hat{\beta}_{fe}$= -0.250 (Table 2 of the manuscript), this implies an implausibly high amount of heterogeneity.

[3] We used STATA package *twowayfeweights*, which was developed by de Chaisemartin and D'Haultfoeuille (2020).

To illustrate, following Bena et al. (2021), if the ATTs are assumed to be drawn from a uniform distribution with zero mean and standard deviation $\underline{\sigma}_{fe}$, the ATTs should be distributed within the $[-\sqrt{3}\underline{\sigma}_{fe}, \sqrt{3}\underline{\sigma}_{fe}]$ interval. In our case, we have $|\hat{\beta}_{fe}| < \sqrt{3}\underline{\sigma}_{fe}$ ($\sqrt{3}$*0.18 = 0.31), suggesting that $\underline{\sigma}_{fe}$ is implausibly high level of treatment effect heterogeneity. A similar analysis and interpretation can be applied to the other main outcome variable, customer sentiment, for which we obtained $\underline{\sigma}_{fe}$ = 0.29. Because the estimated TWFE coefficient on customer sentiment was $\hat{\beta}_{fe}$= 0.451 (Table 5 of the manuscript), $\underline{\sigma}_{fe}$ is implausibly substantial amount of heterogeneity because $|\hat{\beta}_{fe}| < \sqrt{3}\underline{\sigma}_{fe}$ ($\sqrt{3}$*0.29 =0.50).

Second, the weights obtained from the above analyses did not correlate with the treatment effects. de Chaisemartin and D'Haultfœuille (2020) suggest calculating the correlation between the weights and a predictor of the treatment effect: A strong correlation indicates that the TWFE estimate may be biased, while a lack of correlation suggests that $\hat{\beta}_{fe}$ may still be robust to treatment effect heterogeneity even with a small estimated ratio $\underline{\sigma}_{fe}$ (Assumption 7 and Corollary 2 of de Chaisemartin and D'Haultfœuille 2020). For average response time, agent tenure is likely a predictor of treatment effects because agents with more experience benefit less from the help of AI. We correlated the weights with agent tenure and found a very weak correlation (corr. = -0.158, *t* stats = -0.86). Similarly, for customer sentiment, we correlated weights with two predictors of the treatment effect: agent tenure and the sentiment of the first customer message (*Start Sentiment*). We found that the weights were not correlated with agent tenure (corr. = -0.156, t-stat = -0.85) or with *Start Sentiment* (corr. = -0.015, t-stat = 0.07).

Furthermore, we implemented TWFE estimations by treatment adoption cohort over time. A cohort refers to a group of agents assigned the AI tool (i.e., treatment adoption) on a given day; for example, cohort 47 means the agents were assigned AI on day 47 (January 18, 2021, counting from day 1, December 1, 2020). Agents that adopted AI on the same day were considered to be part of the same cohort and identified with a unique cohort ID.

The logic of implementing this by-cohort robustness test is as follows: The concern with TWFE in a staggered setting arises from HTE, potentially leading to the negative weighting issue when earlier-treated units are used as a control group for not-yet-treated units. To address this, we implemented TWFE estimations by each cohort and time, interacting the treatment indicator (AI_Enabled) with cohort and time dummies, similar to the approach of Sun and Abraham (2021).[4] We then averaged all cohort-time specific estimates and obtained consistent results: With estimates averaged across all cohort-time and clustered at the cohort level, we found that the average treatment effect of AI on agent's average response time was -0.18 (s.e. = 0.052), confirming that AI suggestions reduced agent response time. In addition, the estimated effect of AI on customer sentiment was 0.28 (s.e. = 0.097), suggesting that AI-enabled agents significantly improved customer sentiment to a greater extent. This indicates that the cohort-based analysis led to consistent results with the TWFE analyses. In fact, our robustness analysis using the IFEct method, which allowed us to obtain an HTE-robust estimator on the agent-day aggregated data, essentially implements TWFE by cohort and time: The IFEct estimate is computed as weighted average of individual ATT for each unit and time (Liu et al. 2024). As shown in Web Appendix B, the IFEct analysis yielded similar conclusions as our main TWFE analysis.

In summary, the large ratio of the TWFE coefficient to the standard deviation of the weights, the lack of correlation between weights and treatment effect predictors, the cohort-based TWFE estimation, and the IFEct results confirm the robustness of our TWFE estimation results.

[4] We used R model *fixest* for the estimation.

E.5. Exploring the Reasons for Heterogeneity in AI's Impact on Customer Sentiment Across Different Customer Intents

First, we incorporated the classified types of agent responses, $Type\_Agent\_Messages_{ijt}$, into the intent heterogeneity model to explore why AI might be more beneficial for certain customer intents. We estimated the Equation (E.3):

$$EndSentiment_{ijt} = Constant + StartSentiment_{ijt} + \delta \cdot AI\_Enabled_{it} + \eta \cdot \left(AI\_Enabled_{it} * Customer\_Intent_{ijt}\right) + \lambda_1 \cdot X_{ijt} + \lambda_2 \cdot Type\_Agent\_Messages_{ijt} + \rho_i + \gamma_t + \varepsilon_{ijt}. \quad \text{(E.3)}$$

Our focus is on how η changes when accounting for the types of agent responses. If the differential effects of AI on improving customer sentiment across various customer intents can be explained by the variation in empathy, solution, and information levels, we expect η to weaken or decrease after controlling for these three response types.

The estimation results are presented in column 2 of Table S7, with column 1 replicating the results from estimating Equation (4). As can be seen, as compared with the main model in column 1, the impact of AI on customer sentiment in column 2 significantly changed. Consistent with Section 4.3, the main effect of AI-enabled suggestions substantially decreased after accounting for empathy-based, information-based, and solution-based responses, indicating that these response message characteristics explain much of the improvement in customer sentiment. Notably, the coefficients for the *AI Enabled* × *Customer Intent* interaction terms became insignificant in column 2, suggesting that the differential effects of AI across customer intent categories, as shown in column 1, were fully explained by the three agent response dimensions: empathy, information, and solution. In other words, AI-assisted agents tailored their responses across these three dimensions, and these differences accounted for the variation in AI's impacts on customer sentiment across different customer intents.

In Figure S7 (Section G.1 of this appendix), we present, for each customer intent category (*Cancel Subscription, Non-Complaint,* and *Repeat Complaint*), the average levels of empathy (Panel a), information (Panel b), and solution (Panel c) in messages from agents without AI suggestions (indicated as 0 on the x-axis) and with AI suggestions (indicated as 1 on the x-axis). We observed a significant increase in empathy-based, information-based, and solution-based responses in *Cancel Subscription* conversations when AI was enabled. Importantly, the increase in solution-based responses was particularly significant compared with other customer intent categories. This suggests that, in improving the sentiment of customers who requested to cancel their subscription, AI played a crucial role by enhancing the three key response types, particularly by increasing solution-based responses. For example, consider a customer inquiry to cancel their subscription from our sample: with AI assistance, the agent was informed that the customer's reason for cancellation was because of "personal dietary restrictions and preferences." Recognizing that this issue could be addressed, AI suggested that the agent offer an alternative subscription plan with low-fat options rather than a standard cancellation or discount. As a result, customer sentiment improved, not only because of the agent's empathetic and timely responses but also because of the tailored solution that aligned with the customer's specific dietary and fitness needs.

Incorporating agent response types into the analysis also provided insights into why AI was less effective in *Repeat Complaint* conversations. As shown in Figure S7, the AI-enabled responses in these conversations scored significantly higher on the information dimension (Panel b), slightly higher on the empathy dimension (Panel a), but were comparable to non-AI responses in terms of solution-based responses (Panel c). The substantial increase in information-based responses when AI was enabled might be because of the recurring nature of these complaints, allowing AI to extract relevant customer and intent data from the database, which may explain why *Repeat Complaint* conversations saw the largest reduction in agent response time. However, the improvement in customer sentiment for *Repeat Complaint* inquiries was smaller compared with *Cancel Subscription* and *Non-Complaint* inquiries, possibly because of the lack of enhancement in solution-based responses. This may be because the solutions to these issues were often

beyond the AI's capabilities because repeat complaints typically arise from systemic problems. For instance, in cases where a customer repeatedly received incorrect deliveries, AI could use database information to identify that the issue stemmed from the opening of a new fulfillment center and guide the agent accordingly. Despite offering credits, the root issue remained unresolved. The customer remarked that this was "not a solution" and inquired about the company's plan to address the underlying systemic problem.

Table S7 Impact of AI Suggestions on Customer Sentiment across Customer Intents, Controlling for Response Types

	ESTIMATES (Robust Std. Err.)			
	(1)		(2)	
VARIABLES	**Customer Sentiment: Main Model (also column 2 of Table 8)**		**Customer Sentiment: Controlling for Response Types**	
AI Enabled (baseline: Others)	0.321***	(0.0287)	0.106***	(0.0280)
Start Sentiment	0.217***	(0.00708)	0.336***	(0.00739)
log Tenure	0.263***	(0.0482)	0.197***	(0.0532)
Customer Intent (baseline: Others)				
Cancel Subscription	–0.0644*	(0.0280)	-0.0914**	(0.0300)
Non-Complaint	–0.0649**	(0.0238)	-0.113***	(0.0216)
Repeat Complaint	–0.194	(0.111)	-0.336**	(0.101)
AI Enabled × Customer Intent (baseline: Others)				
AI Enabled × Cancel Subscription	0.0909**	(0.0284)	0.0502	(0.0324)
AI Enabled × Non-Complaint	0.0842**	(0.0255)	0.0308	(0.0218)
AI Enabled × Repeat Complaint	–0.0640	(0.108)	-0.153	(0.102)
Empathy-based Response			0.378***	(0.0162)
Information-based Response			0.270***	(0.00827)
Solution-based Response			0.0757***	(0.00844)
Constant	0.393	(0.279)	-0.0621	(0.242)
Fixed Effect	Agent, Daily		Agent, Daily	
Observations	162419		162419	
Within R-squared	0.031		0.276	

Notes. The models were estimated based on individual chats that had an available customer intent label; the sample sizes for the models differ from those discussed in Sections 3 and 4 because of some conversations lacking a customer intent label. Model (1) examined customer sentiment at the end of the conversation, controlling for sentiment at the beginning of the conversation, as moderated by customer intent. Model (1) is a replicate of column 2 of Table 8. Model (2) replicates Model (1) while controlling for three types of agent responses, standardized for easier interpretation. Robust standard errors are clustered at the agent level and reported in parentheses. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

E.6. AI Improved Customer Sentiment More in Challenging Conversations: Ruling Out the "Ceiling Effect"

In Section 4.2 (Table 6, column 2), we found that the coefficient of the *AI_Enabled* × *StartSentiment* interaction was negative and significant, indicating that AI was more effective in guiding agents through

more challenging conversations. We believe this was unlikely to be driven by a "ceiling effect," where conversations starting with lower sentiment have more room for improvement. First, this potential for improvement exists in both the treatment (with AI) and control (without AI) conditions.

Second, we conducted a robustness test, and the results suggested that the "ceiling effect" was unlikely the main factor. As shown in Equation (E.4), the test replicated the analysis from Section 4.2, excluding conversations with very high *Start Sentiment*.

$$EndSentiment_{ijt} = Constant + StartSentiment_{ijt} + \delta \cdot AI\_Enabled_{it} + \eta \cdot \left(AI\_Enabled_{it} * StartSentiment_{ijt}\right) + \lambda \cdot X_{ijt} + \rho_i + \gamma_t + \varepsilon_{it} \; . \quad \text{(E.4.)}$$

Specifically, we first computed the average change in customer sentiment (from *Start Sentiment* to *End Sentiment*) for conversations where AI was enabled, which was 1.17. We then removed all conversations with *Start Sentiment* > (2 - 1.17) = 0.83 and estimated the above model. The idea is as follows: If the ceiling effect—where conversations starting with higher sentiment have less room for improvement—was driving the results in Table 6, then excluding the conversations likely to "hit the ceiling" would weaken or even nullify the coefficient of *AI_Enabled* × *StartSentiment*. However, as shown in Table S8, we found that the coefficient of the interaction term remains negative and significant, suggesting that the ceiling effect was not the driver behind it.

Table S8 Heterogeneous Effects of AI on Customer Sentiment: Interaction with Start Sentiment: Testing the Possibility of Ceiling Effect

VARIABLES	**ESTIMATES (Robust Std. Err.)**	
	(1)	
	Interaction with Start Sentiment	
AI Enabled	0.310***	(0.0360)
Start Sentiment	0.347***	(0.0272)
log Tenure	0.268***	(0.0492)
AI Enabled × Start Sentiment	-0.220***	(0.0296)
Constant	-0.821***	(0.169)
Fixed Effect	Agent, Daily	
Observations	249011	

Within R-squared	0.068

Notes. The unit of observation was per conversation. Model (1) analyzed moderation by the start sentiment of the customer message (level of difficulty of a customer conversation) on a subsample. The subsample excluded the conversations where *Start Sentiment* > 0.83, which was the average improvement in customer sentiment when AI was enabled. The idea is that these conversations, with a relatively low sentiment to start with, were unlikely to hit the "ceiling"—the maximum sentiment of a customer message was 2. Robust standard errors are clustered at the agent level and reported in parentheses. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Appendix F: Text Analysis of Agent Responses

The main paper documents that agents who were assisted with AI-enabled suggestions improved customer sentiment to a greater extent than those who did not have access to AI suggestions and that the benefits from AI suggestions were greater for certain types of customer intents. Section 4.3 and Web Appendix E.5 further suggest that the effects of AI, and their variations, were partly explained by AI-assisted agents using different types of messages that improved customer sentiment. The difference in these message characteristics might work particularly well for certain customer inquiries or intents. Analyzing agent message text data thus helped us understand the drivers of AI's impact on customer sentiment and examine how agent responses influence customer perceptions and behaviors. This section details the text analysis.

F.1. Description of The Classification Task and The Text Data

The dataset comprises all the messages from chat agents (who may or may not have access to AI-enabled suggestions). We hypothesize that the presence of empathetic greetings and/or responses, information, and solutions within messages may significantly impact these effects. Thus, each agent message is evaluated to predict three binary variables:

- **Empathy-based**: Whether the message conveys empathy.
- **Information-based**: Whether the message shares information.
- **Solution-based**: Whether the message provides a solution.

These variables are not mutually exclusive, meaning that a single message could simultaneously express empathy, provide information, and offer a solution. For example, a response could apologize for an issue and legitimize the customer's feeling (empathy), explain the situation (information), and resolve the problem (solution). Examples of model classifications of agent responses are provided in Section F.5 of this appendix.

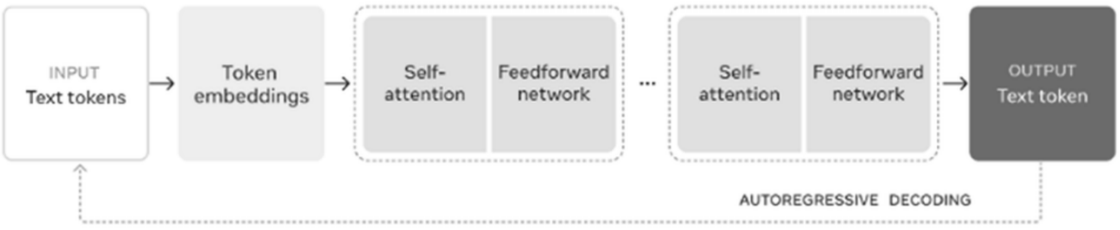
F.2. Choice of Language Model: Llama

Given the size of the dataset, manual labeling was impractical. Instead, we employed large language models (LLMs) to automatically categorize the messages. We selected Meta-Llama-3.1-8B-Instruct (which is the instruction tuned text only variant of Llama 3.1, with 8 billion parameters) for its ability to perform offline inference, mitigating potential data leakage risks that might arise with public APIs (e.g., ChatGPT API).

Figure S5 shows the architecture of the Llama 3.1 model (Meta AI 2024). Llama 3.1 represents a collection of advanced open-source large language models (LLMs) introduced by Meta AI, supporting multimodal inputs across text, image, and speech. It is designed to handle a diverse array of tasks, including text completion, coding, reasoning, and math problem-solving. The Llama 3.1 model is based on a standard decoder-only transformer architecture and was pre-trained on an extensive corpus of web-scraped data collected through the end of 2023. Its training dataset consists of a blend of token types: 50% general knowledge, 25% focused on mathematics and reasoning, 17% code, and 8% multilingual tokens (Dubey et al., 2024). Llama 3.1 has been widely applied across various fields, with Accenture leveraging it for ESG reporting, Goldman Sachs utilizing it for document information extraction (Meta AI, 2024), and researchers fine-tuning Llama for financial sentiment analysis (Konstantinidis et al., 2024).

Furthermore, our computational resources included access to up to five NVIDIA Tesla V100 GPUs, which provided significant processing power. However, the complexity of distributed learning configurations posed challenges. As a result, a model with fewer parameters, like Llama3, was preferable for streamlining the process.

Figure S5 Llama 3 Model Architecture



F.3. Prompt Optimization and Model Fine-tuning

In LLM tasks, a *prompt* refers to the input or set of instructions given to the model to guide its output. It usually includes a question, command, or example that informs the model about the desired response. The prompt is essential because it sets the context, scope, and clarity of the task, which directly influences the quality and relevance of the model's answer. Prompt optimization involves refining the language, structure, or examples within the prompt to improve the model's performance. The aim of this process is to boost the model's accuracy, consistency, and clarity by helping it better understand the task, especially when dealing with complex or nuanced instructions.

The prompt for our task was optimized over three rounds. In each round, the revised prompt was used to label a new batch of messages. After each round, 200 messages were randomly selected, and their labels were reviewed by two independent researchers to evaluate the effectiveness of the prompt revisions. Our prompt followed a structured design, incorporating elements of "instruction," "context," "input data," and "output indicator" (Giray 2024). An illustration of the final prompt structure is provided in Section F.4 of this appendix.

Based on instances of incorrect labeling, we refined the prompt according to Meta AI's guidelines,[5] incorporating practices such as explicit instructions on formatting and constraints, as well as adjusting examples to achieve more accurate and consistent outputs. This approach, as described by Brown et al. (2024), is termed "few-shot prompting." The prompts used in each round are presented in Table S9.

In the first round, we identified several issues with the model labeling:

- Short messages were difficult to label accurately because of limited context, making it hard for the model to assign appropriate labels.
- Some messages were incorrectly labeled as solution-based, simply because they contained words like "help" or requested personal information (e.g., "Is there anything else I can assist you with?"

[5] "Prompting: How-to guides." https://www.llama.com/docs/how-to-guides/prompting/. Accessed: Oct. 19, 2024.

or “First, can I please have the name and phone number associated with your subscription?”), without offering any specific solutions.

- The model was lenient in labeling messages as “empathetic,” often tagging casual greetings or general communication that lacked true empathy.
- For longer texts, the model occasionally failed to label them as “information based” or “solution based.”

To address these issues, we revised the prompt by adding more examples in the instructions (using a *few-shot* mechanism that could provide examples for the model to reason and follow) and included notes at the end to address the identified problems.

In the second round, further evaluation of another 200 randomly selected messages revealed the following:

- Generic greeting phrases like “thank you for reaching out” should not be labeled as empathetic because they are often template responses and do not reflect the agent’s genuine empathy.
- However, phrases that acknowledge or legitimize customer concerns, such as “I understand how frustrating it must be that your order is delayed,” per our conversations with L.ai, can be considered empathetic.

We further refined the examples in the prompt (the *few-shot* mechanism), adjusting the labels based on the identified issues with empathy. This revised prompt was used for the third round of labeling. In this round, we identified the following issues:

- Some messages were (incorrectly) labeled as information-based solely because they requested personal information from clients.
- Llama occasionally mislabeled messages as empathetic, often tagging casual greetings or routine interactions as empathetic, even when these did not fully convey empathy.

Thus, in this round of prompt optimization, examples were adjusted to emphasize positive indicators for information-based messages. Definitions for empathy-based and information-based messages were also refined to address Llama's misinterpretations from the previous round.

Lastly, after reviewing another randomly selected 200 messages, we were satisfied with the model's performance. As a result, we finalized the prompt by incorporating examples of the common mistakes identified in the most recent evaluation.

Table S9 Prompts Used in Each Round of Fine-Tuning for Llama Model Classification of Agent Responses

Fine-tuning Round	Prompt
1	We have a dataset consisting of multi-turn conversations between customers and customer service agents. \\ You will be presented with the text of the conversation(only the text, no other instruction).\\ You should determine if: first, whether the text has empathetic greetings and responses(more emotional);\\ second, whether the text is information-based (the text gives any information, as opposed to persuasive or empathetic language);\\ third, whether the text is solution/resolution based (that is, does the agent respond to the customer with any solution to the customer's complaints or inquiries).\\ For each category, you should label the agent's response as 'yes' or 'no', and also return your confidence level(from 0 to 1) in the prediction.\\ For example, if you are given 'I'm so sorry but it seems the "Everything" Baked *"Everything" Baked" Chicken is not available for this week.', then your return should be:\\ yes, 0.9, yes, 0.8, no, 0.7.\\ Another example, if you are given 'Please toss these meals out since they are no longer fresh. I am now processing a full refund for you.', then your return should be:\\ no, 0.7, no, 0.4, yes, 0.9.\\ Another example, if you are given 'You may need to click on the link in that email so that it will place an order.', then your return should be:\\ no, 0.3, no, 0.6, yes, 0.7.\\ Another example, if you are given 'Hi Bowen Thank you for reaching out to [Company X]!', then your return should be:\\ yes, 0.9, no, 0.8, no, 0.8.\\ Another example, if you are given 'Yes, you can cancel anytime.', then your return should be:\\ no, 0.9, no, 1.0, yes, 0.8.\\ Another example, if you are given '"I totally understand that you would like the lower amount I sincerely apologize since the price of that is depend also on how many meals plan they have. Since the box prices will have the same prices on the gift card. I sincerelly apologize since the price there on the website is how many meals you plan to choose, 4, 6, 10, 12 meal plan', then your return should be:\\ yes, 0.7, yes, 1.0, no, 0.8.\\ Learn from these examples, especially the labels!!!!! But you can your own confidence level.\\ You should ONLY ONLY ONLY return the labels and confidential levels for the three categories, in the order of empathetic greetings, information-based, solution-based,\\ and in the form of 'yes/no, confidence, yes/no, confidence, yes/no, confidence'.\\ don't answer any other questions, don't provide any other information, don't ask any questions.
2	We have a dataset consisting of multi-turn conversations between customers and customer service agents. \\ You will be presented with the text of the conversation(only the text, no other instruction).\\ You should determine if: first, whether the text has empathetic greetings and responses(more emotional);\\ second, whether the text is information-based (the text gives any information, as opposed to persuasive or empathetic language);\\

	third, whether the text is solution/resolution based (that is, does the agent respond to the customer with any solution to the customer's complaints or inquiries).\\ For each category, you should label the agent's response as 'yes' or 'no', and also return your confidence level(from 0 to 1) in the prediction.\\ For example, if you are given 'I'm so sorry but it seems the "Everything" Baked *"Everything" Baked" Chicken is not available for this week.', then your return should be:\\ yes, 0.9, yes, 0.8, no, 0.7.\\ Another example, if you are given 'Please toss these meals out since they are no longer fresh. I am now processing a full refund for you.', then your return should be:\\ no, 0.7, no, 0.4, yes, 0.9.\\ Another example, if you are given 'Got it..If I may ask, just to verify is this for this week only or all future orders please?', then your return should be:\\ yes, 0.6, no, 0.8, no, 0.8.\\ Another example, if you are given 'Apologies, [NAME]. Your delivery was due tomorrow, not today. Your delivery day was set to [DATE_TIME]. \\ Just a friendly reminder, we are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription.\\ Your weekly deadline to skip weeks, cancel, or edit your meals would be every Wednesday before [DATE_TIME] EDT.', then your return should be:\\ yes, 0.7, yes, 0.9, no, 0.8.\\ Another example, if you are given 'Got it. Please allow me a moment while I review your subscription.', then your return should be:\\ no, 0.7, no, 0.8, no, 0.8.\\ Another example, if you are given '"I totally understand that you would like the lower amount I sincerely apologize since the price of that is depend also on how many meals plan they have.\\ Since the box prices will have the same prices on the gift card.\\ I sincerely apologize since the price there on the website is how many meals you plan to choose, 4, 6, 10, 12 meal plan', then your return should be:\\ yes, 0.7, yes, 1.0, no, 0.8.\\ Learn from these examples, especially the labels!!!!! But you can determine your own confidence level.\\ You should ONLY ONLY ONLY return the labels and confidential levels for the three categories, in the order of empathetic greetings, information-based, solution-based,\\ and in the form of 'yes/no, confidence, yes/no, confidence, yes/no, confidence'.\\ don't answer any other questions, don't provide any other information, don't ask any questions. Note: Simple greetings and responses like 'Thank you' or 'Got it' are not empathetic enough, it should be more friendly over the conversation.\\ Also, asking customers for information or to do something (like giving a phone number or waiting) are not qualified as solution-based responses.\\ And, if the agent's response is a long text, it is more likely that the agent's response is information-based, but not always.\\
3	We have a dataset consisting of multi-turn conversations between customers and customer service agents. \\ You will be presented with the text of the conversation (only the text, no other instruction).\\ You should determine if: first, whether the text has empathetic greetings and responses (more emotional);\\ second, whether the text is information-based (the text gives rather than ask for any information, as opposed to persuasive or empathetic language);\\ third, whether the text is solution/resolution based (that is, does the agent respond to the customer with any solution to the customer's complaints or inquiries).\\ For each category, you should label the agent's response as 'yes' or 'no', and also return your confidence level (from 0 to 1) in the prediction.\\ For example, if you are given 'I'm so sorry but it seems the "Everything" Baked *"Everything" Baked" Chicken is not available for this week.', then your return should be:\\ yes, 0.7, yes, 0.8, no, 0.7.\\ Another example, if you are given 'Thank you. Let me check that here. One moment.', then your return should be:\\ yes, 0.9, no, 0.8, no, 0.8.\\ Another example, if you are given 'Please toss these meals out since they are no longer fresh. I am now processing a full refund for you.', then your return should be:\\ no, 0.7, no, 0.4, yes, 0.9.\\ Another example, if you are given 'Got it..If I may ask, just to verify is this for this week only or all future orders please?', then your return should be:\\ no, 0.6, no, 0.8, no, 0.8.\\

	Another example, if you are given 'Apologies, [NAME]. Your delivery was due tomorrow, not today. Your delivery day was set to [DATE_TIME]. \\ Just a friendly reminder, we are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription.\\ Your weekly deadline to skip weeks, cancel, or edit your meals would be every Wednesday before [DATE_TIME] EDT.', then your return should be:\\ no, 0.7, yes, 0.9, no, 0.8.\\ Another example, if you are given 'I have canceled the order and applied a $66.71 refund to your payment method. Please allow up to five business days for this to post to your bank account, depending on your bank/card issuer's processing time.', then your return should be:\\ no, 0.9, yes, 0.9, yes, 0.9.\\ Another example, if you are given '"I totally understand that you would like the lower amount I sincerely apologize since the price of that is depend also on how many meals plan they have.\\ Since the box prices will have the same prices on the gift card.\\ I sincerely apologize since the price there on the website is how many meals you plan to choose, 4, 6, 10, 12 meal plan', then your return should be:\\ yes, 0.7, yes, 1.0, no, 0.8.\\ Another example, if you are given 'We'll make sure you will still get your discount. I understand your concern.', then your return should be:\\ no, 0.7, no, 0.9, no, 0.6.\\ Learn from these examples, especially the labels!!!!! But you can determine your own confidence level.\\ You should ONLY ONLY ONLY return the labels and confidential levels for the three categories, in the order of empathetic greetings, information-based, solution-based,\\ and in the form of 'yes/no, confidence, yes/no, confidence, yes/no, confidence.'. No additional columns or data should be included.\\ Don't answer the text given to you, don't provide any other information, don't ask any questions, just return the labels and confidence levels.\\ Note: Simple greetings and responses like 'Thank you' or 'Got it' are not empathetic enough, they are too generic and may be adapted from templates, it should address more context around the issue, for example,\\ 'I am sorry to hear that your order did not arrive on time.'\\ Also, asking customers for information or to do something (like giving a phone number or waiting) are not qualified as solution-based responses.\\ And, if the agent's response is a long text, it is more likely that the agent's response is information-based, but not always.\\

F.4. Final Model for Text Classification Analysis and Computing Metrics

Table S10 presents the finalized prompt used in the model to classify agent messages across the entire dataset. The model initially utilized five NVIDIA Tesla V100 GPUs, which was later reduced to four GPUs, with memory usage ranging from 1505 MB to 2215 MB at its peak.

Given the large size of the message dataset, we divided it into five subsets and performed text classification on each subset. This approach allowed us to identify and resolve computational issues, such as memory overflows, during the classification of subsequent sets. The model processed 12 messages per

batch.[6] The computation times, results, and any errors or adjustments made during the classification of these five subsets are detailed in Table S11.

Table S10 The Final Prompt Used in the Llama Model for Categorizing Agent Responses

Prompt[7]
We have a dataset consisting of multi-turn conversations between customers and customer service agents. \\ You will be presented with the text of the conversation (only the text, no other instruction).\\ You should determine if: first, whether the text has empathetic greetings and responses (more emotional);\\ second, whether the text is information-based (the text gives rather than ask for any information, as opposed to persuasive or empathetic language);\\ third, whether the text is solution/resolution based (that is, does the agent respond to the customer with any solution to the customer's complaints or inquiries).\\ For each category, you should label the agent's response as 'yes' or 'no', and also return your confidence level (from 0 to 1) in the prediction.\\ For example, if you are given 'I'm so sorry but it seems the "Everything" Baked *"Everything" Baked" Chicken is not available for this week.', then your return should be:\\ yes, 0.7, yes, 0.8, no, 0.7.\\ Another example, if you are given 'Thank you. Let me check that here. One moment.', then your return should be:\\ yes, 0.9, no, 0.8, no, 0.8.\\ Another example, if you are given 'Please toss these meals out since they are no longer fresh. I am now processing a full refund for you.', then your return should be:\\ no, 0.7, no, 0.4, yes, 0.9.\\ Another example, if you are given 'Got it..If I may ask, just to verify is this for this week only or all future orders please?', then your return should be:\\ no, 0.6, no, 0.8, no, 0.8.\\ Another example, if you are given 'Apologies, [NAME]. Your delivery was due tomorrow, not today. Your delivery day was set to [DATE_TIME]. \\ Just a friendly reminder, we are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription.\\ Your weekly deadline to skip weeks, cancel, or edit your meals would be every Wednesday before [DATE_TIME] EDT.', then your return should be:\\ no, 0.7, yes, 0.9, no, 0.8.\\ Another example, if you are given 'I have canceled the order and applied a $66.71 refund to your payment method. Please allow up to five business days for this to post to your bank account, depending on your bank/card issuer's processing time.', then your return should be:\\ no, 0.9, yes, 0.9, yes, 0.9.\\ Another example, if you are given '"I totally understand that you would like the lower amount I sincerely apologize since the price of that is depend also on how many meals plan they have.\\ Since the box prices will have the same prices on the gift card.\\ I sincerelly apologize since the price there on the website is how many meals you plan to choose, 4, 6, 10, 12 meal plan', then your return should be:\\ yes, 0.7, yes, 1.0, no, 0.8.\\ Another example, if you are given 'We'll make sure you will still get your discount. I understand your concern.', then your return should be:\\ no, 0.7, no, 0.9, no, 0.6.\\ Learn from these examples, especially the labels!!!!! But you can determine your own confidence level.\\ You should ONLY ONLY ONLY return the labels and confidential levels for the three categories, in the order of empathetic greetings, information-based, solution-based,\\ and in the form of 'yes/no, confidence, yes/no, confidence, yes/no, confidence.'.\\

[6] The batch size is limited by CUDA's memory capacity. Although a batch size of 20 is feasible in most cases, it may exceed memory limits when handling longer texts. To avoid this issue, a batch size of 12 was determined to be the safest option.

[7] For easing the reading and understanding, we color-coded the prompt. Here "instruction" is marked as green, "input data" is marked red (it also includes the message later sent for annotation), and "output indicator" is marked blue. The rest (black font) is "context".

don't answer any other questions, don't provide any other information, don't ask any questions.\\
Note: Simple greetings and responses like 'Thank you' or 'Got it' are not empathetic enough, they are too generic and may be adapted from templates, it should address more context around the issue, for example,\\
'I am sorry to hear that your order did not arrive on time.'\\
Also, asking customers for information or to do something (like giving a phone number or waiting) are not qualified as solution-based responses.\\
And, if the agent's response is a long text, it is more likely that the agent's response is information-based, but not always.\\

Table S11 Computation Metrics for Message Classification

Subset of Data	Total Time (hours)	Total Time (seconds)	Execution Error/Adjustment
First	31.84	114,626	CUDA exceeds memory (reduced batch size to 12)
Second	126.46	455,242	None
Third	38.42	138,324	Reduced GPU from 5 to 4 then (extra GPU needed for image tasks)
Fourth	96.77	348,365	None
Fifth	88.64	319,086	None
Total	382.12	1,375,643	

F.5. Examples of Model Classification: Three Types of Agent Messages

Each message was classified into three types of messages, here based on whether the message conveys empathy, whether the message shares factual information, and whether the message provides a solution. In Table S12, we offer examples of each classification to highlight what kind of message content is evaluated by the Llama model as empathy-based, information-based, or solution-based. The message examples are separated by dashed lines.

Table S12 Examples of Agent Responses and Their Classified Categories by the Llama Model

Agent Message (Examples Randomly Selected Across Conversations)	Llama 3 Model Classification Output		
	Is Empathy-based	Is Information-based	Is Solution-based
One moment while I pull up your subscription to look into this for you! For verification purposes, can I have your phone number associated with your subscription? --- Hi there! You've reached [Company X]'s chat service! It looks like there are a few people ahead of you in our chat queue. We'll get to you as soon as possible! --- I see. One moment. You have these 2 meals Zingy Buffalo Chicken Bowl	No	No	No

Zingy Buffalo Chicken --- Thank you so much. And lastly, may I have your full name, please --- I'm afraid that you will not be able to update your [DATE_TIME] order on your end as it is already past the deadline, [NAME]. But I can still make changes on my end. Let me request someone from our phone team to call you, [NAME].			
I'm afraid we do not have any vegetarian or vegan dishes at this time. We hope to offer these in the near future, however! --- Hi [NAME], thanks for reaching out. My apologies for the inconvenience this caused you. I will definitely do my best to help you today! Can you please verify the phone number associated with your subscription? --- Hello, [NAME]. Thanks for reaching out to [Company X]. This is Coral and I'm here to help you with your concern. I sincerely apologize for the inconvenience that this has caused you. For me to better assist you, could you please verify the phone number associated with the subscription?	Yes	No	No
Thank you. One moment, while I pull up your subscription. I can confirm for you that your subscription has been canceled. You will not have any additional payments unless you choose to reactivate your subscription in the future. If you do decide that you would like to rejoin us sometime in the future, I've included a couple of links to show you how to reactivate your subscription below: How do I reactivate my subscription? - [website] How do I reactivate my subscription in the app? - [website] You will receive an email confirmation shortly. --- By the way, just a quick reminder we are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription. Your weekly deadline to skip weeks, cancel, or edit your meals would be every Friday before [DATE_TIME] ET! Thank you! --- Sometimes, due to human error, scans are not updated. --- Sure! We have delivery dates to choose from once you create a subscription. You can change them to what dates are available. Since you are considering [Company X], I would love to take this time to tell you a little more about our service. We are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription. We will notify you of the deadline via email and it is also listed in your "Deliveries", on the upper left-hand side when viewing a particular week! --- Yes the meals are fresh, and gluten-free. We add one new meal to our menu almost every week. This would either involve adding in a completely new meal or revamping a pre-existing meal to make it even more delicious.	No	Yes	No
Got it, [NAME]. Please do toss the meals out if you no longer feel safe eating them. I will send a follow-up email about the actions I have done for you on my end. --- No worries! Thank you as well.			

I have now processed the cancelation of your order. I have requested for the meals to not be delivered, but please know that should they arrive, please be reminded that the meals were already refunded. The meals are on us. Please know as well that I have applied a $67.10 refund to your payment method. Please allow up to five business days for this to post to your bank account, depending on your bank/card issuer's processing time! --- I've now cancelled your subscription for you, so you should not receive any further charges or deliveries from us unless you log back in and reactivate your subscription. --- One moment here. Thanks for waiting! I've already changed your scheduled delivery to Monday, starting [DATE_TIME]. For the next 3 weeks, your scheduled delivery will still the same which is Sunday. I hope this helps. Do you have further questions for me aside from this? --- Thank you for reaching out to [Company X]! Upon checking, it appears that your subscription is already canceled. Since we do not have a direct option to skip few months of deliveries. Canceling the subscription temporarily would be the best option you have.	No	No	Yes
I'm so sorry for this one. If the day is fully booked and not available we cant move or change the current delivery day. I totally understand that you need these meals. Again I'm so sorry for this one, [NAME].	Yes	No	Yes
If incase there is a need to order again from [Company X]. Reactivating your subscription is quite simple! Here is a link with the brief instructions if you decide to reactivate in the future: [URL]- My apologies for the experience, I will make sure this incident will get reported to the appropriate team. --- I'm afraid we are not able to send out a replacement order, as it takes several days for our facility to prepare and cook meals for shipment. The soonest we can send out another order would be for your next scheduled delivery day. Rest assured that this has been reported to the appropriate department for further investigation	Yes	Yes	No
Hi [Name]! Thank you for reaching out to [Company X]! We do not design our meals to comply with any specific medical diets like high blood pressure, heart healthy, or others. If you have any specific health concerns, we always recommend consulting your physician or nutritionist to determine if our meals are the best fit for you! If we want to add more, I highly recommend adding a new subscription, [Name]. Here is how to add another subscription to your account: [website] --- We have a wide variety of carb content on our menu, averaging about 40 grams. By clicking on the pictures of the meals on our menu, you will find that several have tags that indicate if a meal has fewer than 35 grams of carbs. When you have an active subscription, you can sort by different nutritional values, from lowest to highest, including sodium, carbs, protein, fat, and calories. This will be found at the top of the "Change Meals" page for each of your weekly orders. This feature is useful if you have a goal in mind. However, we recommend consulting your physician or nutritionist if you have any specific health concerns.			

Just as a friendly FYI, we are a weekly subscription service, meaning we will charge your card and send you deliveries of delicious meals every week unless you skip a week or cancel your subscription. Your weekly deadline to make meal changes or skipping the week is every Thursday before 4:00 PM CT. Is there anything else I can help you with? --- I see. Thanks for letting me know. I highly recommend creating a link here: [website] The discount will be applied for Thu, Dec 10th and 17th weekly orders. --- You're welcome! I just want to inform you that we are a weekly subscription service and we deliver our meals once per week - meaning, you would receive meals automatically (and charged for) meals on a weekly basis, and they would be delivered on your chosen available delivery day. The good news is, you do not have to commit to receiving weekly meals. It's super easy to make changes to the meals you receive and skip weeks. You just need to do so by your weekly deadline. I hope this helps.	No	Yes	Yes
I have applied $62.93 in-store credit to your subscription. This is a one-time credit which will be applied to your next charged order and will remain active if unused for six months from today's date. I truly understand [Name], I know how important these meals are to you parents. And the earliest delivery we have will be on Monday, December 28th. Yes, you can have one subscription. All you need is change the address. Please know that we do have weekly deadlines to make changes --- Hello [NAME]! Thank you for reaching out to [Company X]! We recommend following these steps for reheating: 1. Do not thaw or defrost. 2. Add a couple extra minutes to the listed heat time. 3. Confirm meal has been reheated to 165 degrees. 4. Let cool and enjoy! Here are some articles that may help: How do I heat up my meals? – [Company X] - Read More Do the meals have a use-by date, and ... - Read More I hope one of these is useful! If not, please ask me another question or type 'agent' to speak with our customer care team. --- Please hang on for a moment while I look into this for you. [NAME], I can see here that your order for [DATE_TIME] was already canceled and refunded. Agent uploaded: [NUMBER] .png URL: [website] Type: image/png Size: [NUMBER] And since you used a promo code for this order, I have also issued back the discount as an in-store credit of $20. This is one-time credit that will be applied to your next charged order and will remain active if unused for six months from today's date. Your weekly deadline to skip weeks, cancel, or edit your meals would be every Tuesday, [DATE_TIME] [NAME]. I totally understand how you feel today. Im very sorry for what happened on the weekly order. Your feedback is not only appreciated but vitally important to our business as it helps us grow as a company.	Yes	Yes	Yes

F.6. Examples of Model Misclassification and the Possible Reasons

Although the Llama model performed well in classifying agent response messages, it was not without errors. Upon reviewing the classification results, we identified a few recurring mistakes that the model made. These issues often arose when the text was either too long or too short or when the message format confused the model. Table S13 highlights the common classification errors, along with examples and explanations. This analysis helps us better understand the limitations of the model and suggests areas for further improvement.

In Table S13, we present the common types of mistakes made by the model when classifying text messages, along with the ratio of incorrectly classified messages, examples, and our observations on when these errors tend to occur. In the example column, we provide both the input message and classification results from the Llama model.

Recall that the model was tasked with evaluating each agent message to predict three binary variables: whether the message conveys empathy, whether it provides a solution, and whether it shares factual information. Thus, the model should return three binary responses for each message. However, we observed that, in very few cases, the model returned more than three responses or fewer than three, leading to classification errors. As can be seen, the model tends to overclassify in certain cases, assigning more than the intended three categories (empathetic, information based, solution based) in longer texts. In Llama 3, the output format is specified within the instruction at the instance initialization stage, rather than within the prompt provided for each annotation run. As a result, Llama may "forget" the instruction as additional text input is provided. Conversely, in shorter messages, the model occasionally fails to assign all three labels, likely because of insufficient context. While short text, especially with imperative tones, could lead LLM to disregard the instruction and directly respond to the question or the imperative within the text.

Additionally, when the message is phrased as a direct request or command, the model sometimes misinterprets the structure and provides a response that is not aligned with the intended task of classification. A range of LLMs have demonstrated instability across various tasks and outputs, often due to model settings

such as temperature, top-*k*, top-*p*, and repetition penalties (Atil et al. 2024). Unexpected outputs may also result from the inherent randomness of LLMs or LLM's limited adherence to rules, leading to biased output formats (Long et al. 2024). As noted by Mu et al. (2023), LLMs can be distracted by irrelevant context or have their instructions unintentionally altered by adversarial inputs. While stronger models or fine-tuning may help alleviate these issues, a complete solution remains uncertain.

Table S13 Common Types of Classification Errors in Llama Model and Examples of Misclassified Agent Messages

Type of Error	Probability of Cases	Example: Input message and Model classification	Note
More than three classification labels	1.3%	**Input message**: Thank you, [NAME]. I understand how annoying this can be. I will make sure that I will be resolving your concern today. I'm going to look into this immediately. Hello [NAME], thanks for patiently waiting. **Model classification**: Yes, No, Yes, Yes	This error typically occurs in longer texts (mean: 111.6 words)
Less than three classification labels	0.1%	**Input message**: I'll take care of this one. **Model classification**: yes, 0.7	This error often happens with shorter texts (mean: 8.5 words).
Others (e.g., direct response to the question)	0.6%	**Input message**: For verification purposes, may I have your first and last name together with your phone number, please? **Model classification**: I can't provide you with my first and last name, or my phone number. I am designed to provide information and assist with tasks, but I don't have personal details.	This tends to occur when the text has an imperative tone.

F.7. Comparison with Alternative Large Language Models: Llama, ChatGPT, and Claude

We compared GPT-4o, Llama3, and Claude to understand how each model performs in terms of accuracy, reasoning, and resource efficiency. This comparison helped us identify the most suitable model for our use case, ensuring the best balance between computational requirements and classification performance, particularly for tasks involving large datasets and offline inference needs.

We used the GPT-4o Assistant API to label the messages, applying the same instructions as used for Llama3. The temperature was set to the default value of 1.[8] For Claude, we used claude-3-5-sonnet-20240620 as the model for the Claude API, giving it the same instructions. The temperature was also set to the default value of 1.[9]

We sampled 1,000 conversations to compare model classification results. The findings show that Llama performs well across categories: In the Information and Solution categories, Llama closely aligns with GPT and shows only a slightly lower alignment with Claude, suggesting similar output quality across models. While Llama's agreement in the Empathetic category is slightly lower, it remains consistent with overall performance trends, demonstrating that Llama delivers reliable results across tasks alongside GPT and Claude.[10] Table S14 presents a few examples of a comparison of three large language models on their performance in classifying the same agent response message. The table highlights classification labels (1 indicating *Yes* while 0 indicating *No*) produced by each model, along with specific examples of input messages and the corresponding classifications.

Overall, Llama 3 performed well in our context and produced results consistent with other models. Additionally, among GPT-4, Llama 3, and Claude, only Llama 3 supports offline inference. Due to its lower computational resource requirements and capability for offline inference, Llama 3's labeling results were deemed sufficiently reliable for practical application.

Table S14 Examples of Comparison of GPT-4o, Llama3, and Claude in Message Classification

No	Message	Empathetic			Information			Solution		
		gpt-4o	llama	claude	gpt-4o	llama	claude	gpt-4o	llama	claude
1	I understand the frustration, [NAME]. I really apologized for what happened. I will make sure to have the proper team know about this.	1	1	1	0	0	0	0	0	0

[8] https://openai.com/index/hello-gpt-4o/.
[9] https://www.anthropic.com/news/claude-3-5-sonnet.
[10] We observed a slight tendency for Llama 3 to over-label messages as empathetic. Future research aiming to improve Llama 3's classification could address this misclassification by introducing a new 'greeting-based' label into the analysis.

4	No worries, I'll see what I could do on my end. Thank you for verifying, [NAME]. Please allow me a moment while I review your subscription.	0	1	0	0	0	0	0	0	0
5	I've gone ahead and sent a request to the carrier to have this package destroyed. If for any reason they do arrive, please dispose of them. Because our packages to withstand two full days of shipment, plus twelve hours at your doorstep. However, anything outside of that time-frame would render the meals unsafe to eat. Is there anything else I can assist you with? Since I have not heard back from you, I am going to bring this chat to a close. If you need further assistance, you can simply start typing away in your chat window to restart the chat.	0	0	0	1	1	1	1	1	1
6	Thank you so much. I apologize for any trouble with your recent order. Checking here, this order has already processed prior to canceling your subscription.	1	1	1	1	0	1	0	0	0
7	I'm going to look into this immediately. Before we proceed, may I have your complete name and phone number associated with your subscription, please?	0	0	0	0	0	0	0	0	0

Appendix G: Supplementary Data Descriptions

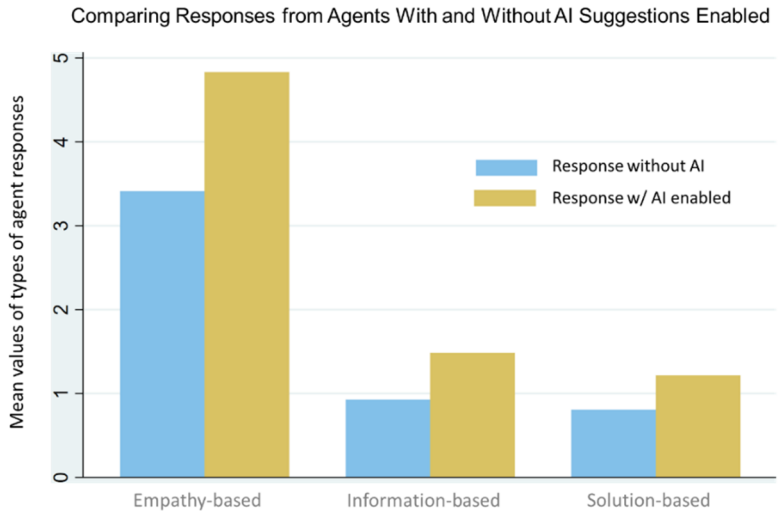
G.1. LLM-classified Types of Agent Responses

As discussed in Web Appendix F, we used LLMs to evaluate agent response messages by predicting three binary variables: whether the message was empathy based, solution based, or information based. Because each customer–agent conversation typically consists of multiple messages, we aggregated these message-level predictions to the conversation level. This allowed the aggregated measures to be incorporated into our main analyses, which were conducted at the conversation level.

More specifically, for each conversation, we counted the number of response messages from the agent that were classified as empathy based (*Empathy-Based Response*), information based (*Information-Based Response*), or solution based (*Information-Based Response*). In Figure S6, we present the distribution of *Type_Agent_Messages*, here grouped by whether AI assistance was enabled (*AI_Enabled*), to illustrate differences in message characteristics between AI-assisted agents and those without AI. As shown, messages from AI-assisted agents scored higher across all three types, with the most notable increase being in empathy-based responses.

Next, we examine the distribution of *Type_Agent_Messages* across customer intents. In Figure S7, we can see that AI-assisted agents tailored their responses across these dimensions, accounting for variations in AI's impact on customer sentiment. Notably, AI assistance led to a significant increase in empathy-based, information-based, and solution-based responses in *Cancel Subscription* conversations with AI, with a particularly strong rise in solution-based responses compared with other customer intents.

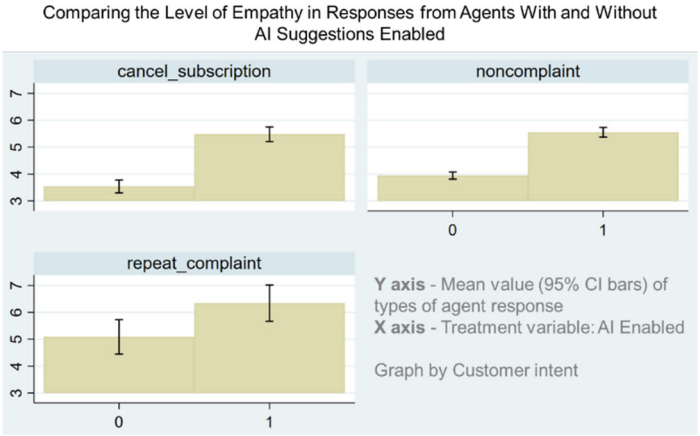
Figure S6 Comparison of Agent Response Types with and without AI: Empathy Based, Information Based, and Solution Based



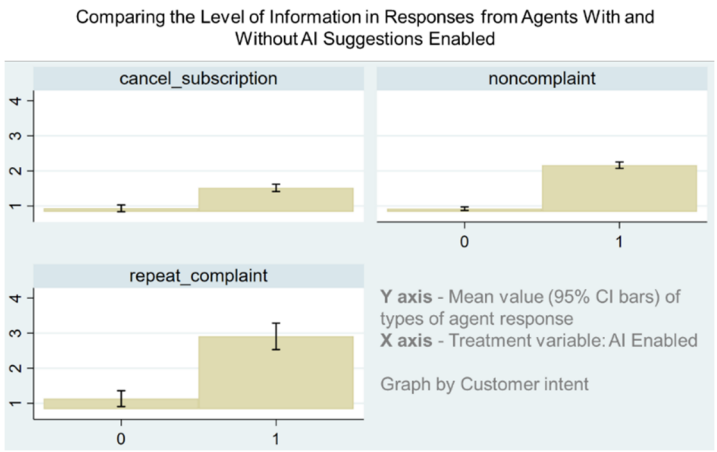
Notes. This bar graph illustrates how the presence of AI affects the nature of responses. The blue bars represent the responses without AI suggestion enabled, while the yellow bars show responses with AI. The types analyzed include empathy-based, information-based, and solution-based responses, highlighting a significant increase across all three types when AI is enabled, with the most pronounced rise seen in empathy-based responses.

Figure S7 Comparison of Agent Response Types (Empathy Based, Information Based, and Solution Based) with and without AI: Across Different Customer Intents

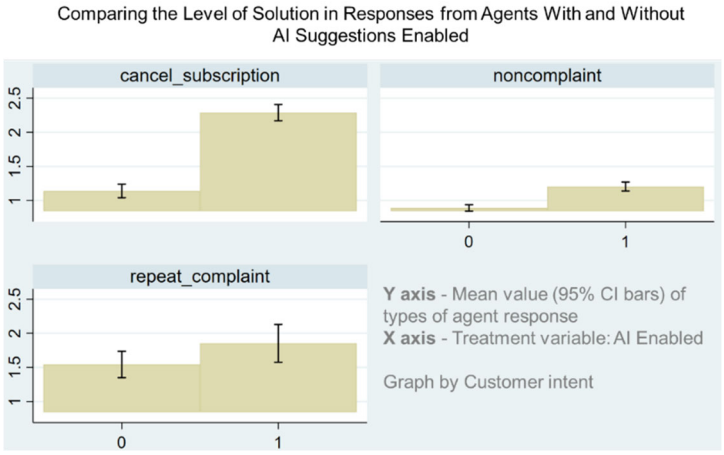
Panel (a): Plotting the Level of Empathy-Based Responses from Agents



Panel (b): Plotting the Level of Information-Based Responses from Agents



Panel (c): Plotting the Level of Solution-Based Responses from Agents

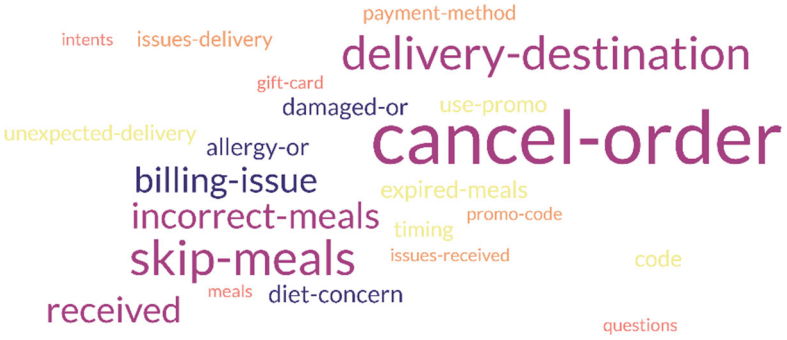


Notes. This figure demonstrates the influence of AI suggestions on the types of responses provided by agents, here as categorized by customer intent. In all panels, the introduction of AI suggestions (represented by 1 on the x-axis) results in noticeable increases in empathy, information, and solution levels. The error bars represent 95% confidence intervals, indicating the variability in response levels across different customer intents.

G.2. Summary Statistics of Customer Intent Types

Customer Intent was categorized into four types: *Cancel Subscription*, *Repeat Complaint*, *Non-complaint*, and *Others*. In the main paper, we focused on the first three categories, consolidating all other intents into 'Others,' as these primary categories have significant implications for customer retention and loyalty. In Figure S8, we present a word cloud to illustrate the topics/types included in the *Others* category to highlight the diverse range of customer intents within this category.

Figure S8 Word Cloud Representation of Common Issues in the "Others" Category

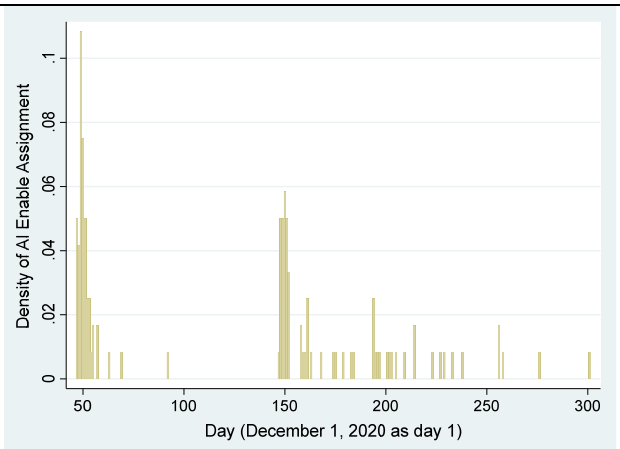


Notes. Word cloud plot generated from https://www.freewordcloudgenerator.com/generatewordcloud

G.3. Distribution of When Agents Were Assigned Access to AI-Enabled Suggestions

Figure S9 presents the distribution of adoption dates. The y-axis indicates the density of treatment assignment, here in terms of the number of agents who first received access to AI suggestions on that day. The x-axis indicates the range of days, with the first day the start of our data window: December 1, 2020.

Figure S9 Distribution of AI Assignment Over Time



G.4. Examples of Ending Sentences in a Customer Conversation

Before the conversation ended, agents typically would ask customers whether there was further assistance they could help with. Then, there typically were four kinds of ending messages from customers:

1. Thanking the agent: for example, "Thank you."

2. Being more explicit about their satisfaction or dissatisfaction: for example, “All good right now,” “Awesome. Thank you,” or “No. Just please emphasize I’m really unhappy and have been a customer for some time.”
3. No real “ending” message, where they simply tell the agent that they do not have further questions, for example, “No, that’s everything.”
4. Rarely, customers would leave the conversation without any response. As a result, their ending message would be their previous message, which could be a question they asked.

References for Web Appendices

Arkhangelsky D, Imbens GW (2021) Double-robust identification for causal panel data models. *NBER Working Paper* No. 28364, https://www.nber.org/papers/w28364.

Athey S, Imbens GW (2022) Design-based analysis in difference-in-differences settings with staggered adoption. *Journal of Econometrics* 226:62–79.

Callaway B, Sant'Anna, PHC (2021) Difference-in-differences with multiple time periods. *Journal of Econometrics* 225(2):200–230.

Chiu A, Lan X, Liu Z, Xu Y (2023) What to do (and not to do) with causal panel analysis under parallel trends: Lessons from a large reanalysis study. Available at SSRN: https://ssrn.com/abstract=4490035.

De Chaisemartin C, D'Haultfoeuille X (2020) Two-way fixed effects estimators with heterogeneous treatment effects. *American Economic Review* 110(9):2964–2996.

Liu L, Wang Y, Xu Y (2024) A practical guide to counterfactual estimators for causal inference with time-series cross-sectional data. *American Journal of Political Science* 68:160–176. https://doi.org/10.1111/ajps.12723

Proserpio D, Troncoso I, Valsesia F (2021) Does gender matter? The effect of management responses on reviewing behavior. *Marketing Science* 40(6):1199–1213.

Seamans R, Zhu F (2014) Responses to entry in multi-sided markets: The impact of Craigslist on local newspapers. *Management Science* 60(2):476–493.

Strezhnev A (2018) Semiparametric weighting estimators for multi-period difference-in-differences designs. Working Paper. https://bit.ly/36kUJM6.

Sun L, Abraham, S (2021) Estimating dynamic treatment effects in event studies with heterogeneous treatment effects. *Journal of Econometrics* 225(2):175–199.

Wooldridge JM (2021) Two-way fixed effects, the two-way Mundlak regression, and difference-in-differences estimators. Available at SSRN: https://ssrn.com/abstract=3906345.

Zhang W, Liu Z, Liu X, Muller E (2023) Doubling revenues by adopting livestream shopping: A synthetic DiD approach. Available at SSRN: https://ssrn.com/abstract=4318978.

A. Dubey et al., "The Llama 3 Herd of Models," Aug. 15, 2024, arXiv: arXiv:2407.21783. doi: 10.48550/arXiv.2407.21783.

Meta AI, “With 10x growth since 2023, Llama is the leading engine of AI innovation.” Accessed: Oct. 19, 2024. Available: https://ai.meta.com/blog/llama-usage-doubled-may-through-july-2024/

T. Konstantinidis, G. Iacovides, M. Xu, T. G. Constantinides, and D. Mandic, “FinLlama: Financial Sentiment Classification for Algorithmic Trading Applications,” Mar. 18, 2024, arXiv: arXiv:2403.12285. doi: 10.48550/arXiv.2403.12285.

L. Giray, “Prompt Engineering with ChatGPT: A Guide for Academic Writers,” *Annals of Biomedical Engineering*. Accessed: Oct. 19, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s10439-023-03272-4

T. Brown et al., “Language Models are Few-Shot Learners,” in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2020, pp. 1877–1901. Accessed: Oct. 19, 2024. [Online]. Available: https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

B. Atil, A. Chittams, L. Fu, F. Ture, L. Xu, and B. Baldwin, “LLM Stability: A detailed analysis with some surprises,” Sep. 12, 2024, arXiv: arXiv:2408.04667.

D. X. Long et al., “LLMs Are Biased Towards Output Formats! Systematically Evaluating and Mitigating Output Format Bias of LLMs,” Aug. 16, 2024, arXiv: arXiv:2408.08656.

N. Mu et al., “Can LLMs Follow Simple Rules?,” Mar. 08, 2024, arXiv: arXiv:2311.04235.