We thank the reviewers for their careful reading of the paper. First, we would like to emphasize that all three reviewers agree the neural JSDE is an interesting model that introduces discrete events into continuous latent ODE framework. Reviewer 1 and Reviewer 2 further describe the model as clean/parsimonious, and they both state that neural JSDE have a very broad range of applications.

That being said, we would like to respond to the specific suggestions provided by Reviewers 1 and 2 on (1) including neural ODE as baselines and (2) details of the experimental setup; as well as the concerns raised by Reviewer 3 on (3) incremental prediction improvements on marker prediction and (4) the derivation of Eq. (11). The remaining minor points in the reviews should be easy to address in a final version of the paper.

(1) Including the original neural ODE as a baseline. Since neural ODE cannot model event effects, we thought comparing against it would be unfair. Essentially, neural ODE only captures Poissonian behavior in time series. That being said, we can still use the model to predict the conditional intensity of the point process datasets — we would just expect the model to only work well for Poisson process which does not depend on the event history. Indeed, the results in Table 1, which shows the mean absolute percentage errors (MAPE), demonstrates this. The accuracy of neural ODE for the Poisson process is on par with our neural JSDE. However, for the Hawkes process (Exponential), Hawkes process (Power-Law), and self-correcting process, neural ODE gives much larger predictions errors. Again, we should expect this behavior — a primary goal of neural JSDE is to account for the event history, a shortcoming of the original neural ODE framework.

Table 1: Neural ODE / JSDE predicted conditional intensity error.

MAPE	ODE	JSDE
Poisson	1.2	1.3
Hawkes (E)	172.0	5.9
Hawkes (PL)	91.4	17.1
Self-Correcting	27.2	9.3

(2) Details on the experimental setup. As we discuss here, neural JSDE introduces minimal additional architecture to the neural ODE framework. We will add the following details to the paper. For the point processes experiments, we used a 5-dimensional latent state and parameterized the dynamics function f, the jump function w, and the intensity function λ using MLPs with one hidden layer and 20 hidden units. For the social/medical datasets, we used a 20/64-dimensional latent state and parameterized the functions with two-hidden-layer MLPs with 32/64 hidden units. The run time of neural JSDE is dominated by the underlying neural ODE dynamics and is therefore higher than the baseline RNN network, but the neural JSDE is much better suited for modeling complex dynamics and irregularly spaced time series, as demonstrated by its performance gain.

Reviewer 1 also noted the Poisson dataset does not fit well to the Poisson process. This is because the average sequence length in the Poisson dataset is relatively small (*e.g.* 20 events for Poisson *vs.* 200+ events for self-correcting process). The time series modeling software that we used is designed for long event sequences and ignores the idle time after the last event. We find that using longer Poisson sequences remedies this issue.

(3) Results on marker prediction. The main contribution of this work is introducing event handling into the continuous neural ODE framework with minimal computational overhead, while still maintaining its memory efficiency and ability to train end-to-end. We have demonstrated its strong performance in a range of settings in an attempt to highlight the flexibility of the framework. For the specific case of the Stack Overflow and MIMIC datasets, the baselines we compare against (RMTPP and neural Hawkes) are already quite strong, and prior advances in prediction on these datasets is also incremental [1, 2]. Our goal with this experiment is to demonstrate the modeling capability of neural JSDE rather than to blow competitive baselines out of the water. We will emphasize this in paper revisions.

(4) Derivation of Eq. (11). Reviewer 3 claims that Eq. (11) appears in [3]. We assume that Reviewer 3 is referring to the "SDE for Hawkes process" equation in section 3.2 of that paper. It turns out that Eq. (11) is considerably different. First, it uses a neural network to parameterize the continuous dynamics and jump; and second, it specifies the time evolution of latent state as opposed to directly modeling the conditional intensity or opinion. Moreover, the general idea of a "jump stochastic differential equation" is established with a long history in the financial mathematics literature. We felt that this was a standard-enough concept, but we are happy to include a reference to [3] in the final version of the paper, as it is certainly still relevant to our research.

References

[1] Hongyuan Mei and Jason M Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems 30*. 2017.

[2] Nan Du et al. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[3] Yichen Wang et al. A stochastic differential equation framework for guiding online user activities in closed loop. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 2018.