RETHINKING POSITIONAL EMBEDDING: A CASE STUDY IN TEMPORAL EVENT SEQUENCE MODELLING

Anonymous authors
Paper under double-blind review

ABSTRACT

We present an empirical comparison of the transformer's sinusoidal positional encoding and time-decaying encoding a time series application of the education domain involving 14,043 question-solving interactions from 1,260 students. We argue that Including time-based attention can be beneficial for time-series application domains where model prediction is crucial.

1 INTRODUCTION

Transformer architectures need positional embedding. In this paper, we ask: do we always need positional embedding? We present an empirical analysis of temporal event sequence modeling to encode timestamps as an alternative to positional embedding. We argue that temporal information can be more valuable to capture sequential events in the health and education sector.

2 BACKGROUND AND RELATED WORK

Researchers have used time-decaying attention Ghosh et al. (2020), inter-event embedding Gu (2021) instead of positional encoding. We present our study in the context of deep knowledge tracing (DKT) Piech et al. (2015); Pandey & Karypis (2019); Pandey & Srivastava (2020); Ghosh et al. (2020). Given a student's historical question performance, the goal is to predict the student's performance on the next question. In the education domain, the *forgetting curve theory* states that the students' memory decay exponentially with time Ebbinghaus (2013), Thus, researchers incorporated forgetting behavior: elapsed time between the consecutive events Shin et al. (2021); Nagatani et al. (2019), and elapsed time between events with similar skill tagging Nagatani et al. (2019).

3 METHODOLOGY

We present our model architecture below.

Text Embedding. We use the universal sentence encoder USE Cer et al. (2018) to encode action text into a *d = 512*vector, $\mathbf{E} \in \mathbb{R}^d$
Action Type Embedding. We use an embedding size of *d = 512* to encode four action types within the system.
Score Encoding. We extend the score of each question to a vector size *d*.
Response Encoding. We compute the cosine similarity between questions and responses and extend the value to a vector size *d*.

We concatenate these four feature vectors to form an input for each timestamp i, $\mathbf{x_i} \in \mathbb{R}^{4d}$.

Time Relation. This component takes into account students' forgetting behavior. We took the exponential of the elapsed time Δ_i between the T_{th} action timestamp, t_T and i_{th} previous action's timestamp, t_i as: $\mathbf{E_i} = [exp(\Delta_1), exp(\Delta_2), ..., exp(\Delta_{T-1})]$

Question Relation. This component computes the cosine similarities between the T_{th} question embedding, $\mathbf{E_T}$ and a previous action (question attempt) text embedding at t_i, $\mathbf{E_i}$.

Attention: We computed the dot product attention $\alpha_{i,j}$ Vaswani et al. (2017) by taking $\mathbf{q_i}$ is the query vector of a student's question attempted at time i. We combine $\alpha_{i,j}$ with the **Time** and

Table 1: Classification: Mean of 5-fold CV on test dataset and Model Parameter Count

Method	AUC	Accuracy	Model Parameter	Memory
Time-relation	0.81	0.76	17516564	16.705MB
Sinusoidal	0.86	0.79	20664340	19.707MB

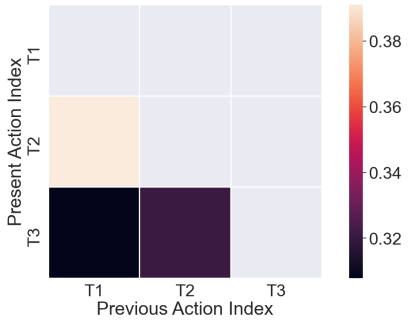

Figure 1: Attention Visualization on Test Fold. Weight is Averaged on Four Heads.
.

Question relation, $\mathbf{R}$ $\beta_{i,j} = \lambda\alpha_{i,j} + (1-\lambda)\mathbf{R_j}$, λ a trainable parameter. The final attention is $\mathbf{Att_i} = \sum_{j<i} \beta_{i,j}.\hat{\mathbf{x_j}}\mathbf{W^V}$, $\mathbf{v_j}$ = value vectors (See Appendix).

4 Experiment Design and Result

We present two sets of experiments: one with sinusoidal position embedding Vaswani et al. (2017) and one with forgetting behavior with exponential time decay. We report five-fold cross-validation results. The task is a binary classification task. Hyperparameter details are in the Appendix Section.

Dataset Statistics. Our dataset is from a K12 science reading platform, Actively Learn. The dataset statistics are: the number of assignments = 378, number of questions = 1,260, number of students = 1,680, and number of interactions = 14,043. The mean, standard deviation, and median of the sequence are 6.4, 3.6, and 6.

Result. The sinusoidal embedding performs better whereas our model's parameter size is relatively smaller. Our goal is not to achieve the best performance. Rather, we argue for an alternative of positional encoding specific to domain application. In this context, the time-decaying attention captures (i) forgetting behavior and (ii) an explanation of the model's prediction. In Figure 1, the student attempted two questions at T1 and T2. We observe the model puts more weight on T2 as a time-decaying result. The question at T1: "In your own words, explain how the arches were formed. Use some of the science terms we used in the past", at T2: Explain the meaning of physical change. What does it affect? The question at T3 is a resubmission of T2. The student scored zero on both questions at T1 and T2. The model predicted zero at T3.

5 Discussion and Conclusions

Incorporating time and model interpretation sequential prediction is important in the health sector, (e.g., disease progression modeling (Zhang, 2019)) and education domain (See Section 2). This paper presents an empirical analysis of alternative encoding.

URM Statement

"The author acknowledges that the sole author of this work meets the URM criteria of ICLR 2023 Tiny Papers Track. The author meets the Gender and First-time submitters at ICLR criteria."

REFERENCES

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 169–174, 2018.

Hermann Ebbinghaus. Memory: A contribution to experimental psychology. *Annals of neurosciences*, 20(4):155, 2013.

Aritra Ghosh, Neil Heffernan, and Andrew S Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2330–2339, 2020.

Yulong Gu. Attentive neural point processes for event forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7592–7600, 2021.

Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. Augmenting knowledge tracing by considering forgetting behavior. In *The world wide web conference*, pp. 3101–3107, 2019.

Shalini Pandey and George Karypis. A self-attentive model for knowledge tracing. *International Educational Data Mining Society*, 2019.

Shalini Pandey and Jaideep Srivastava. Rkt: Relation-aware self-attention for knowledge tracing. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 1205–1214, 2020.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 1*, pp. 505–513, 2015.

Dongmin Shin, Yugeun Shim, Hangyeol Yu, Seewoo Lee, Byungsoo Kim, and Youngduck Choi. *SAINT+: Integrating Temporal Features for EdNet Correctness Prediction*, pp. 490–496. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450389358. URL https://doi.org/10.1145/3448139.3448188.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Yuan Zhang. Attain: Attention-based time-aware lstm networks for disease progression modeling. In *In Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI-2019), pp. 4369-4375, Macao, China.*, 2019.

A APPENDIX

Hyperparameter: We used a batch size of 200, embedding size of 512, 300 epochs, learning rate = $1e^{-3}$, number of attention heads = 4, and early stopping if the validation AUC did not increase over five epochs.

Attention:

$$\begin{aligned} \alpha_{i,j} &= \text{Softmax}(\frac{\mathbf{q_i}\mathbf{W^Q}.(\mathbf{k_j}\mathbf{W^K})^\mathbf{T}}{d}), \text{all} j < i \\ \mathbf{Att_i} &= \sum_{j<i} \alpha_{i,j}.\mathbf{v_j}\mathbf{W^V} \end{aligned} \tag{1}$$

$\mathbf{k_j}$ and $\mathbf{v_j}$ denote respectively key and value vectors for previous actions, $j < i$.