A APPENDIX

A.1 Implementation Details

We used a 100 dimensional embedding layer to represent the ICD10 codes as dense vectors followed by a single-layer LSTM with 100 hidden units. In a pilot study, we tuned the embedding dimension and the number of LSTM units based on the model performance on the validation set. We empirically found that the model performance is robust to these hyperparameters. To make sure that the α and λ parameters are positive, we reparametrized them using the exponential function e.g. $\alpha = \exp(\alpha')$. For learning, we used the Stochastic Gradient Descent algorithm with momentum set to 0.9 (Sutskever et al., 2013). We initialized the α and λ parameters using the learnt parameters of the Hawkes process. We empirically found that initializing the parameters with sensible values often accelerates the optimization process. All other parameters are initialized to random values from N(0, 1). We performed early stopping based on prediction loss on the held-out validation set. The model is implemented in PyTorch and is available at `https://bitbucket.org/mvdschaar/mlforhealthlabpub/src/master/alg/dynamic_disease_network_ddp/`.

A.2 Comparison of Intensity Functions

Figure 1 contains a illustrative example of the difference in intensity functions between DDP and the Hawkes process. In Hawkes processes, including cHawk, each event's contribution to the intensity function has the same magnitude and shape. In DDP, the event's contribution depends on the weights given by the LSTM (yellow circle), which in turn depends on the full event history. This allows for a richer representation of the overall intensity function and the relationship between events.

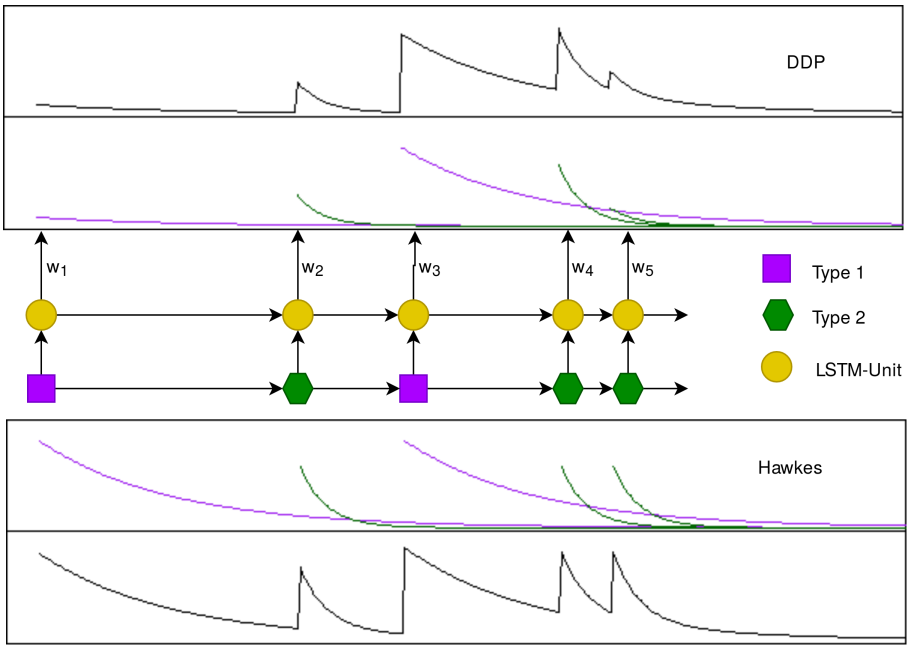

Figure 1: A side-by-side comparison between the intensity functions of DDP and the Hawkes process. The event timeline in the middle shows the occurrence of two types of events (polygons). The weights given by the LSTM at each step are represented as yellow circles. The coloured lines show each event's contribution to the intensity function of the type 1 event. The black lines are the overall intensity function of type 1 event, which are the sum of the coloured lines.

A.3 Disease Progression Is Non-Markovian

Figure 2 shows an example based on real patient data. All patients illustrated here had Pneumonia but depending on whether they had previous comorbidities in the respiratory system or the circulatory system, their chances for developing new comorbidities in the respective systems will differ. This example shows that the Markovian

assumption is clearly violated in disease network modelling. Our approach to model disease network as a dynamic graph captures the observation that past comorbidities have varying effects on the future diseases.

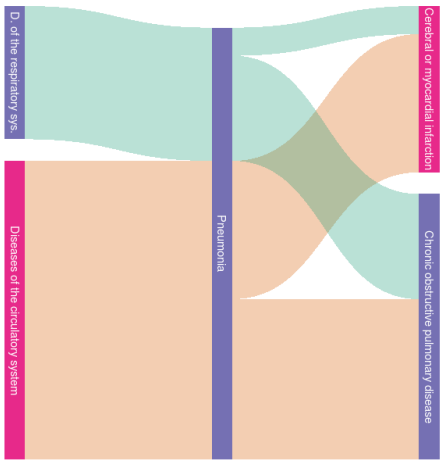

Figure 2: Disease pathway is moderated by previous comorbidities. The diseases are colour coded using their ICD10 chapters.

A.4 ICD10 Codes Reference

We aggregated the original ICD codes to 3-digit level (I25.10 becomes I25) and chose the most prevalent ones in the cohort to train and evaluate the model. These codes represent the most common comorbidities among colorectal cancer patients (various gastrointestinal disorders and cardiovascular diseases).

Table 1: ICD10 Codes Reference Table.

ICD10 Code	Disease Name
I50	Heart failure
N39	Other disorders of urinary system
A41	Other Sepsis
D12	Neoplasm of digestive organs
E86	Volume depletion
I25	Chronic ischemic heart disease
K63	Other diseases of intestine
K83	Other diseases of biliary tract

A.5 Schematics of the Validation Setup

The primary dataset of colorectal cancer patients was randomly split into training (60%), validation (20%) and testing (20%) sets. Hyperparameter tuning was based on the validation set, whereas model evaluation was based on the test set. Out-of-domain data of stomach cancer patients was used as a separate test set.

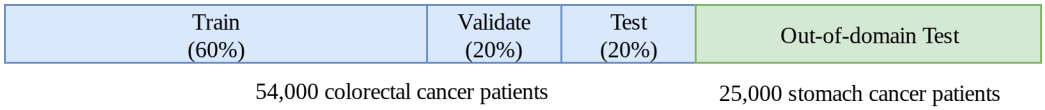

Figure 3: Schematics of the validation setup.

A.6 Predictive Performance over Different Time Horizon

We stratified the test data set based on the prediction time t_{k+1}, and evaluated the AUC score in each stratum. The results are presented in figure 4 and 5 for colon cancer and out-of-domain stomach cancer patients respectively[1]. Unsurprisingly, we observe that in general the prediction accuracy decreases as the time horizon increases.DDP outperforms the benchmarks in the majority of cases especially in the longer time horizon. The long-term prediction accuracy is highly important in disease network modelling because many interesting patterns of disease interaction evolve gradually through time and they may not be observable in the short time window.

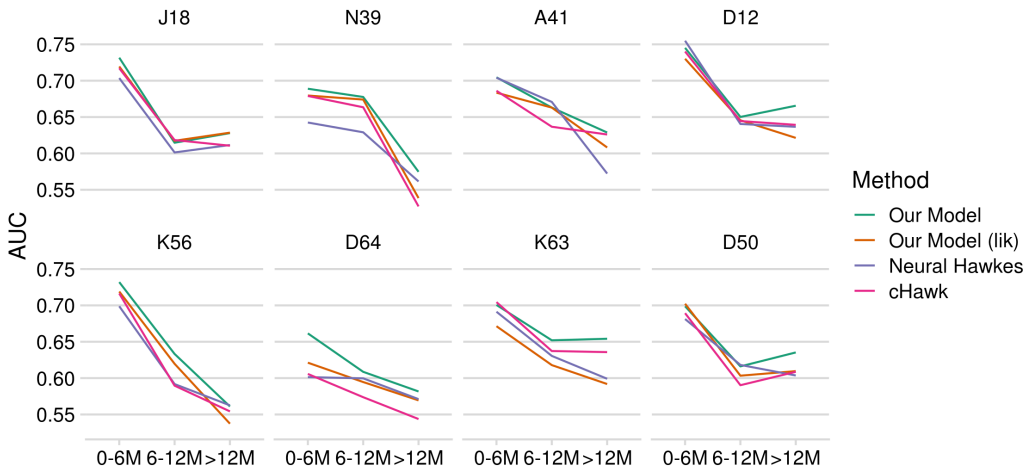

Figure 4: Prediction AUC for different time windows: 0-6 months, 6-12 months and greater than one year.

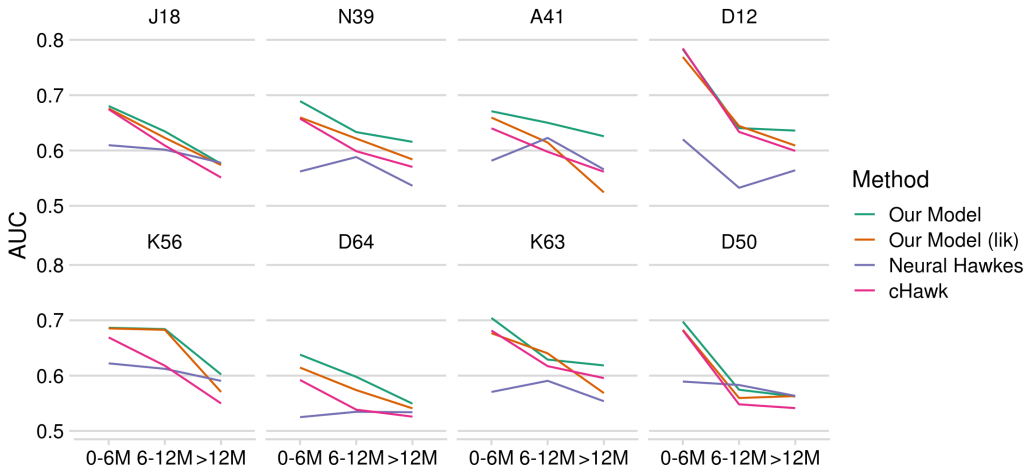

Figure 5: Prediction AUC for patients with stomach cancer.

A.7 Visualization of Disease Networks for Colorectal and Stomach Cancer Patients

We randomly sampled a subset of patients from the colorectal and stomach cancer patients and calculated their distance matrix based on the Jaccard index. We applied t-SNE to the distance matrix and embed each patent's disease network in $\mathbb{R}^2$. The results are visualized in Figure 6, which indicates no clear difference between the

[1]Since Charlson Score's performance is worse than all models by a big margin it is not shown in the figure.

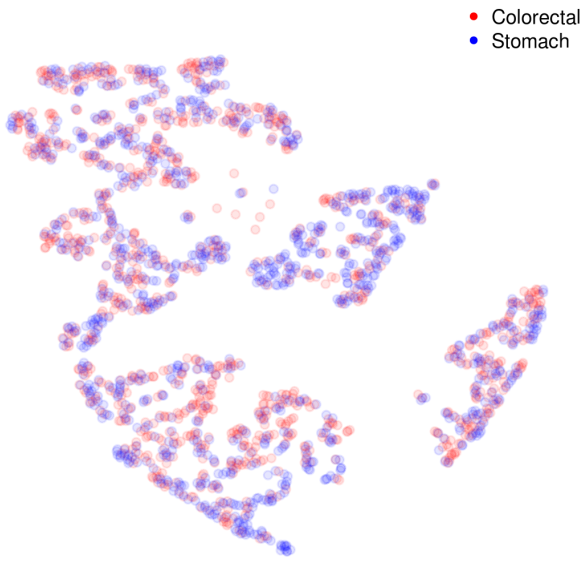

Figure 6: t-SNE Visulization of the Dynamic Disease Networks. There is no clear separation between stomach and colorectal cancer patients.

networks for stomach and colorectal cancer patients. In other words, the disease networks learnt by DDP applies to both sets of patients and is **invariant** across cancer sites.

A.8 Dynamic Disease Network on Individual-level

Now, we prodivde an additional example to illustrate how the dynamic disease network can be used in the clinical setting to help doctors gain a better understanding about the comorbidity pathway for individual patient, which may eventually lead to better treatment plans.

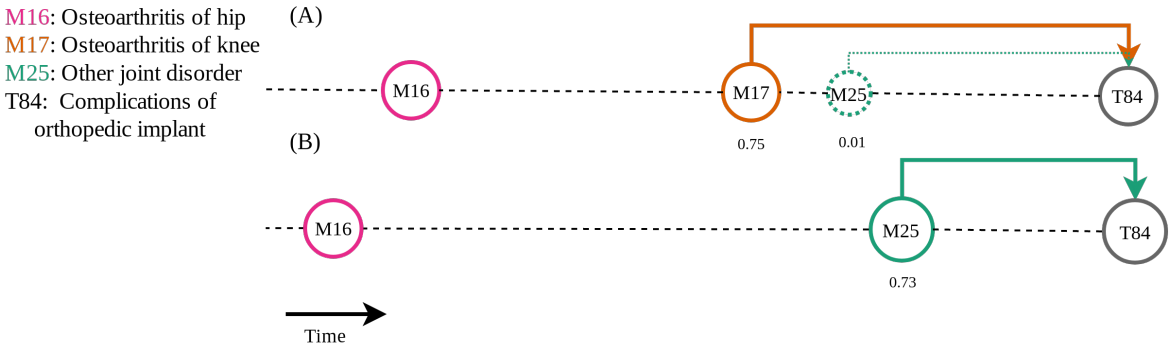

Figure 7: The disease network for two real patients A and B plotted on a timeline. For patient A, the impact of M25 on T84 was inhibited by the presence of M17. For patient B, no inhibition was present. The RNN output was shown below the circles. To protect privacy, the time between diseases is not drawn to scale.

Figure 7 gives an example of two patients, A and B. The sequence of diseases are plotted on a timeline. Both patients had a history of hip osteoarthritis (M16) and other joint disorders (M25). However, patient A also had knee osteoarthritis (M17) shortly before M25. It is well-known that the standard treatment for late stage osteoarthritis, hip or knee replacement, leads to various postoperative complications (T84) depending on existing comorbidities (Gupta et al., 2001) (Greenfield et al., 1993). Therefore, whether the doctor will recommend joint replacement surgery for each patient partly depends on how much preexisting conditions elevate the risk of

complications. The dynamic disease network learnt on individual level helps answer this question.

In fact, on the population level, DDP finds that all three morbidities (M16, M17 and M25) strongly increase the risk of postoperative complications. However, the population level disease relationship only captures the average scenario and it does not *always* hold for each individual. In contrast, the dynamic network DDP constructed for each patient takes into account their unique comorbidity history. In this example, the network for patient A suggests that the presence of knee osteoarthritis (M17) inhibits M25's impact on postoperative complications. In a sense, the impact of M25 was *explained away* by M17. On the contrary, the network for patient B does not show any inhibition effect and M25 is still believed to increase the risk of postoperative complications.

If we hadn't taken into account the existing comorbidity pathway, we would have treated all previous comorbidities equally as is done by the Hawkes process, and therefore overestimated patient A's risk of having postoperative complications. This could lead to sub-optimal treatment plan for the patient.

A.9 Static Population Level Disease Network

In addition to producing dynamic comorbidity network, DDP can also learn a static population-level graph by averaging out the time-varying components. Figure 8 shows the static comorbidity network learnt by DDP from the real patient data. It is worth highlighting that the graph is learnt in a fully-automated way without any input of prior medical knowledge. We applied a standard community detection algorithm (Blondel et al., 2008) to the graph and colored the diseases based on the cluster membership.

One can immediately observe that the diseases in the same cluster usually share the initial letter, which corresponds to the chapter code in the ICD10 system. For example, all the M codes are related to bone or connective-tissue disorders and they are all assigned to the red cluster on the top of the graph. The diseases I20, I23, I25, and R07 are closely linked together (the blue cluster on the left of the graph) and they are all related to heart diseases.

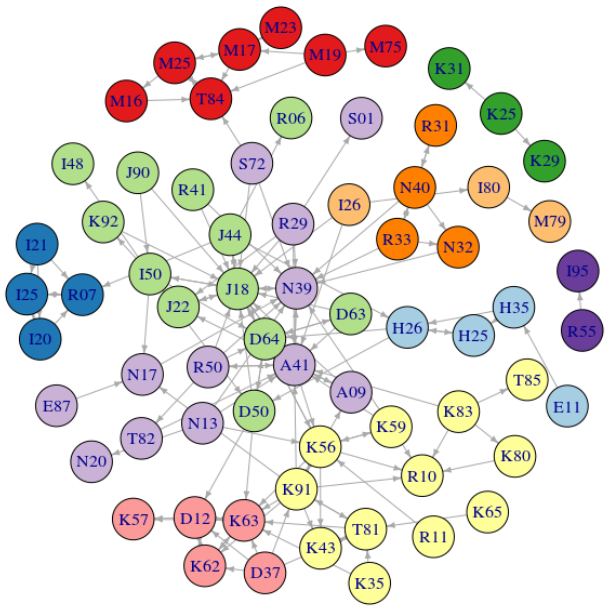

Figure 8: The population level static disease network. Node colour represents the cluster membership of the disease. The initial letter for the ICD code represents a broad classification of diseases. For visualization purposes, we have removed edges with weights below a certain threshold.

A.10 Simulation Study

In this simulation study, we verify that DDP is able to recover the static network entailed by the Hawkes process. We consider random samples from a Hawkes process with randomly chosen parameters. We proceed by fitting a

Hawkes process on the sample by maximizing likelihood until convergence and measure the L1 distance between the true and the learned infectivity matrix (sum(abs(X-Y))). This serves as a lower-bound on error because there is no model mis-specification (red dotted line in Figure 9). We then train DDP and track the L1 distance between the DDP network (eq 12) and the ground truth during training. We observe that after a rapid drop at the beginning, the error gradually decreases and approaches the lower bound. This means that DDP is able to uncover the Hawkes process network.

The code to reproduce this simulation is available online. The simulation parameters are as follows:

1. *Number of sequences*: 5000
2. *Sequence length*: uniformly selected between 5 to 20 for each sequence
3. *Number of event types*: 50
4. *Ground truth infectivity matrix*: randomly drawn from Uniform(0, 1)
5. *Initial DDP infectivity matrix*: randomly drawn from Uniform(0, 1)
6. *Initial values for DDP LSTM weights*: randomly drawn from Uniform(-0.1, 0.1) (Pytorch default)
7. *DDP LSTM hidden layer size*: 100
8. *SGD learning rate*: 0.001
9. *SGD momentum*: 0.9
10. *Size of mini-batch*: 100

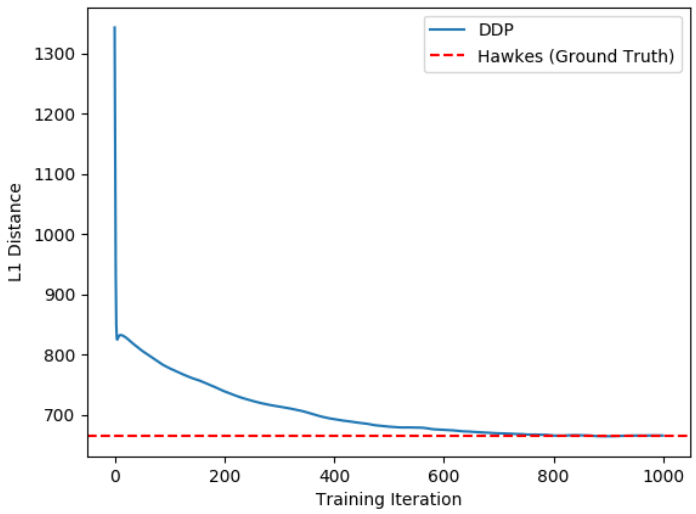

Figure 9: Simulation Results: The DDP network approaches distance lower bound during training.

A.11 Additional Experiments on MIMIC Data

Data Set We evaluate the predictive performance of DDP on the publicly available Medical Information Mart for Intensive Care data (MIMIC). The data set has been used by many authors as a test bed for new algorithms including Du et al. (2016) and Mei & Eisner (2017). To ensure a fair comparison with existing work, we use the pre-processced data curated by Mei & Eisner (2017). The data set contains a collection of de-identified clinical records of Intensive Care Unit (ICU) patients. During a patient's stay in ICU, a total of 75 types of events are tracked along with their timestamps. We're interested in predicting what event will happen next given the events happened in the part.

Evaluation Setting We adopt the same evaluation setting as the previous works. The models predict the identity of the next event given the occurrence time of the event, and they are evaluated based on the error rate (also known as 0-1 loss). We run five-fold cross validation to estimate the variation of the loss metric. Within each fold, the original training data is further split into a new training set and a validation set for hyper-parameter tuning and early stopping.

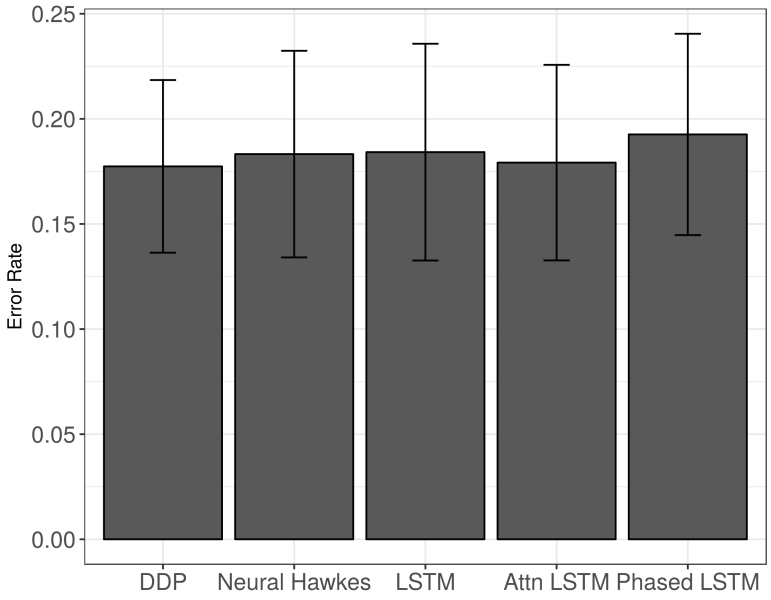

Figure 10: Algorithm Performance on MIMIC data.

Benchmarks We used a variety of highly-performant algorithms as benchmark. **Neural Hawkes** (Mei & Eisner, 2017) is a point process model that has achieved the state-of-the-art performance on event prediction. **LSTM** (Gers et al., 2000) is a RNN that has become the standard of sequence modelling. To capture the variable duration between events, we use the time gap between events as an additional input dimension to LSTM. **RETAIN** (Choi et al., 2016) (labelled as Attn LSTM in the figure) employs the temporal attention mechanism akin to RPPN model in Xiao et al. (2019). Finally, **Phased LSTM** is a continuous time RNN that naturally handles the unequal spacing between events.

Results The results are presented in Figure 10. As we can see, DDP achieves predictive performance on par with, if not slightly better than, the state-of-the-art benchmarks. However, compared with the benchmarks, DDP enjoys the unique advantage of being able to explicitly learn the relationship between events and represent those relationship as a dynamic graph.

References

VD Blondel, JL Guillaume, R Lambiotte, and E Lefebvre. Fast unfolding of community hierarchies in large network, 2008. *J. Stat. Mech. P*, 1008, 2008.

Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*, pp. 3504–3512, 2016.

Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. 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*, pp. 1555–1564. ACM, 2016.

FA Gers, J Schmidhuber, and F Cummins. Learning to forget: continual prediction with lstm. *Neural computation*, 12(10):2451, 2000.

Sheldon Greenfield, Giovanni Apolone, Barbara J McNeil, and Paul D Cleary. The importance of co-existent disease in the occurrence of postoperative complications and one-year recovery in patients undergoing total hip replacement. comorbidity and outcomes after hip replacement. *Medical care*, 31(2):141–154, 1993.

Rakesh M Gupta, Javad Parvizi, Arlen D Hanssen, and Peter C Gay. Postoperative complications in patients with obstructive sleep apnea syndrome undergoing hip or knee replacement: a case-control study. In *Mayo Clinic Proceedings*, volume 76, pp. 897–905. Elsevier, 2001.

Hongyuan Mei and Jason M Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, pp. 6754–6764, 2017.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pp. 1139–1147, 2013.

Shuai Xiao, Junchi Yan, Mehrdad Farajtabar, Le Song, Xiaokang Yang, and Hongyuan Zha. Learning time series associated event sequences with recurrent point process networks. *IEEE transactions on neural networks and learning systems*, 2019.