We thank the reviewers for their helpful comments and suggestions. Due to space limitation, we focus on 1) role of doubly-robustness, 2) comparison with Bastani and Bayati, 3) hyperparameters and 4) real-world experiments.

Role of Doubly-Robustness (DR): A key strength of doubly-robust (DR) method is to obtain an estimate of the reward corresponding to the average context $\bar{b}(t)$. Since $\bar{b}(t)$ satisfies the compatibility condition, we can make use of the fast convergence of the Lasso estimator. If the reward for every arm is observed, the problem becomes much easier. However in bandits, only $r_{a(t)}(t)$ is observed and the rewards of other arms remain missing. In missing data literature, there are two approaches: inverse probability weighting (IPW) and imputing (IMP). IPW and IMP require correct specification of the probability of observation and the imputation model, respectively. DR method uses both auxiliary models, but the consistency of the estimator is guaranteed when either of the models is correctly specified. **In the bandit setting, the probability of observation is known, thus both IPW and DR yield consistent and unbiased estimators.** IPW was used in the EXP4.P bandit algorithm of Beygelzeimer et al. (2011). **Another strength of DR is that when both models are correctly specified, the resulting estimator has the minimum variance.** This efficiency is obtained by projecting the IPW estimating function on to the tangent space spanned by nuisance parameters in the probability of observation (or allocation in bandit setting). The form of $\hat{r}$ reflects this adjustment from an IPW form. We show below that our DR estimator $\hat{r}(t)$ guarantees efficiency gain over the IPW estimator. Let $\tilde{r}(t)$ denote the IPW esimator of the reward corresponding to $\bar{b}(t)$ and let $\hat{r}(t)$ be the DR estimator as defined in the text. Hence, $\tilde{r}(t) = \frac{r_{a(t)(t)}}{N\pi_{a(t)}(t)}$ and $\hat{r}(t) = \tilde{r}(t) + \bar{b}(t)^T\hat{\beta}(t-1) - \frac{b_{a(t)}(t)^T\hat{\beta}(t-1)}{N\pi_{a(t)}(t)}$. The variance of $\tilde{r}(t)$ given filtration $\mathcal{F}_{t-1}$ is $\mathbb{E}\Big[\Big\{\frac{\eta_{a(t)}(t)}{N\pi_{a(t)}(t)} + \frac{b_{a(t)}(t)^T\beta}{N\pi_{a(t)}(t)} - \bar{b}(t)^T\beta\Big\}^2\Big|\mathcal{F}_{t-1}\Big]$. Due to Assumption 4 on the sub-gaussian error η, the first term $\eta_{a(t)}(t)/N\pi_{a(t)}(t)$ is bounded by a constant when $\pi_{a(t)}(t) \geq O\big(\frac{1}{N}\sqrt{(\log d + \log t)/t}\big)$. However, the second term $\frac{b_{a(t)}(t)^T\beta}{N\pi_{a(t)}(t)}$ can still be large. Constant variance is important because the variance ($\tilde{R}^2$ in the text) appears in the regret bound in Theorem 4.1. To achieve a constant variance, we need $\pi_{a(t)}(t)$ be larger than a predetermined constant value, $\frac{1}{N}p_{min}$. **Simply truncating the value will produce a biased estimate when $\pi_{a(t)}(t)$ is actually smaller than $\frac{1}{N}p_{min}$, and the Lasso property (Lemma 3.2) will not hold either.** If we instead directly constrain $\pi_{a(t)}(t)$ to be larger than $\frac{1}{N}p_{min}$, this will lead to suboptimal choices of arms and Theorem 4.1 will not hold. In contrast, the variance of $\hat{r}(t)$ is $\mathbb{E}\Big[\Big\{\frac{\eta_{a(t)}(t)}{N\pi_{a(t)}(t)} + \frac{b_{a(t)}(t)^T\beta^*}{N\pi_{a(t)}(t)} - \bar{b}(t)^T\beta^*\Big\}^2\Big|\mathcal{F}_{t-1}\Big]$, where $\beta^* = (\beta - \hat{\beta}(t-1))$. Now, we have a constant variance under $\pi_{a(t)}(t) \geq O\big(\frac{1}{N}\sqrt{(\log d + \log t)/t}\big)$ with high-probability due to the fact $||\beta^*||_1 \leq O\big(\sqrt{(\log d + \log t)/t}\big)$ with high-probability ($\because$ Lemma 3.2 on observations until $t-1$). The regret bound in Theorem 4.1 holds under (i) $\pi_{a(t)}(t) \geq O\big(\frac{1}{N}\sqrt{(\log d + \log t)/t}\big)$ but not under (ii) $\pi_{a(t)}(t) \geq \frac{1}{N}p_{min}$. (We skipped details on this part, but the relevant part is the bound on $\sum_{t=z_T}^{T} m_t$ in Section 4.1.) Also note that the restriction (i) is much weaker than (ii) since the term $\sqrt{(\log d + \log t)/t}$ converges to 0 as t increases, inducing exploration in early stages and greedy choices when t is large.

Comparison with Bastani and Bayati (BB): As Reviewer 2 mentioned, **our method and BB deal with different settings and direct comparison may not be meaningful. There are no previous works dealing with our setting of large N.** Having said, we agree with Reviewer 2 that if we use our method **in the setting of BB, when N is not that large**, it will produce a regret of order $O(s_0 \log(dNT)\sqrt{T})$ which is larger than that of BB. We stress that our method and the method of BB are designed for different settings. BB guarantees the best performance when number of arms is moderate, and when we can assume that each arm has different regression parameter. In this case, BB has an advantage.However, there are cases where the number of arms is very large. Especially, when we recommend a news article or a shopping item, or when we place an advertisement on the web page, the number of possible action selections is very large. Moreover, the lists of news articles, shopping items, or advertisements change day by day (even change a lot in a single day). In this case, it would not be feasible to assign a different parameter for every new incoming item, and also to conduct forced-sampling of arms according to a predetermined schedule. Therefore, in cases where the number of arms is large and the arm set changes with time, our method will show advantage over BB.

Hyperparameters: In online learning, it is difficult to simultaneously tune the hyperparameters and achieve high reward, and it is crucial to have a smaller number of hyperparameters. Due to difficulty in simultaneous tuning and optimization, at the beginning rounds, we should sacrifice learning the tuning the hyperparamters. In this stage, the accumulation of rewards remains slow because we do not know yet which values of hyperparameters are best suited to our algorithm. When we tune the values by grid search, then the amount of time required for tuning is exponential in the number of tuning parameters. Therefore, having one less tuning parameter can result in much short time required for tuning process.

Real data experiments: We have some encouraging real data example using the YAHOO news article recommendation log data. We will include the results in the supplementary material in the future.