Table 5: Hyper-parameters used by the models in each predictive modeling experiments

Experiment	Model	m	r	l	L_2	Dropout rate
Disease progression modeling (Sutter data)	GRAM+	500	500	100	0.0001	0.6
	GRAM	500	500	100	0.0001	0.6
	RandomDAG	500	500	100	0.0001	0.6
	RNN+	550	500		0.0001	0.6
	RNN	550	500		0.0001	0.6
	SimpleRollUp	500	500		0.0001	0.4
	RollUpRare	500	500		0.0001	0.2
Disease progression modeling (MIMIC-III)	GRAM+	400	400	100	0.0001	0.6
	GRAM	400	400	100	0.001	0.6
	RandomDAG	400	400	100	0.001	0.6
	RNN+	550	400		0.001	0.8
	RNN	550	400		0.001	0.8
	SimpleRollUp	400	400		0.001	0.6
	RollUpRare	400	400		0.0001	0.0
HF prediction (Sutter HF cohort)	GRAM+	200	100	100	0.001	0.6
	GRAM	200	100	100	0.001	0.6
	RandomDAG	300	100	200	0.001	0.6
	RNN+	200	100		0.0001	0.6
	RNN	200	100		0.001	0.6
	SimpleRollUp	300	200		0.001	0.4
	RollUpRare	100	100		0.001	0.6

C HYPER-PARAMETER TUNING

We define five hyper-parameters for GRAM:

- dimensionality m of the basic embedding $\mathbf{e}_i$: [100, 200, 300, 400, 500]
- dimensionality r of the RNN hidden layer $\mathbf{h}_t$ from Eq. (4): [100, 200, 300, 400, 500]
- dimensionality l of $\mathbf{W}_a$ and $\mathbf{b}_a$ from Eq. (3): [100, 200, 300, 400, 500]
- L_2 regularization coefficient for all weights except RNN weights: [0.1, 0.01, 0.001, 0.0001]
- dropout rate for the dropout on the RNN hidden layer: [0.0, 0.2, 0.4, 0.6, 0.8]

We performed 100 iterations of the random search by using the above ranges for each of the three prediction experiments. For sequential diagnoses prediction on Sutter data, we used 10% of the training data to tune the hyper-parameters to balance the time and search space. To match the baselines' number of parameters to GRAM's, we add 550 to the list of m's possible values. This will make the baseline's largest possible number of parameters comparable to the GRAM's largest possible number of parameters.

For SimpleRollUp and RollUpRare, the number of input codes is smaller than other models due to the grouping. Therefore, to match their largest possible number of parameters to GRAM's, we need to add much larger values to m. However, after preliminary experiments, as expected, setting m to too large a value degraded the performance due to overfitting. Since the number of input codes decreased due to the grouping, increasing the dimensionality of $\mathbf{e}_i$ is not a logical thing to do. Therefore, for SimpleRollUp and RollUpRare, we use the same list of values for m as other baselines.

Table 5 describes the final hyper-parameter settings we used for all models for each prediction experiments.