Appendices for "How Forced Intervention Facilitates AI Adoption"

Xinyu Cao[1], Chenshan Hu[2], Jiankun Sun[3], Dennis J. Zhang[4]

1. CUHK Business School, The Chinese University of Hong Kong
2. Leeds School of Business, University of Colorado Boulder
3. Imperial Business School, Imperial College London
4. Olin Business School, Washington University in St. Louis

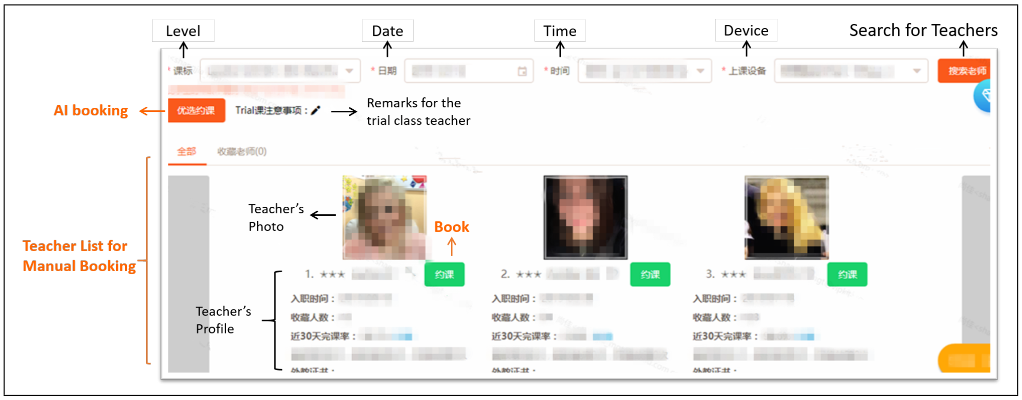
Appendix A: Figures and Tables

Table 7 Definitions of Variables

	Panel A: Booking-level Variables
UseAlgorithm	Binary. Whether the booking is made using the algorithm (= 1) or manually (= 0).
Purchase	Binary. Whether the booking converts to a purchase within 17 days (= 1) or not (= 0).
BookDate	Categorical. The date on which the booking is made.
DuringExp	Binary. Whether the booking is made during the experiment (= 1) or not (= 0).
AfterExp	Binary. Whether the booking is made after the experiment (= 1) or not (= 0).
StudentAge	Categorical. Student age, ranging from 4 to 15.
QualityLead	Binary. Whether the lead is high-quality (= 1) or not (= 0), defined by the company.
Call	Binary. Whether the booking is made through a phone call (= 1) or WeChat (= 0).
	Panel B: Worker-level Variables
AlgorithmGroup	Binary. Whether the worker is assigned to the Algorithm group (= 1) or not (= 0).
Seniority	Categorical. Seniority of the worker, defined by the company according to her tenure. *Seniority* = 1 if the worker has been working in the company for less than 3 months, *Seniority* = 2 if between 3 and 6 months, *Seniority* = 3 if between 6 and 12 months, and *Seniority* = 4 if more than 12 months.
ConversionDiff	Numerical. Change in the worker's experienced conversion rate from the pre-intervention period to the during-intervention period.
Δ*AlgorithmUsageRatio*	Numerical. Change in the worker's average algorithm usage ratio from the pre-intervention period to the post-intervention period.
AlgoUsageRatioPreExp	Numerical. Average algorithm usage ratio of the worker before the experiment.
ConversionAlgoPreExp	Numercial. Conversion rate of algorithmic bookings by the worker before the experiment.
ConversionManPreExp	Numercial. Conversion rate of manual bookings by the worker before the experiment.
ConversionDiffAlgo	Numerical. Change in the conversion rate of algorithmic bookings made by the worker from the pre-intervention period to the during-intervention period.
ConversionDiffMan	Numerical. Change in the conversion rate of manual bookings made by the worker from the pre-intervention period to the during-intervention period.
	Panel C: Worker-day-level Variables
NumBooking	Numerical. Number of bookings made by the worker during the day.
AlgoUsageRatioPrevD	Numerical. Average algorithm usage ratio of the worker on the previous day.
AlgoUsageRatioPrevW	Numerical. Average algorithm usage ratio of the worker in the previous week preceding the day.
ConversionAlgoPrevD	Numercial. Conversion rate of algorithmic bookings by the worker on the previous day.
ConversionAlgoPrevW	Numercial. Conversion rate of algorithmic bookings by the worker in the previous week preceding the day.
ConversionManPrevD	Numercial. Conversion rate of manual bookings by the worker on the previous day.
ConversionManPrevW	Numercial. Conversion rate of manual bookings by the worker in the previous week preceding the day.

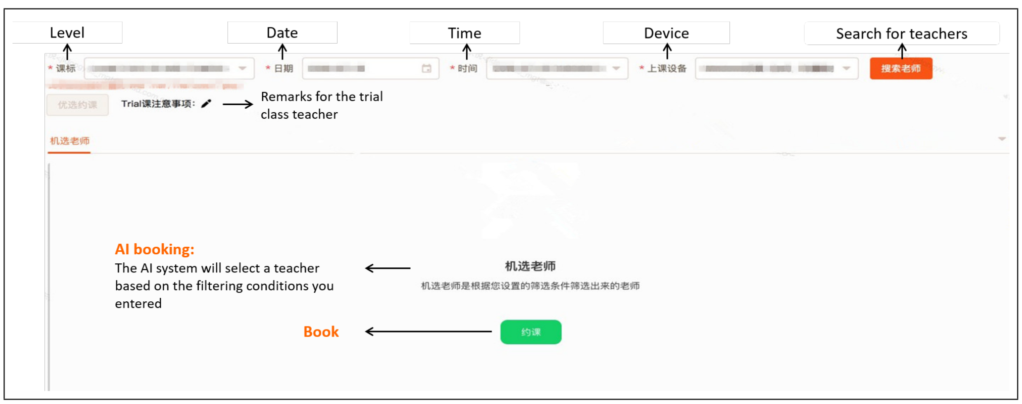
Figure 6 User Interface of Sales Workers

(a) Control group: Both manual booking and algorithmic booking are available



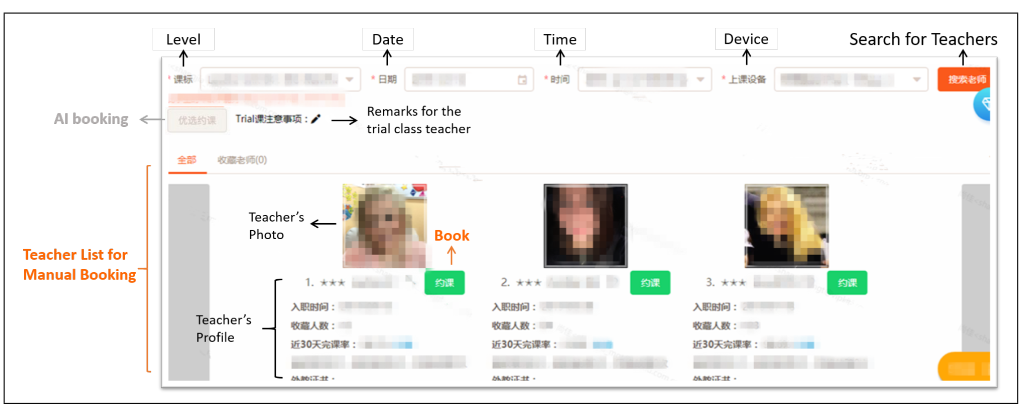
After inputting the filtering conditions, workers click the orange button "Search for teachers." Then a list of available teachers will be displayed below, and workers can manually book a teacher from the list by clicking the green button "Book," or they can click the orange button "algorithmic booking" to let the algorithm book a teacher for the lead.

(b) Algorithm group: Only algorithmic booking is available



After inputting the filtering conditions and clicking "Search for Teachers," there will be no list of teachers displayed. The workers can only click the green button "Book" to let the algorithm book a teacher for the lead.

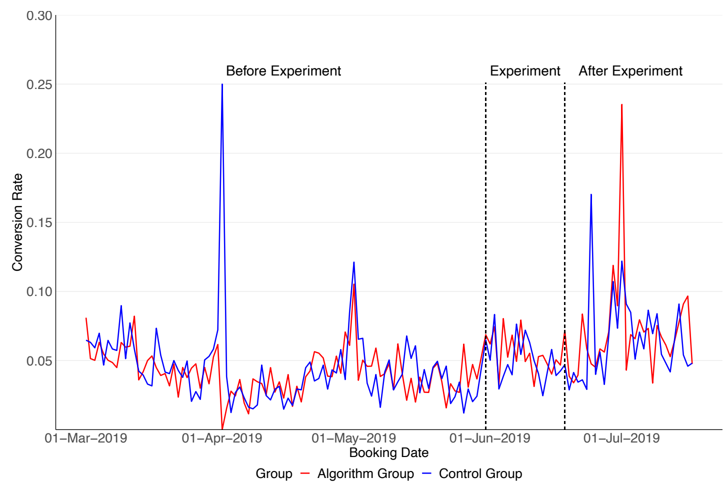
(c) Manual Group: Only manual booking is available



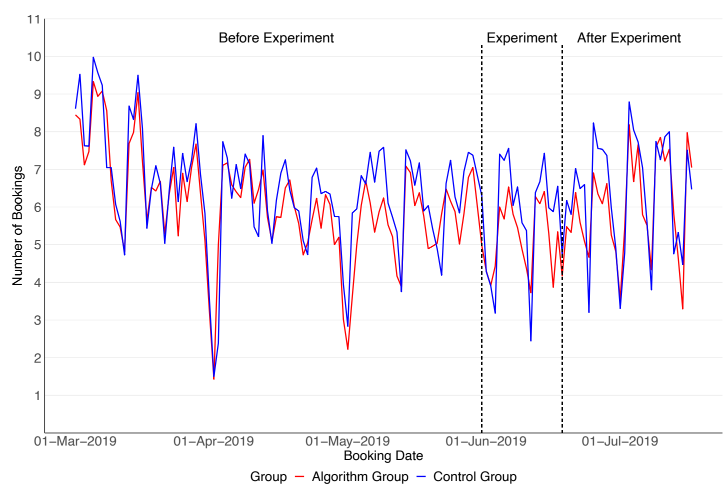
The button "algorithmic booking" is disabled. Workers can only book manually.

Figure 7 Daily Average Operational Performance

(a) Daily Average Conversion Rate



(b) Daily Average Number of Bookings per Worker



(c) Daily Average Number of Converted Bookings per Worker

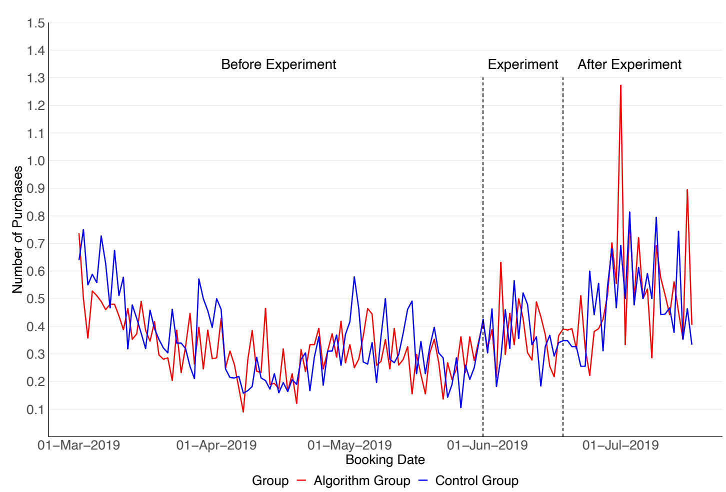


Table 8 Distribution of Sales Workers Across Sales Areas and Offices

Sales Area	Sales Office	Treatment Group			Total
		Algorithm Group	Manual Group	Control Group	
Beijing	First Office	7	5	8	20
		(11.86)	(9.43)	(13.56)	(11.70)
	Second Office	4	0	3	7
		(6.78)	(0.00)	(5.08)	(4.09)
	Third Office	7	4	4	15
		(11.86)	(7.55)	(6.78)	(8.77)
	Fourth Office	2	3	5	10
		(3.39)	(5.66)	(8.47)	(5.85)
	Sixth Office	2	3	2	7
		(3.39)	(5.66)	(3.39)	(4.09)
	Seventh Office	5	2	6	13
		(8.47)	(3.77)	(10.17)	(7.60)
	Ninth Office	1	5	1	7
		(1.69)	(9.43)	(1.69)	(4.09)
	Experienced Office	6	8	3	17
		(10.17)	(15.09)	(5.08)	(9.94)
Shanghai	Fifth Office	3	3	6	12
		(5.08)	(5.66)	(10.17)	(7.02)
	Tenth Office	11	9	12	32
		(18.64)	(16.98)	(20.34)	(18.71)
Chengdu	Eleventh Office	1	0	1	2
		(1.69)	(0.00)	(1.69)	(1.17)
	Twelfth Office	5	4	2	11
		(8.47)	(7.55)	(3.39)	(6.43)
Shenzhen	Fifteenth Office	5	7	6	18
		(8.47)	(13.21)	(10.17)	(10.53)
	Total	59	53	59	171
		(100.00)	(100.00)	(100.00)	(100.00)

Note: This table reports the number of workers in each sales office allocated to each treatment group. We report the percentage that it accounts for the total number of workers in the corresponding treatment group in parentheses. The Fisher's exact test across sales areas has p-value$= 0.882$, and the Fisher's exact test across sales offices has p-value$= 0.697$. The test results indicate that the three treatment groups do not have significantly different distributions of workers across sales areas or sales offices.

Table 9 Balance Check: Summary Statistics Before the Experiment (Three Groups)

	Each Group: Mean (Std. Dev.)			Between-group Comparison: p-value		
	Algorithm	Manual	Control	Algorithm vs. Control	Manual vs. Control	Algorithm vs. Manual
	(1)	(2)	(3)	(4)	(5)	(6)
Panel A: Booking-level Variables						
UseAlgorithm	0.213	0.181	0.215	0.948	0.391	0.403
	(0.409)	(0.385)	(0.411)			
Purchase	+0.0015	−0.0024	–	0.743	0.616	0.419
	–	–	–			
StudentAge	8.140	8.139	8.143	0.963	0.952	0.993
	(2.667)	(2.634)	(2.644)			
QualityLead	0.519	0.502	0.548	0.466	0.255	0.676
	(0.500)	(0.500)	(0.498)			
Call	0.829	0.848	0.824	0.755	0.084*	0.117
	(0.377)	(0.359)	(0.381)			
Observations	27,696	26,365	29,266			
Panel B: Worker-level Variables						
Seniority	2.559	2.396	2.492	0.765	0.682	0.486
	(1.236)	(1.230)	(1.223)			
Observations	59	53	59			
Panel C: Worker-day-level Variables						
NumBooking	6.189	6.638	6.602	0.134	0.911	0.166
	(3.158)	(3.488)	(3.254)			
NumPurchase	−0.0069	−0.0146	−	0.835	0.673	0.825
	(−)	(−)	(−)			
Observations	4,475	3,972	4,433			

Note: This table presents the summary statistics and balance check results of the key variables before the experiment in each group. Columns (1)-(3) report the mean and standard deviation of each variable in each group before the experiment, with standard deviations given in parentheses. Columns (4)-(6) report the p-values of the three t-tests that compare the means of the variables in each pair of the groups. Due to the NDA, we cannot report the means and standard deviations of *Purchase* and *NumPurchase*. Instead, we report the difference in the means of *Purchase* and *NumPurchase* among the three groups, using the Control group as the benchmark. Standard errors are clustered at the worker level for the t-tests in Panels A and C. *p<0.1; **p<0.05; ***p<0.01.

Table 10 Balance Check: Summary Statistics During the Experiment (Three Groups)

	Each Group: Mean (Std. Dev.)			Between-group Comparison: p-value		
	Algorithm	Manual	Control	Algorithm vs. Control	Manual vs. Control	Algorithm vs. Manual
	(1)	(2)	(3)	(4)	(5)	(6)
StudentAge	8.057	7.909	7.939	0.185	0.739	0.102
	(2.708)	(2.636)	(2.614)			
QualityLead	0.571	0.535	0.563	0.835	0.497	0.382
	(0.495)	(0.499)	(0.496)			
Call	0.830	0.831	0.823	0.681	0.630	0.956
	(0.375)	(0.375)	(0.382)			
Observations	4,118	4,534	5,051			

Note: This table presents the summary statistics of the lead characteristics during the experiment in each group. Columns (1)-(3) report the mean and standard deviation of each variable in each group during the experiment, with standard deviations given in parentheses. Columns (4)-(6) report the p-values of the three t-tests that compare the means of the variables in each pair of the groups. Standard errors are clustered at the worker level. *p<0.1, **p<0.05, ***p<0.01.

Table 11 Balance Check: Summary Statistics After the Experiment (Three Groups)

	Each Group: Mean (Std. Dev.)			Between-group Comparison: p-value		
	Algorithm	Manual	Control	Algorithm vs. Control	Manual vs. Control	Algorithm vs. Manual
	(1)	(2)	(3)	(4)	(5)	(6)
StudentAge	8.487	8.356	8.453	0.711	0.269	0.124
	(2.751)	(2.683)	(2.691)			
QualityLead	0.629	0.570	0.608	0.601	0.331	0.155
	(0.483)	(0.495)	(0.488)			
Call	0.808	0.822	0.814	0.732	0.675	0.422
	(0.394)	(0.383)	(0.389)			
Observations	7,005	7,493	7,677			

Note: This table presents the summary statistics of the lead characteristics after the experiment in each group. Columns (1)-(3) report the mean and standard deviation of each variable in each group after the experiment, with standard deviations given in parentheses. Columns (4)-(6) report the p-values of the three t-tests that compare the means of the variables in each pair of the groups. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01.

Table 12 Number of Workers Before and After the Experiment (Three Groups)

	Algorithm Group	Manual Group	Control Group	Total
Experiment Participant Count (May 30, 2019)	59	53	59	171
Post-experiment Participant Count (June 19, 2019)	48	46	47	141
Probability of Staying After the Experiment	0.814	0.868	0.797	0.825
	Algorithm vs. Control	Manual vs. Control	Algorithm vs. Manual	
p-value of Chi-squared Test	0.816	0.315	0.434	

Note: This table reports the number of workers that participated in and stayed after the experiment in each group. It also reports the retention rate of each group and across all the three groups, along with the p-values of the chi-squared tests on the retention rates between each pair of the groups. *p<0.1; **p<0.05; ***p<0.01.

Table 13 Effects of Using Algorithm on Conversion Rate During Experiment

	Dependent variable: *Purchase*	
	(1)	(2)
AlgorithmGroup × *DuringExp*	0.0058	0.0028
	(0.0056)	(0.0051)
AlgorithmGroup	0.0015	
	(0.0046)	
DuringExp	0.0111***	
	(0.0039)	
Date Fixed Effects	No	Yes
Worker Fixed Effects	No	Yes
Lead Characteristics	No	Yes
Observations	66,131	66,131

Note: This table reports the estimated treatment effects of forced algorithm use on conversion rate during the experiment in the Algorithm group compared to the Control group. We use Specifications (1) and (2) with the dependent variable being $Purchase_{ijt}$ and $AfterExp_t$ replaced by $DuringExp_t$ for estimation. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01.

Table 14 Empirical Tests for No Learning Before the Experiment

	Dependent variable: *UseAlgorithm*			
	(1)	(2)	(3)	(4)
AlgoUsageRatioPrevD	0.3707***			
	(0.0376)			
ConversionAlgoPrevD	0.0201			
	(0.0231)			
ConversionManPrevD	0.0146			
	(0.0451)			
AlgoUsageRatioPrevW		0.6190***		
		(0.0467)		
ConversionAlgoPrevW		-0.0370		
		(0.0323)		
ConversionManPrevW		0.0319		
		(0.0759)		
AlgoUsageRatioPreExp			0.6263***	
			(0.1522)	
ConversionAlgoPreExp			-0.0719	
			(0.2876)	
ConversionManPreExp			0.9931	
			(1.0412)	
ConversionAlgoPreExp × *DuringExp*				-0.2637
				(0.3997)
ConversionManPreExp × *DuringExp*				0.7460
				(1.1884)
Date Fixed Effects	Yes	Yes	Yes	Yes
Worker Fixed Effects	Yes	Yes	No	Yes
Lead Characteristics	Yes	Yes	Yes	Yes
Observations	22,642	36,408	3,930	28,026

Note: This table reports the empirical test results for the non-existence of learning before the experiment. The definitions of all the variables are given in Table 7. Columns (1)-(2) use all bookings made by the Algorithm and Control groups before the experiment where a worker has ever used the algorithm for bookings on the previous day (Column (1)) or in the previous week preceding the day of observation (Column (2)). Columns (3) and (4) use all bookings during the experiment (Column (3)) and before and during the experiment (Column (4)) which are made by workers in the Control group that have ever used algorithm for bookings before the experiment. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01.

Columns (1)-(2) show that before the experiment, workers' algorithm usage is positively correlated with their past algorithm usage ratio (on the previous day or in the previous week), but is not affected by the past performance of algorithmic bookings or manual bookings (i.e., the conversion rate of algorithmic bookings or manual bookings made on the previous day or in the previous week). To address the potential concern that one day or one week may be too short for workers to learn the performance of the algorithm, we also explore whether the performance of the algorithm before the experiment can influence workers' algorithm usage during the experiment in Columns (3) and (4). We utilize the data in the Control group for this analysis, since the Control group can freely choose whether to use the algorithm for each lead during the experiment. Columns (3) and (4) show that even if we use the entire pre-experiment period to measure the performances of algorithm or manual bookings, they still do not affect workers' algorithm usage during the experiment. This further validates that there is no learning before the experiment.

Table 15 Summary Statistics of ConversionDiff and ΔAlgorithmUsageRatio

	Statistics	Algorithm group	Control Group
ConversionDiff	Mean	0.0164	0.0156
	Ste. Dev.	0.0264	0.0278
	Min	-0.0378	-0.0420
	Max	0.0793	0.1151
Δ*AlgorithmUsageRatio*	Mean	0.186	-0.003
	Std. Dev.	0.293	0.228
	Min	-0.417	-0.604
	Max	0.846	0.693
	Observations	47	47

Note: This table reports the summary statistics of *ConversionDiff* and Δ*AlgorithmUsageRatio* of all sales workers who have non-empty values of these two variables in the Algorithm and Control groups. Notice that the number of observations in the Algorithm group is smaller than the number of workers who stayed after the experiment in this group, because there is one sales worker in the Algorithm group who stayed after the experiment but it happened that they made no bookings during the experiment, and therefore their *ConversionDiff* is missing.

Table 16 Learning: Empirical Tests Using Alternative Definitions of ConversionDiff

	Dependent variable: *UseAlgorithm*		
	(1)	(2)	(3)
AfterExp × *ConversionDiffAlgo*	1.1998*	0.5166	
	(0.6857)	(0.5198)	
AfterExp × *ConversionDiffMan*			0.1470
			(1.1342)
Date Fixed Effects	Yes	Yes	Yes
Worker Fixed Effects	Yes	Yes	Yes
Lead Characteristics	Yes	Yes	Yes
Observations	32,277	34,739	36,700

Note: This table reports estimation results of Specificaion (11) using alternative measures of the change in conversion rates of a worker, i.e., *ConversionDiffAlgorithm* and *ConversionDiffManual*, with definitions given in Table 7. Column (1) reports the results using the bookings made by the Algorithm group before and after the experiment. Columns (2) and (3) reports the results using the bookings made by the Control group before and after the experiment with non-empty values of *ConversionDiffAlgo* and *ConversionDiffMan*, respectively. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01.
Column (1) confirms that learning exists in the Algorithm group. The reason why the coefficient is smaller than that in Column (1) of Table 5 is possibly because most workers have low algorithm usage before the experiment. In other words, it is likely that workers in the Algorithm group learn the performance of the algorithm by comparing the conversion rate during the experiment to the overall conversion rate before the experiment rather than the conversion rate of algorithmic bookings before the experiment. Columns (2)-(3) again confirm that there is no learning in the Control group.

Appendix B: Additional Analyses

B.1. Additional Discussion on Workers' Exiting Behavior

Table 12 in Appendix A reports the numbers of workers that participated in the experiment and stayed after the experiment, as well as their probabilities of staying after the experiment, for both the Algorithm and Control groups. We notice that 23 workers quited their jobs during the experiment in the two groups. The probability of a worker staying after the experiment is not statistically different between the two groups, and the drop-out rate of the Control group is directionally higher than that of the Algorithm group. Therefore, the treatment (i.e., forcing workers to use the algorithm) cannot be the major driver of these workers' dropouts.

Moreover, in Columns (2) and (3) of Table 1 in Section 5.1, including worker fixed effects in the regression automatically removes the workers who quitted the jobs during the experiment and restricts the analysis to those who stayed after the experiment. If the workers who quitted during the experiment are systematically different between the two groups (even though the average quitting probabilities are similar between the two groups), the estimate of the post-intervention effect with worker fixed effects controlled for will be significantly different from that without worker fixed effects controlled for. In other words, the change in the estimation results reflects the extent to which the post-intervention effect is influenced by workers' self-selection behavior, i.e., workers in the Algorithm group who prefer using the algorithm are more likely to stay after the experiment. Table 1 shows that after adding worker fixed effects, the estimated post-intervention effect for the Algorithm group gets slightly larger but qualitatively unchanged. This indicates that the post-intervention effect on the Algorithm group is not due to workers' self-selection but rather the behavioral change of the workers who stayed, i.e., forcing workers to use the algorithm makes them more likely to continue using it after the experiment.

B.2. Post-intervention Algorithm Usage at the Individual Level

In this section, we examine the individual-level effect of the forced intervention on workers' algorithm usage ratio after the experiment. Specifically, for workers in the Algorithm and Control groups, we conduct OLS regression analysis using the following specification:

$$AvgUseAlgorithm_{it} = \lambda_0 + \lambda_1 AlgorithmGroup_i \times AfterExp_t + \lambda_2 AlgorithmGroup_i + \lambda_3 AfterExp_t + \epsilon_{it}, \tag{14}$$

$$AvgUseAlgorithm_{it} = \lambda_0 + \lambda_1 AlgorithmGroup_i \times AfterExp_t + \tau_t + \mu_i + \epsilon_{it}, \tag{15}$$

where $AvgUseAlgorithm_{it}$ denotes the average algorithm usage ratio of all bookings made by worker i on day t and all other variables are defined as in the paper. Table 17 presents the estimation results. We find that forced intervention significantly increases the post-intervention algorithm usage ratio for workers in the Algorithm group compared to the Control group – by 73.10% without fixed effects included and 83.39% with date and worker fixed effects included in our estimation. These results are consistent with our main findings in Section 5.1, i.e., forced intervention leads to a significant increase in the post-intervention algorithm usage for workers in the Algorithm group.

Table 17 Post-Intervention Effect on Individual-level Algorithm Usage

	Dependent variable: *AvgUseAlgorithm*	
	(1)	(2)
AlgorithmGroup × *AfterExp*	0.1549^{***}	0.1767^{***}
	(0.0559)	(0.0639)
AlgorithmGroup	−0.0023	
	(0.0382)	
AfterExp	−0.0005	
	(0.0340)	
Relative Effect Size	73.10%	83.39%
Date Fixed Effects	No	Yes
Worker Fixed Effects	No	Yes
Observations	213	213

Note: This table reports the estimated treatment effects on sales workers' individual-level algorithm usage after the intervention. Columns (1) does not control for date and worker fixed effects while Column (2) controls for these fixed effects. The relative effect size is computed based on the average algorithm usage of workers from the Algorithm group before the experiment. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01.

B.3. Heterogeneous Treatment Effects of Forced Intervention

In this section, we explore the heterogeneous treatment effects of forced intervention on post-intervention algorithm usage ratio across workers with different characteristics. Specifically we examine how the treatment effects are moderated by the following variables: the seniority of a worker ($Seniority_i$), the algorithm usage ratio of a worker before the experiment ($AlgoUsageRatioPreExp_i$), and the conversion rate of all bookings made by a worker before the experiment ($ConversionPreExp_i$). We conduct the OLS regression analyses with the following specification:

$$\begin{aligned} UseAlgorithm_{ijt} = &\alpha_0 + \alpha_1 AlgorithmGroup_i \times AfterExp_t + \alpha_2 Moderator_i \times AfterExp_t \\ &+ \alpha_3 AlgorithmGroup_i \times AfterExp_t \times Moderator_i + \tau_t + \mu_i + X_j + \epsilon_{ijt}, \end{aligned} \tag{16}$$

where $Moderator_i \in \{Seniority_i, AlgoUsageRatioPreExp_i, ConversionPreExp_i\}$ and all other variables are defined as in the paper. Table 18 reports the estimation results. We find that the coefficient for *AlgorithmGroup* × *Seniority* × *AfterExp* lacks statistical significance, showing that workers of different seniority levels do not exhibit differential changes in algorithm usage in response to the forced intervention. Similarly, the triple interaction coefficients in Columns (2) and (3) are also insignificant, indicating that the forced intervention does not produce heterogeneous effects on algorithm adoption across workers with varying pre-intervention algorithm usage ratio or conversion rate. These findings collectively demonstrate that the impact of forced intervention on post-intervention algorithm usage has no significant heterogeneity across workers with different seniority or past performance measures.

B.4. Power Analysis for the Post-intervention Operational Performance Evaluation

Given that the sample size for the individual-level analysis in Table 4 is smaller than that in other analyses of the paper, the corresponding insignificant findings require an examination of statistical power. Specifically, we conduct a Monte Carlo power analysis to examine whether our sample size is sufficient to detect meaningful treatment effects. The simulation procedure involves generating $1,000$ datasets that replicate the structure characteristics of our experiment, including the number of workers, observation days, and the empirically observed within- and between-worker variance. We then analyze each simulated dataset using Specifications (4) and (5). The resulting statistical power is subsequently calculated as the proportion of these simulations in which the interaction term for the treatment effect is statistically significant at the 5% level.

Table 18 Heterogeneous Treatment Effects on Algorithm Usage Ratio

	Dependent variable: *UseAlgorithm*		
	(1)	(2)	(3)
AlgorithmGroup × *Seniority* × *AfterExp*	0.0255		
	(0.0401)		
AlgorithmGroup × *AlgoUsageRatioPreExp* × *AfterExp*		0.0479	
		(0.2972)	
AlgorithmGroup × *ConversionPreExp* × *AfterExp*			−1.0246
			(2.0516)
Seniority × *AfterExp*	−0.0214		
	(0.0239)		
AlgoUsageRatioPreExp × *AfterExp*		-0.4234^{**}	
		(0.1773)	
ConversionPreExp × *AfterExp*			−0.4418
			(0.9512)
AlgorithmGroup × *AfterExp*	0.1314	0.1986^{***}	0.2436
	(0.1202)	(0.0702)	(0.1006)
Date Fixed Effects	Yes	Yes	Yes
Worker Fixed Effects	Yes	Yes	Yes
Lead Characteristics	Yes	Yes	Yes
Observations	71,644	71,644	71,644

Note: This table reports the heterogeneous treatment effects on workers' post-intervention algorithm usage. Standard errors are clustered at the worker level. *p<0.1; **p<0.05; ***p<0.01

Specifically, we evaluate the statistical powers of detecting 5% and 10% effect sizes in Specifications (4) and (5) in the paper (i.e., the treatment effect λ_1 is 5% or 10% of the pre-intervention average of the Control group) given our sample size. For Specification (4), our sample size achieves a statistical power of 84.7% for detecting a 5% effect size and 100% for a 10% effect size. For Specification (5), the statistical powers of our sample size are 89.7% and 100% for detecting 5% and 10% effect sizes, respectively. These results suggest that our sample size is sufficient to detect small-to-moderate treatment effects. Thus, the insignificant interaction terms in Columns (3)-(6) of Table 4 likely reflect the treatment effects being absent or too small to be detected, rather than a result of insufficient sample size.

Appendix C: Extension of the Theoretical Model in Section 6.2

C.1. Extension to General Utility Functions

In this section, we extend the theoretical framework of Section 6.2 by allowing for more general utility functions. We demonstrate that testing rule established in Section 6.2 remains valid: If learning exists, a worker who experiences a greater positive update in her belief about algorithm performance during the intervention (relative to before the intervention) should exhibit a larger increase in her algorithm usage ratio after the intervention (relative to before).

Consider three time periods: $t = 0$ (before the experiment), $t = 1$ (during the experiment), and $t = 2$ (after the experiment). In each period t, worker i holds a belief about algorithm performance, $v_{ti} \sim N(\mu_{ti}, \sigma_{ti}^2)$. When $t = 0$ or 2, worker i chooses $a_{ti} \in [0, 1]$, the probability of using the algorithm to book a trial class for a lead in period t. The worker selects a_{ti} to maximize her expected utility from serving a lead, given her belief $v_{t-1,i}$ and action $a_{t-1,i}$ in the previous period $t - 1$:

$$U(a_{ti}; \mu_{t-1,i}, a_{t-1,i}) = \mathrm{E}[R(a_{ti}, v_{t-1,i}) \mid \mu_{t-1,i}] - C(a_{ti}) - H(|a_{ti} - a_{t-1,i}|). \tag{17}$$

Here, $R(a_{ti}, v_{t-1,i})$ denotes the reward from serving a lead given the algorithm usage ratio a_{ti} and belief about algorithm performance in the last period $v_{t-1,i}$. $C(a_{ti})$ is the effort cost of serving a lead given the algorithm usage ratio a_{ti}. $H(|a_{ti} - a_{t-1,i}|)$ is a "habit cost" that captures the disutility of deviating from the previous algorithm usage level. In period $t = 0$, the habit cost should be omitted, as there is no prior action. This omission does not affect our subsequent analysis, so we do not discuss this case separately. For simplicity, we drop the subscripts i and t where there is no ambiguity, and denote a_0 as the action in the previous period to distinguish it from the action a in the current period.

We impose the following assumptions on the model primitives. First, the reward function $R(a,v)$ is twice continuously differentiable in both arguments on $[0,1] \times V$, where $V \subseteq \mathbb{R}$, and weakly concave in a. The weak concavity of R in a reflects the diminishing marginal returns from using the algorithm progressively on less promising leads. We also assume the cross-partial derivative satisfies $R_{av}(a,v) = \partial^2 R/\partial a \partial v > 0$ for all (a,v), which captures the idea that an improvement in the belief about algorithm performance raises the marginal benefit of increasing the algorithm usage level. Second, the cost function $C(a)$ is twice continuously differentiable and strictly convex in a. The convexity of $C(a)$ reflects increasing marginal effort cost as a increases: As workers expand algorithm usage on additional leads, the marginal savings in booking effort (i.e., effort spent on selecting teachers) decreases, since workers tend to automate the most time-consuming manual bookings first. However, the marginal decision effort increases, as it becomes increasingly difficult to determine on borderline cases whether relying on the algorithm or personal judgment is better. Third, the habit cost $H(x)$ is twice continuously differentiable, non-decreasing, and weakly convex in $x \geq 0$. These conditions capture the idea that the disutility of deviating from previous algorithm usage level increases with the magnitude of deviation, and the marginal cost of further deviation is non-decreasing. Finally, we assume that the marginal utility of increasing algorithm usage at the endpoints satisfy $U_a(0;\mu,a_0) > 0$ and $U_a(1;\mu,a_0) < 0$ for all μ and a_0. This reflects the situation that a worker finds it worthwhile to use the algorithm on some, but not all, leads, which is typical in the cold-start stage of algorithm development as discussed in the paper.

We now show that, for any fixed a_0, the worker's optimal action $a^*(\mu,a_0) = \arg\max_{a\in[0,1]} U(a;\mu,a_0)$ is strictly increasing in μ. First, under the stated assumptions, $R(a,v)$ is concave in a, while $C(a)$ and $H(|a - a_0|)$ are convex in a, with $C(a)$ strictly convex. Thus, $U(a;\mu,a_0)$ is strictly concave in a, and the boundary conditions ensure the existence of a unique interior maximizer $a^*(\mu,a_0) \in (0,1)$ for every μ and a_0.

Next, we compute the cross-partial derivative of the utility function with respect to a and μ. Since C and H do not depend on μ, we have

$$U_{a\mu}(a;\mu,a_0) = \frac{\partial}{\partial a \partial \mu} \int_{-\infty}^{\infty} R(a,v) f(v;\mu) dv = \frac{\partial}{\partial \mu} \int_{-\infty}^{\infty} R_a(a,v) f(v;\mu) dv, \tag{18}$$

where $f(v;\mu)$ denotes the density function of the normal distribution $N(\mu,\sigma^2)$. By the dominated convergence theorem,

$$\frac{\partial}{\partial \mu} \int R_a(a,v) f(v;\mu) dv = \int R_a(a,v) \frac{\partial}{\partial \mu} f(v;\mu) dv. \tag{19}$$

Recall that $\frac{\partial}{\partial \mu} f(v;\mu) = \frac{v-\mu}{\sigma^2} f(v;\mu)$. Therefore,

$$U_{a\mu}(a;\mu,a_0) = \frac{1}{\sigma^2} \int R_a(a,v)(v-\mu) f(v;\mu) dv = \frac{1}{\sigma^2} \operatorname{Cov}_\mu(R_a(a,V),V). \tag{20}$$

By Stein's lemma for the normal distribution,

$$\mathrm{Cov}_\mu(R_a(a,V),V) = \sigma^2 \mathbb{E}_\mu[R_{av}(a,V)], \tag{21}$$

and thus

$$U_{a\mu}(a;\mu,a_0) = \mathbb{E}_\mu[R_{av}(a,V)] > 0, \tag{22}$$

where the positivity follows from $R_{av}(a,v) > 0$ for all (a,v).

The monotonicity of $a^*(\mu,a_0)$ in μ now follows from the implicit function theorem. The first-order condition for optimality is

$$\frac{\partial U}{\partial a}(a^*(\mu,a_0);\mu,a_0) = 0. \tag{23}$$

Differentiating both sides with respect to μ yields

$$U_{aa}(a^*(\mu,a_0);\mu,a_0) \cdot \frac{\partial a^*(\mu,a_0)}{\partial \mu} + U_{a\mu}(a^*(\mu,a_0);\mu,a_0) = 0, \tag{24}$$

and thus

$$\frac{\partial a^*(\mu,a_0)}{\partial \mu} = -\frac{U_{a\mu}(a^*(\mu,a_0);\mu,a_0)}{U_{aa}(a^*(\mu,a_0);\mu,a_0)}. \tag{25}$$

Because $U_{a\mu} > 0$ and $U_{aa} < 0$ (by strict concavity), it follows that $\frac{\partial a^*(\mu,a_0)}{\partial \mu} > 0$. Therefore, the worker's optimal algorithm usage $a^*(\mu,a_0)$ is strictly increasing in her belief about algorithm performance μ, holding the previous action a_0 fixed.

To conclude, for a worker i in the algorithm group, let $\Delta\mu_i = \mu_{1i} - \mu_{0i}$ and $\Delta a_i = a^*_{2i}(\mu_{1i},a_{1i}) - a^*_{0i}(\mu_{0i})$. Since $a^*_{2i}(\mu_{1i},a_{1i})$ is strictly increasing in μ_{1i} for given a_{1i}, it follows that, holding μ_{0i} and $a^*_{0i}(\mu_{0i})$ fixed, Δa_i is strictly increasing in $\Delta\mu_i$. This establishes that the testing rule of Section 6.2 holds under the more general utility formulation considered here.

C.2. Further Extension to Day-to-Day Variation in a Worker's Behavior

We can extend the model further by refining the temporal granularity of the model so that each period t corresponds to one calendar day in the experiment. This allows for day-to-day variation in a worker's belief about the algorithm performance $v_t \sim N(\mu_t, \sigma_t^2)$ and in her algorithm usage ratio a_t. For notational simplicity, we omit the worker index i. Let T_0, T_1, T_2 denote the sets of days corresponding to the pre-experiment, experiment, and post-experiment periods, respectively. Assume that, for each k, the sequence $\{\mu_t(\bar{\mu}_k) : t \in T_{k+1}\}$ is a family of random vectors indexed by $\bar{\mu}_k$, where $\bar{\mu}_k = \mathbb{E}[\frac{1}{|T_{k+1}|}\sum_{t\in T_{k+1}} \mu_t]$. That is, a worker's belief about algorithm performance in period T_{k+1} is a random vector with component-wise average being her average belief formed in the previous period T_k. We require that $\{\mu_t(\bar{\mu}_k) : t \in T_{k+1}\}$ is strictly stochastically increasing in $\bar{\mu}_k$; that is, for any $\bar{\mu}_k^1 < \bar{\mu}_k^2$, the random vector $\{\mu_t(\bar{\mu}_k^2) : t \in T_{k+1}\}$ strictly stochastically dominates $\{\mu_t(\bar{\mu}_k^1) : t \in T_{k+1}\}$.

Define $\Delta\bar{\mu} = \bar{\mu}_1 - \bar{\mu}_0$ and $\Delta\bar{a} = \bar{a}_2^* - \bar{a}_0^*$, where $\bar{a}_k^* = \frac{1}{|T_k|}\sum_{t\in T_k} a_t^*(\mu_{t-1}, a_{t-1})$ is the average optimal algorithm usage ratio during period T_k. We aim to show that the testing rule in Section 6.2 remains valid under this extension; specifically, that $\Delta\bar{a}$ is strictly increasing in $\Delta\bar{\mu}$ almost surely.

Recall the first-order condition characterizing the optimal algorithm usage ratio $a^*(\mu, a_0)$:

$$U_a(a^*(\mu, a_0); \mu, a_0) = 0, \tag{26}$$

where U is strictly concave in a and twice continuously differentiable. Differentiating both sides with respect to a_0 yields

$$U_{aa}(a^*(\mu, a_0); \mu, a_0) \cdot \frac{\partial a^*(\mu, a_0)}{\partial a_0} + U_{aa_0}(a^*(\mu, a_0); \mu, a_0) = 0, \tag{27}$$

so

$$\frac{\partial a^*(\mu, a_0)}{\partial a_0} = -\frac{U_{aa_0}(a^*(\mu, a_0); \mu, a_0)}{U_{aa}(a^*(\mu, a_0); \mu, a_0)}. \tag{28}$$

Given that $H(x)$ is non-decreasing and weakly convex for $x \geq 0$ and enters U as a cost $-H(|a - a_0|)$, we compute:

$$U_{aa_0}(a, \mu, a_0) = H''(|a - a_0|) \geq 0. \tag{29}$$

Since $U_{aa} < 0$ (by strict concavity of U) and $U_{aa_0} \geq 0$, we have $\frac{\partial a^*}{\partial a_0} \geq 0$; that is, $a^*(\mu, a_0)$ is weakly increasing in a_0.

We next show that $\bar{a}_2^*$ strictly increases with $\bar{\mu}_1$ almost surely. Take any $\bar{\mu}_1^1 < \bar{\mu}_1^2$. By the stochastic dominance assumption and Strassen's theorem, there exists a coupling such that, $\mu_t(\bar{\mu}_1^1) < \mu_t(\bar{\mu}_1^2)$ for all $t \in T_2$ almost surely. Denote τ_2 as the first day in T_2. Fixing $a_{\tau_2 - 1}$, let $\{a_t^*(\bar{\mu}_1^1) : t \in T_2\}$ and $\{a_t^*(\bar{\mu}_1^2) : t \in T_2\}$ be the sequences of optimal actions in period T_2 corresponding to the sequences of beliefs $\{\mu_t(\bar{\mu}_1^1) : t \in T_2\}$ and $\{\mu_t(\bar{\mu}_1^2) : t \in T_2\}$, respectively.

We proceed by induction. On the first day $t = \tau_2$, since $a^*(\mu, a_0)$ is strictly increasing in μ and weakly increasing in a_0, we have

$$a_{\tau_2}^*(\mu_{\tau_2 - 1}(\bar{\mu}_1^1), a_{\tau_2 - 1}) < a_{\tau_2}^*(\mu_{\tau_2 - 1}(\bar{\mu}_1^2), a_{\tau_2 - 1}). \tag{30}$$

almost surely. Assume that for some $t > \tau_2$, $a_{t-1}^*(\bar{\mu}_1^1) < a_{t-1}^*(\bar{\mu}_1^2)$ almost surely. Then, since $a^*(\mu, a_0)$ is strictly increasing in μ and weakly increasing in a_0,

$$a_t^*(\mu_{t-1}(\bar{\mu}_1^1), a_{t-1}^*(\bar{\mu}_1^1)) < a_t^*(\mu_{t-1}(\bar{\mu}_1^2), a_{t-1}^*(\bar{\mu}_1^2)) \tag{31}$$

almost surely. By induction, this holds for all $t \in T_2$.

Therefore,

$$\bar{a}_2^*(\bar{\mu}_1^1) = \frac{1}{|T_2|} \sum_{t \in T_2} a_t^*(\bar{\mu}_1^1) < \frac{1}{|T_2|} \sum_{t \in T_2} a_t^*(\bar{\mu}_1^2) = \bar{a}_2^*(\bar{\mu}_1^2) \tag{32}$$

almost surely. Thus, $\bar{a}_2^*$ is strictly increasing in $\bar{\mu}_1$. It follows that $\Delta \bar{a}$ is a strictly increasing function of $\Delta \bar{\mu}$ almost surely holding $\bar{\mu}_0$ and $\bar{a}_0^*$ constant, so the testing rule in Section 6.2 remains valid under this extension.