Feature Restricted Group Dropout for Robust Electronic Health Record Predictions

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recurrent neural networks are commonly applied to electronic health records to capture complex relationships and model clinically relevant outcomes. However, it is commonplace for the covariates in electronic health records to change distributions. This work extends restricted feature interactions in recurrent neural networks to address foreseeable and unexpected covariate shifts. We extend on the previous work by 1) Introducing a deterministic feature rotation so that hyperparameter tuning can search through all combinations of features, 2) Introduce a sub-network specific dropout to ablate the influence of entire features at output of the hidden network, and 3) Extend the feature restrictions to the GRU-D network, which has been shown to be a stronger baseline for covariate shift recovery. We show that feature restricted GRU-D's may be more robust to certain perturbations. Manual intervention was not needed to confer robustness. Despite this, the LSTM was still the best model in nearly 50% of the cases.

1 Introduction

In healthcare, it is inevitable that the generation of the data reflected in electronic health records (EHRs) will evolve; new equipment is added with different margins of error, suppliers of laboratory tests will switch, and clinicians administering the tests change. Under these anticipated model changes, how can we be confident that the deep learning model we have deployed can continued to be used until it is remediated or replaced? We define this problem as one of a sudden covariate change, $p(x_i) \rightarrow q(x_i)$. sudden covariate change is distinct from continual learning setting as the covariate changes rapidly. Sudden covariate shift is more mild than complete domain shift $p(X) \rightarrow q(X)$. This makes it challenging, because it raises doubts as to whether or not a model should be taken offline. If a model has been demonstrated to improve care, and the covariate shift is marginal, it may be unnecessary or even harmful to withhold the predictions. We also permit $p(y|x_i) \neq q(y|x_i)$, such that even if the old covariate distribution is able to be mapped to the new distribution, the patterns in which they are used may shift to reflect healthcare usage. Such patterns might be present if a new covariate differs in cost, complexity, or speed.

Current methods to mitigate covariate shift often focus on gradual shift. Work has been done to causally predict which feature will be robust during test time [1]. However this setting assumes that features are already experiencing distributional anomalies. Once a small subset of the new distribution are available, it is possible continually fine-tune the model, however this could lead to catastrophic forgetting [2]. It is also possible to initiating the model from scratch and upweight the samples in the new distribution [3], however this comes at a training cost and still requires that the model be taken offline while the new distribution is being collected. Others have sought to apply batch normalisation during inference [4], or Bayesian neural networks

Submitted to Workshop on Robustness in Sequence Modeling at the 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Previous work has shown that restricting the feature interactions in the latent space improves model performance when generalising to an unseen test set [5]. We extend this method by adding a seeded rotation to permit all features to interact with each other. Crucially, we train the model with a sub-network specific dropout such that each sub-network containing feature x_i can be simultaneously omitted. Finally, we introduce the masking method from Zhang et al. [5] into the GRU-D model [6], which has shown to be more robust to dramatic distributional changes than LSTM networks [7].

2 Methods

Data is Sourced from `GEMINI`, a data warehouse containing 214837 General Internal Medicine inpatient encounters from 6 different sites [8]. The data is extracted and prepared by replicating protocols from previous EHR extractions [9]. Site characteristics are shown in Table 1. The target selected to demonstrate this task is a delirium label with 30% positively rate. The data are split by time such that 80% of the training data falls before the training end date, 10% of the data falls between the training end date, and the validation end date, and the remaining 10% occurs after the validation end date and is relegated to the test set. A GRU-D architecture [6] embeds the site-specific model, while a LSTM network decodes to the target. The parameters for each site, including the traditional dropout probability, the number of feature interaction groups, the network widths, and the number of decoder layers are selected by taking the network with the best AUROC on a prospective validation set. Both the unperturbed, feature-restricted GRU-D and the standard GRU-D models are allowed to choose their optimal architectures independently. These models are compared to an LSTM architecture which has no method to deal with faulty covariates.

To evaluate the robustness of the model, data are randomly perturbed by randomly resampling the feature, flipping the sign of the feature, or by linearly scaling the feature. We show the performance of the LSTM and GRU-D under these conditions. As a remedy, we intervene by consciously decaying the input unit of this feature in the GRU-D to the training mean. The feature-restricted GRU-D is also evaluated on the perturbed input. Finally, the sub-networks involving that feature are "turned off" by intervening with a sub-network specific dropout.

	patients	encounters	sex	age	mortality	LOS	palliative	delirium	
	count	count	mean	med.	mean	Days	mean	-	+
Site									
THPM	46074	71166	0.49	75	0.06	9.9	0.07	360	181
SBK	34237	53091	0.52	77	0.04	9.5	0.06	656	296
UHNTG	30779	48690	0.47	66	0.05	8.5	0.13	412	93
SMH	23071	39894	0.42	66	0.04	9.3	0.03	762	231
UHNTW	25072	42343	0.49	74	0.05	9.6	0.09	369	123
THPC	31101	49080	0.54	74	0.05	11.0	0.06	309	70
MSH	24503	36749	0.53	69	0.05	9.3	0.11	0	0

Table 1: Summary of the sites available in the GEMINI dataset.

3 Results

The performance of the LSTM, GRU-D, and feature restricted GRU-D are demonstrated in Table 2. The validation performance under the GRU-D restricted feature interaction tends to outperform GRU-D and LSTM. However the generalisation to the prospective test set tends to be weaker with several notable drops in performance from validation time to test time. This could be attributed to the sites having few limited numbers of samples to do model selection, or to form robust measurements of the AUROC at both validation time and test time. Alternatively, some of the performance drop can be attributed to real world data-set shift, such as with the introduction of COVID-19 into the hospital system during test time (but not during training or validation time).

To evaluate the robustness of the model to sudden incremental covariate shifts, one feature is perturbed at a time. The test set performance of the GRU-D, intervened-upon GRU-D, uncontrolled feature restricted GRU-D, and intervened-upon feature-restricted GRU-D are demonstrated in Tables 3

	Size		Validation			Test		
	Validation	Test	LSTM	GRU-D	GRU-D R	LSTM	GRU-D	GRU-D R
THPM	63	49	0.735	0.744	**0.849**	**0.761**	0.623	0.691
SBK	68	80	0.740	0.726	**0.749**	**0.742**	0.686	0.622
UHNTG	35	50	0.672	0.672	**0.836**	0.542	0.366	**0.807**
SMH	88	94	0.773	**0.774**	0.769	**0.735**	0.641	0.587
UHNTW	44	48	0.753	**0.810**	0.773	0.445	**0.634**	0.485
THPC	57	60	**0.751**	0.721	0.745	0.525	0.523	**0.551**

Table 2: AUROC scores for the LSTM, GRU-D, and GRU-D with feature restriction across 6 sites for delirium prediction.

& 4& 5 for flipped, resampled, and linearly scaled perturbations. Here we see that both the GRU-D and GRU-D Restricted are capable of handling feature corruption as well as manual intervention. The corruption itself, however is occasionally devastating, such as the case with UHNTG GRU-D Restricted model dropping to near random performance with any type of corruption.

model	LSTM		GRU-D		GRU-D *	GRU-D R		GRU-D R *
corruption	None	flipping	None	flipping	flipping	None	flipping	flipping
metric	AUC	AUC	AUC	AUC	AUC	AUC	AUC	AUC
site								
THPM	0.76	0.58 ± 0.00	0.62	0.64 ± 0.02	0.64 ± 0.02	0.69	0.59 ± 0.08	0.57 ± 0.07
SBK	0.74	0.60 ± 0.01	0.69	0.61 ± 0.11	0.60 ± 0.10	0.62	0.46 ± 0.05	0.48 ± 0.06
UHNTG	0.54	0.62 ± 0.01	0.37	0.39 ± 0.09	0.43 ± 0.07	0.81	0.53 ± 0.01	0.53 ± 0.01
SMH	0.74	0.57 ± 0.01	0.64	0.62 ± 0.02	0.62 ± 0.02	0.59	0.62 ± 0.05	0.62 ± 0.05
UHNTW	0.45	0.44 ± 0.05	0.63	0.51 ± 0.11	0.51 ± 0.11	0.49	0.50 ± 0.11	0.50 ± 0.10
THPC	0.53	0.43 ± 0.01	0.52	0.45 ± 0.06	0.45 ± 0.06	0.55	0.54 ± 0.03	0.52 ± 0.07

Table 3: AUROC scores for the GRU-D and GRU-D with feature restriction across 6 sites for delirium prediction. Corruption is applied to features through flipping the sign. * indicates manual intervention.

model	LSTM		GRU-D		GRU-D *	GRU-D R		GRU-D R *
corruption	None	resample	None	resample	resample	None	resample	resample
metric	AUC	AUC	AUC	AUC	AUC	AUC	AUC	AUC
site								
THPM	0.76	0.58 ± 0.00	0.62	0.61 ± 0.03	0.61 ± 0.03	0.69	0.58 ± 0.07	0.55 ± 0.04
SBK	0.74	0.57 ± 0.01	0.69	0.57 ± 0.09	0.57 ± 0.09	0.62	0.41 ± 0.03	0.47 ± 0.04
UHNTG	0.54	0.62 ± 0.01	0.37	0.35 ± 0.08	0.39 ± 0.07	0.81	0.53 ± 0.01	0.53 ± 0.01
SMH	0.74	0.57 ± 0.01	0.64	0.62 ± 0.03	0.62 ± 0.03	0.59	0.59 ± 0.02	0.60 ± 0.03
UHNTW	0.45	0.49 ± 0.03	0.63	0.56 ± 0.09	0.56 ± 0.09	0.49	0.50 ± 0.10	0.48 ± 0.10
THPC	0.53		0.52			0.55		

Table 4: AUROC scores for the GRU-D and GRU-D with feature restriction across 6 sites for delirium prediction. Corruption is applied to features through resampling the corrupted column. * indicates manual intervention.

In general, When the LSTM was the top performer, it lost $0.22(0.18 \cdot 0.24, 95\%CI)\%$,$0.23(0.21 \cdot 0.24, 95\%CI)\%$ and $0.14(-0.02 \cdot 0.24, 95\%CI)\%$ of its performance for flipping, resampling, and scaling perturbations, respectively. In contrast, GRU-D lost it lost $0.14(-0.02 \cdot 0.43, 95\%CI)\%$,$0.12(-0.04 \cdot 0.42, 95\%CI)\%$ and $0.32(-0.02 \cdot 0.56, 95\%CI)\%$ of its performance for flipping, resampling, and scaling perturbations, respectively, when it was the top performer. Compare this to the GRU-D with restriction which lost $0.13(-0.10 \cdot 0.36, 95\%CI)\%$,$0.15(-0.05 \cdot 0.35, 95\%CI)\%$ and $0.15(-0.18 \cdot 0.34, 95\%CI)\%$ of its performance for flipping, resampling, and scaling perturbations, respectively. GRU-D was the most susceptible model to scaling noise, while the LSTM was the most susceptible model to flipping and resampling. Neither intervention taken on the GRU-D or GRU-D Restricted models improved the performance. After corruption, the LSTM

model	LSTM		GRU-D		GRU-D *	GRU-D R		GRU-D R *
corruption	None	scaling	None	scaling	scaling	None	scaling	scaling
metric	AUC	AUC	AUC	AUC	AUC	AUC	AUC	AUC
site								
THPM	0.76	0.58 ± 0.00	0.62	0.61 ± 0.14	0.59 ± 0.12	0.69	0.54 ± 0.10	0.56 ± 0.12
SBK	0.74	0.70 ± 0.05	0.69	0.57 ± 0.09	0.57 ± 0.09	0.62	0.50 ± 0.07	0.51 ± 0.07
UHNTG	0.54	0.45 ± 0.10	0.37	0.39 ± 0.18	0.39 ± 0.18	0.81	0.53 ± 0.00	0.53 ± 0.00
SMH	0.74	0.65 ± 0.05	0.64	0.59 ± 0.08	0.59 ± 0.08	0.59	0.55 ± 0.06	0.55 ± 0.06
UHNTW	0.45	0.39 ± 0.07	0.63	0.42 ± 0.08	0.42 ± 0.08	0.49	0.56 ± 0.09	0.57 ± 0.09
THPC	0.53	0.47 ± 0.05	0.52	0.51 ± 0.08	0.51 ± 0.08	0.55	0.53 ± 0.08	0.51 ± 0.08

Table 5: AUROC scores for the GRU-D and GRU-D with feature restriction across 6 sites for delirium prediction. Corruption is applied to features through linearly scaling the feature. * indicates manual intervention.

would have still been the best choice 48% of the time, the GRU-D 34% of the time, and the GRU-D with restriction 19% of the time.

4 Conclusion

The modification of the GRU-D with the feature restriction mask allows for sub-unit specific off switches. Across 3 different types of perturbations, the GRU-D with feature restriction experiences less (though not significantly less) harm in performance. The sub-unit specific off switches confer innate robustness to distribution changes which do not require conscious intervention to utilise. The robustness of the intervention exceeds that of a GRU-D and LSTM baseline models. Despite this, the LSTM architecture was still the preferred architecture in almost half of the cases. Future work may investigate if sub-unit specific isolation leads to faster fine-tuning or fewer required samples to refit the new sub-unit.

References

[1] Adarsh Subbaswamy, Peter Schulam, and Suchi Saria. Preventing failures due to dataset shift: Learning predictive models that transport. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 3118–3127. PMLR, 16–18 Apr 2019. URL `http://proceedings.mlr.press/v89/subbaswamy19a.html`.

[2] Sebastian Lee, Sebastian Goldt, and Andrew Saxe. Continual learning in the teacher-student setup: Impact of task similarity. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6109–6119. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/lee21e.html`.

[3] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4334–4343. PMLR, 10–15 Jul 2018. URL `http://proceedings.mlr.press/v80/ren18a.html`.

[4] Zachary Nado, Shreyas Padhy, D. Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift, 2020. URL `https://arxiv.org/abs/2006.10963`.

[5] Kun Zhang, Yuan Xue, Gerardo Flores, Alvin Rajkomar, Claire Cui, and Andrew M. Dai. Modelling ehr timeseries by restricting feature interaction, 2019. URL `https://arxiv.org/abs/1911.06410`.

[6] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8(1):6085, 2018.

[7] Bret Nestor, Matthew B. A. McDermott, Willie Boag, Gabriela Berner, Tristan Naumann, Michael C. Hughes, Anna Goldenberg, and Marzyeh Ghassemi. Feature robustness in non-stationary health records: Caveats to deployable model performance in common clinical machine learning tasks. In Finale Doshi-Velez, Jim Fackler, Ken Jung, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens, editors, *Proceedings of the 4th Machine Learning for Healthcare Conference*, volume 106 of *Proceedings of Machine Learning Research*, pages 381–405. PMLR, 09–10 Aug 2019. URL `https://proceedings.mlr.press/v106/nestor19a.html`.

[8] Amol A Verma, Sachin V Pasricha, Hae Young Jung, Vladyslav Kushnir, Denise Y F Mak, Radha Koppula, Yishan Guo, Janice L Kwan, Lauren Lapointe-Shaw, Shail Rawal, Terence Tang, Adina Weinerman, and Fahad Razak. Assessing the quality of clinical and administrative data extracted from hospitals: the General Medicine Inpatient Initiative (GEMINI) experience. *Journal of the American Medical Informatics Association*, 28(3):578–587, 11 2020. ISSN 1527-974X. doi: 10.1093/jamia/ocaa225. URL `https://doi.org/10.1093/jamia/ocaa225`.

[9] Shirly Wang, Matthew B. A. McDermott, Geeticka Chauhan, Marzyeh Ghassemi, Michael C. Hughes, and Tristan Naumann. Mimic-extract: A data extraction, preprocessing, and representation pipeline for mimic-iii. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, CHIL '20, page 222–235, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450370462. doi: 10.1145/3368555.3384469. URL `https://doi.org/10.1145/3368555.3384469`.