Towards Interpretability of Attention-Based Knowledge Tracing Models

Thales B. S. F. Rodrigues[1], **Jairo F. de Souza**[1,2], **Heder S. Bernardino**[2], **Ryan S. Baker**[3]

[1]LApIC Research Group, Federal University of Juiz de Fora, Brazil

[2]Graduate Program in Computer Science, Computer Science Department,
Federal University of Juiz de Fora, Brazil

[3]Graduate School of Education, University of Pennsylvania, United States

Abstract. *Knowledge Tracing (KT) models based on attention mechanisms have demonstrated in literature the capability to predict student performance more accurately than previous models in some datasets. However, they fail to directly infer student knowledge. In this paper, we apply a proposed extension already seen in KT literature in order to infer latent knowledge to these models. We apply the extension to four different attention-based KT models, to investigate whether these models can better infer the knowledge outside the learning system than previous models. We find that attention-based models can generate better knowledge estimate correlations with student's scores than the previous models.*

1. Introduction

During the COVID-19 pandemic, online education proved to be important to move the gears of education around the world [Cao et al. 2021, Penteado and Fornazin 2021]. Although this type of educational approach have presented difficulties and pedagogical problems [Santos et al. 2020], the number of Online Intelligent Tutoring System (ITS) users increased during the period of social distancing due to the pandemic [Pantelimon et al. 2021, Akyuz 2020]. Such an increase in the amount of users led to an increase in the amount of data generated by these students, thus boosting existing research on Artificial Intelligence (AI) and Big Data applied to education [Liu et al. 2021].

Within the scope of AI applied to education, the Knowledge Tracing problem (KT) [Liu et al. 2021] consists of using student interaction data on the ITS to infer the student's knowledge state and their mastery in the concepts related to the exercises proposed by the ITS [Pandey et al. 2021]. Initially, this problem was addressed using probabilistic models such as the Bayesian Knowledge Tracing (BKT) [Corbett and Anderson 1995] and logistic regression models such as the Performance Factors Analysis (PFA) [Pavlik Jr et al. 2009] , known as the classic KT models. These models had the virtue of providing interpretable inferences of student skill, but the recent emergence of KT models based on deep learning has led to substantially better ability to predict future student performance in large data sets [Gervet et al. 2020, Liu et al. 2021]. These deep learning-based approaches are primarily based on Recurrent Neural Networks (RNNs), and the first well-known model, the Deep Knowledge Tracing (DKT) [Piech et al. 2015], has demonstrated that it is capable of capturing relationships of the student's knowledge in the datasets that are often implicit. However, the DKT model demonstrated certain irregularities in its predictions [Yeung and Yeung 2018], which was

DOI: 10.5753/sbie.2022.224685

partially addressed in the model Dynamic Key-Value Memory Networks (DKVMN) [Zhang et al. 2017] for KT. In order to address the limitation of the previous deep learning based-model, the DKVMN model consists of a Memory-Augmented Neural Network (MANN) with introduction of the skill-item matrix in its architecture, which can store knowledge concepts and update its student's mastery during the training phase. More recently, evidence has emerged that attention-based KT models are more able to better generalize their predictions over large amounts of data than models based on RNNs [Pandey and Karypis 2019] and to learn long-range dependencies within the data [Kenton and Toutanova 2019], due to the impressive contribution of the attention mechanism on the generalization over sequential data [Vaswani et al. 2017].

On the other hand, although deep learning-based models are the state of the art for the KT problem, the interpretability of these models is still a research problem [Scruggs et al. 2020, Mao 2018, Ding and Larson 2021, Mandalapu et al. 2021, Wang et al. 2020]. Although these models have a high predictive capacity in large datasets, their parameters and skill estimates are not interpretable in contrast to the classic models [Ding and Larson 2021]. This makes these models less useful for key goals of adaptive learning systems such as informing teachers about student knowledge [Liu et al. 2021]. Thus, this present work sought to explore the predictive capacity of attention-based KT models with more interpretable forms of inference of the student's knowledge state, such as the inference of the student's performance in an external data (post-test) in relation to the training dataset [Corbett and Bhatnagar 1997, Scruggs et al. 2020].

In this paper, we investigate whether attention-based KT models can provide more interpretable results over the deep learning-based models previously evaluated in the literature. We aim to use the extension proposed by Scruggs *et al.* [Scruggs et al. 2020] to evaluate whether new attention-based KT models can better predict the knowledge carried out of the learning system than previous methods in terms of interpretable skills. In order to validate the comparison between the extension applied to the new algorithms addressed in this work with the algorithms seen in the previous study in literature, we reproduce the extension in the DKT+ [Yeung and Yeung 2018] and DKVMN [Zhang et al. 2017] models, as baseline for our study.

2. Problem Definition

Knowledge Tracing (KT) is the problem of modeling the state of knowledge and mastery for different students in a online learning system, based on its interaction sequence within the system [Liu et al. 2021]. The problem aims to take the learning sequence of a student s represented by a sequence of interactions, generally, as $X_s =< x_0, x_1, ..., x_N >$, where N is the maximum sequence length of the student interaction, and the tuple $x_t = (e_t, a_t, r_t)$ represents the student learning interaction at time step t, where e_t represents the exercise solved by the student, a_t represents if the student answered correctly or incorrectly the exercise and r_t represents complementary information over the student interaction.

Within the learning sequence, each exercise e_t is related to unique or multiple knowledge concepts (KCs) or skills, that vary from dataset to dataset. In summary, the knowledge tracing research problem aims, given a historical dataset of a student's interaction in an online learning platform, to predict the student's performance in future exercises within the learning platform.

Interpretable forms of knowledge retrieved from knowledge tracing models has a concrete domain of applicability in the educational environment [Liu et al. 2021]. For example, in the BKT model, the knowledge estimates which are updated in the algorithm process for each student in the data can be used directly to estimate the strength and weakness of the students during the learning process [Lu et al. 2020]. BKT have achieved better results than PFA in some studies [Raposo et al. 2020]. On the other hand, although deep learning-based models achieve better predictive performance than classic models, the results generated by them are poorly interpretable [Liu et al. 2021, Ghosh et al. 2020], as they do not offer a direct inference of skill knowledge, instead making complex predictions for individual items.

In order to approximate the poorly understood prediction mechanism of deep learning-based models to the interpretable parameters already seen in the classic models, an alternative for addressing this limitation in deep learning-based KT models was presented in an extension applied over the output of the KT model, proposed by Scruggs *et al.* [Scruggs et al. 2020]. The proposed extension aims to reconnect deep learning-based KT models with interpretable forms of knowledge modeling outside the learning system which classic KT models are already explored. In this extension, once a deep learning-based KT model is trained, its performance predictions for specific items are averaged within each skill into an overall knowledge estimate, using a human-derived skill-item mapping.

Scruggs *et al.* [Scruggs et al. 2020] compares some KT models in terms of their ability to predict external post-test performance to understand how probabilistic-based models and neural-based models perform. However, the authors consider only two deep learning-based models, the DKT+ and DKVMN model. We extend that work using the same data set and overall approach to compare to recent attention-based KT models as well. We restrict our analysis to attention-based models since these are the more recent KT models in the literature and with promising results.

3. Deep Learning-based Models for KT

In this section, we provide an overview of the deep learning-based models used in the study, as well as differences between then.

3.1. *Deep Knowledge Tracing* +

Deep Knowledge Tracing (DKT) is the first RNNs-based KT model [Piech et al. 2015]. The DKT model is implemented in literature using Long Short-Term Memory Networks (LSTM), a variant of the standard RNN which has a mechanism in its architecture that provides to the model taking into account the forgetfulness in the student sequence and make a continuous representation of the knowledge state during the train process [Liu et al. 2021]. In line with Scruggs *et al.* [Scruggs et al. 2020], we use a variant of the DKT originally from Yeung & Yeung [Yeung and Yeung 2018], called DKT+. This model, in contrast to the standard DKT model, contains a regularization method that addresses problems seen in the original DKT, where performance estimates vary substantially from problem-to-problem, and sometimes the direction of change in estimates does not match student correctness.

3.2. *Dynamic Key-Value Memory Networks for Knowledge Tracing*

Dynamic Key-Value Memory Networks (DKVMN) is based on Memory-Augmented Neural Networks (MANN), a special type of RNN [Zhang et al. 2017]. The main differences between the MANN used in DKVMN model and the LSTM used in DKT+ model are divided in three different aspects. First, the state transitions in MANN focus in local transitions by read and write operations, different from the global transitions that occurs in RNN architecture [Graves et al. 2014]. Second, MANN uses an external memory matrix that increases the storage of memory in the model in relation to the standard LSTM architecture [Sukhbaatar et al. 2015]. Third, in contrast to traditional RNNs, the number of parameters of the MANN should not be tied to the size of the hidden state [Santoro et al. 2016], which implies more memory slots in the model, without increasing the total number of parameters.

Moreover, the DKVMN model can accurately determine a specific student's knowledge state on KCs in the dataset, with the introduction of a static *key* matrix that store latent KCs and a dynamic *value* matrix that update the mastery of corresponding KCs through the read and write process [Liu et al. 2021].

3.3. *Self-Attentive Knowledge Tracing*

Self-Attentive Knowledge Tracing (SAKT) is the first KT model that uses a purely attention mechanism (transformer) method in its architecture [Pandey and Karypis 2019]. The main idea behind this model is to take into account the relevance between exercises and KCs that are related to each other. SAKT identifies these relevant exercises from the past interactions and predicts the student's likelihood to correctly answer future exercises based on those relevant exercises. The mechanism behind this property is the attention weights, which are used to determine the relevance of each of the previous interactions.

The main difference between SAKT and previous RNNs-based KT models, which leads to better performance results in the attention-based model, is due to the attention mechanism which can solve the problem of vibrations in prediction outputs, discussed in Zhu *et al.* [Zhu et al. 2020]. The attention mechanism can solve the problem by capturing the relationships between KCs in the input sequence regardless of the length of the sequence. Moreover, unlike RNNs-based models, SAKT is suitable for parallelism, which makes the model faster than DKT model [Liu et al. 2021].

3.4. *Deep Self-Attentive Knowledge Tracing*

Deep Self-Attentive Knowledge Tracing (DSAKT) [Zeng et al. 2021] is an improvement of the SAKT model, based in a encoder-decoder transformer model. The main difference between both is that DSAKT uses the Multi-Head Attention (MHA) [Vaswani et al. 2017] layer twice, sharing the same weight in decoder. This proposed mechanism in the KT model works as a reinforcement to the capability of the model to retain the relations between the KCs in the dataset, until making the final prediction. Like its ancestor, DSAKT is suitable for parallelism, which makes the model faster than RNNs-based models.

3.5. *Attentive Knowledge Tracing*

In contrast to the previous attention-based KT models, Attentive Knowledge Tracing (AKT) model puts raw embeddings into context and takes account into the students' entire interactions in a representation of past questions and responses, which is named context-aware representation [Ghosh et al. 2020].

Another different aspect between AKT and the previous attention-based KT models is the novel monotonic attention mechanism. Motivated by evidence around memory decay in the student learning process, AKT incorporates a multiplicative exponential decay in the attention scores calculation to down weight the relevance of distant questions in the student interaction sequence.

3.6. *Adversarial Training based Knowledge Tracing*

Despite purely attention-based models being the state-of-the-art in KT task over large datasets, they suffer when dealing with a small amount of data. In order to deal with these problems and enhance the prediction for small datasets, the first KT model based on adversarial training (ATKT) was proposed by Guo *et al.* [Guo et al. 2021]. Adversarial training in deep learning literature is an efficient regularization technique for promoting model robustness [Pang et al. 2021], and in ATKT architecture, led to better prediction results than SAKT and previous deep learning-based models in small datasets.

The model architecture consists of an attentive-LSTM, which like SAKT and DSAKT aggregates information from previous exercises and takes into account relevant KCs to make the predictions. The key difference between ATKT and previous attention-based models is the adversarial examples, which are generated by perturbations in the original input sequence, in order to aggregate robustness in the generalization of the model. Another different aspect is the proposed knowledge hidden state (KHS) attention module, which gradually aggregates information from the previous KHS while highlighting the relevance of the current KHS, to try to make the output prediction more accurate.

4. Proposed Method

Motivated by research that explores the interpretability of deep learning-based KT models, the present work aims to take an alternative extension applied in KT literature over deep learning-based models, shown to lead to better estimates of interpretable knowledge carried out of learning systems both for deep learning-based models and classic models [Scruggs et al. 2020], and investigate the performance of this extension for attention-based KT models and other contemporary KT models which were not evaluated in the previous study.

Figure 1 summarizes the evaluation method proposed in our study. First, we train each algorithm described above using student data. Second, we apply the extension over the output of the algorithms in order to evaluate the performance of the students in an external test, as seen in [Scruggs et al. 2020]. Finally, we calculate the Pearson correlation between the output of the applied extension and student post-test scores for each algorithm, shown in Table 2, in order to estimate how successful these algorithms are at inferring the students' knowledge, measured in data outside the original learning system the models were trained in.

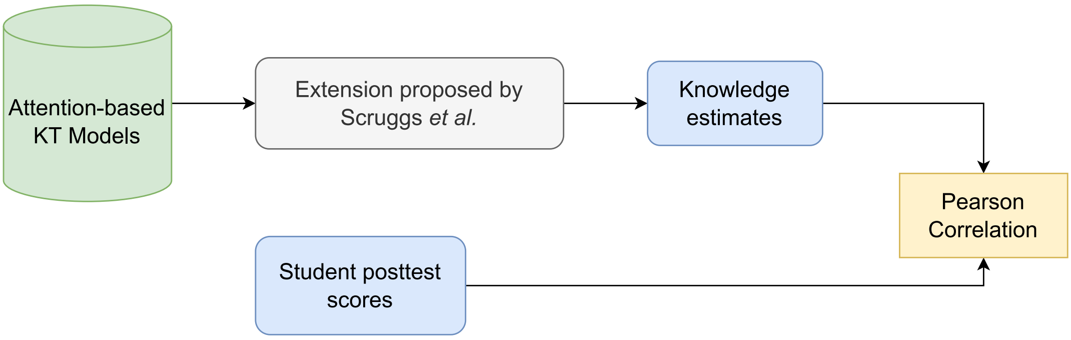
Attention-based KT Models
Extension proposed by Scruggs *et al.*
Knowledge estimates
Pearson Correlation
Student posttest scores

Figure 1. Scheme of the proposed method

4.1. Setup

The training and model code was adapted from public repositories on GitHub in Python language. ATKT[1], AKT[2], SAKT[3], and DSAKT[3] were implemented using PyTorch library [Paszke et al. 2019]. For the baseline models, DKT+[4] model was implemented using TensorFlow [Abadi et al. 2015] and DKVMN[5] model was implemented using MXNet library [Chen et al. 2015]. We trained each model for 300 epochs using a GPU Tesla K80, with the learning rate of 1×10^{-3} and Adam optimizer (but SGD for DKVMN). The other parameters of the models were kept according to the values suggested by the repository owner. Table 1 shows the model hyperparameters values.

Table 1. Hyperparameters for the trained models.

	Batch size	Sequence length	Dropout	Attention heads	Skill embedding dimension
SAKT	128	350	0.2	8	–
DSAKT	128	350	0.7	8	–
AKT	24	300	0.05	8	50
DKT+	32	300	–	–	–
ATKT	24	300	–	–	256
DKVMN	10	350	–	–	50

4.2. Dataset details

The dataset and the post-test data used in this study is available on-line[6] and has been used in previous studies, coming from a series of studies about the effectiveness of erroneous examples on student learning [Richey et al. 2019]. The dataset contains a total of 598 students with 70,552 student attempts of questions about basic math. The students were evaluated in a 43-item post-test divided into four different KCs: *ordering decimals* (22 items),

[1] https://github.com/xiaopengguo/ATKT
[2] https://github.com/arghosh/AKT
[3] https://github.com/Fusion4233919/DSAKT
[4] https://github.com/ckyeungac/deep-knowledge-tracing-plus
[5] https://github.com/jennyzhang0215/DKVMN
[6] https://pslcdatashop.web.cmu.edu/Project?id=67

placement on number line (6 items), *completing the sequence* (4 items), and *decimal addition* (11 items). The item distribution per skill in the post-test is related to the number of common skill misconceptions in the dataset [Richey et al. 2019, Scruggs et al. 2020].

We have processed the data keeping only the first attempt of the student on each item in the interaction sequence, and have trained each algorithm with the same amount of data for training and validation in order to generate knowledge estimates that are then tested on an external test, using the extension described below.

4.3. Extension and Knowledge Estimates

The extension proposed by Scruggs *et al.* [Scruggs et al. 2020] consists of taking the probability of correctness, which is generated by each algorithm after the training phase, over all exercises that a student answered from each KC (were the item to be seen again), and then calculating the mean of those values for each student, within each KC. This final mean is used as knowledge estimates and compared with the students' scores on the post-test questions.

After calculating the knowledge estimates by using the algorithms and obtaining the students' post-test scores, we compute the Pearson correlation between both sets of values, in order to evaluate if these knowledge estimates generated by attention-based models are successful at inferring student knowledge carried outside of the learning system.

5. Results

Table 2 shows the correlation between each knowledge estimate from algorithms and the post-test score for each KC in the training data.

Table 2. Pearson correlation between knowledge estimates and post-test scores for each algorithm.

	Ordering Decimals	Placement on Number Line	Complete the Sequence	Decimal Addition	Sum of Ranks
ATKT	0.72 (#1)	0.63 (#4)	0.37 (#1)	0.55 (#1)	7 (#1)
DSAKT	0.72 (#1)	0.64 (#2)	0.36 (#3)	0.52 (#4)	10 (#2)
AKT	0.71 (#4)	0.67 (#1)	0.36 (#3)	0.53 (#3)	11 (#3)
DKVMN	0.72 (#1)	0.62 (#6)	0.35 (#5)	0.55 (#1)	13 (#4)
SAKT	0.71 (#4)	0.63 (#4)	0.37 (#1)	0.51 (#5)	14 (#5)
DKT+	0.71 (#4)	0.64 (#2)	0.34 (#6)	0.48 (#6)	18 (#6)

For the KC Ordering Decimals, ATKT (r=0.72), DSAKT (r=0.72) and DKVMN (r=0.72) produced the best knowledge estimates in relation to the other models. AKT, SAKT and DKT+ (r=0.71) achieved the same correlation to the post-test scores in this skill. For the skill Placement on Number Line, AKT (r =0.67) produced the best estimates. The others models produced estimates that were substantially worse than AKT, DSAKT (r=0.64), DKT+ (r=0.64), ATKT (r=0.63) and SAKT (r=0.63). The worst result overall for this KC was achieved by DKVMN (r=0.62).

For Complete the Sequence, all models achieved substantially worse results than the other two KCs. ATKT (r=0.37) and SAKT (r=0.37) produced the best estimates, although they did not differ substantially from the other attention-based models AKT (r=0.36) and DSAKT (r=0.36). The worst estimates for this KC were produced by the baseline models, DKVMN (r=0.35) and DKT+ (r=0.34).

For Decimal Addition, the estimates produced by ATKT (r=0.55) and DKVMN (r=0.55) model substantially outperformed the other studied models. AKT (r=0.53), DSAKT (r=0.52) and SAKT (r=0.51) had results that were close to each other. Again, DKT+ (r=0.48) produced substantially worse estimates than the other models for this KC.

Although the models have achieved very close results, we ranked them within each KC and the sum of rankings was calculated (Table 2). Using this technique makes it easier to compare the final model performance. As shown, the ATKT model outperformed the other attention-based models and the baseline models, followed by AKT and DSAKT.

Table 3. Training and prediction time in seconds for each model.

	Training time (s)	Prediction time (s)
SAKT	347	6.94
DSAKT	763	7.34
AKT	1,586	16.22
DKT+	1,943	25.56
ATKT	2,595	2.46
DKVMN	3,490	9.68

Training deep learning-based models is time-consuming and training time can be an important aspect for adoption of the technology on real-word applications or large-scale datasets. Therefore, Table 3 shows a comparison of the training time for each model. In general, attention-based KT models were faster than the baseline models in the training phase. Particularly, SAKT and DSAKT achieved training times better than those reached by the other models. On the other hand, although ATKT produces the best results, it is slower to train than the other attention-based models and DKT+. The DKVMN model had the worst training time between the models used in this study.

Moreover, Table 3 shows the running time required by each model to predict the results for all students. Although ATKT achieved the worse training time when compared to the other attention-based models, its prediction time is the best one. Also, DKVMN has the slowest training time, but its prediction time was similar to those obtained by SAKT and DSAKT, and better than those achieved by AKT and DKT+. In addition, DKT+ obtained the worst prediction time among the models evaluated in this study.

6. Concluding Remarks and Future Works

This work extended the previous work in KT literature presented by Scruggs *et al.* [Scruggs et al. 2020], using four different attention-based knowledge tracing models that were not evaluated previously. Using these new models and the extension proposed in literature, we were able to convert the high predictive performance of attention-based knowledge tracing models into knowledge estimates, which demonstrate certain correlation with the student's score in an external test in relation to the training data. According to the results of this study, despite the similar estimates values, almost all attention-based models achieved better correlation values between knowledge estimates and the post-test scores than the baseline models by the sum of rankings. ATKT model demonstrates substantially better final result over all studied models. This indicates that the adversarial training in attention-based KT models can (i) reach results in learning systems and (ii) infer interpretable student's knowledge using the proposed extension better than the previous deep learning-based models.

Regarding the processing time, attention-based KT models were faster than the baseline models in training, even in small datasets as we used here, but the prediction time did not present the same behavior. Then, their use in real-world applications should consider scalability in relation to retraining needs, model response time, and dataset size (number of student interactions and exercises).

This paper does not present definitive results, and new research is needed to establish our findings as conclusive. Our findings should be replicated in different domains and with other student populations, considering size (number of students) and complexity (number of interactive items). Deep learning-based KT approaches are usually evaluated according to their prediction performance within the system. However, our results show how these approaches behave when used to infer knowledge. Other evaluation metrics and statistical analyzes are also needed to better understand the usefulness of these models for inferring knowledge, including also a larger collection of KT models.

Acknowledgements

The authors thank the financial support provided by CNPq, Capes, FAPEMIG, UFJF, and NSF #DRL-1661121. We thank Bruce McLaren and Richard Scruggs for making the data set available and assistance with replication.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. In *arXiv:1603.04467*. Software available from tensorflow.org.

Akyuz, Y. (2020). Effects of intelligent tutoring systems (its) on personalized learning (pl). *Creative Education*, 11:953–978.

Cao, J., Yang, T., Lai, I. K.-W., and Wu, J. (2021). Student acceptance of intelligent tutoring systems during covid-19: The effect of political influence. *The International Journal of Electrical Engineering & Education*, page 00207209211003270.

Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., and Zhang, Z. (2015). Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274*.

Corbett, A. T. and Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4:253–278.

Corbett, A. T. and Bhatnagar, A. (1997). Student modeling in the act programming tutor: Adjusting a procedural learning model with declarative knowledge. In *User modeling*, pages 243–254. Springer.

Ding, X. and Larson, E. C. (2021). On the interpretability of deep learning based models for knowledge tracing. In *arXiv:2101.11335*.

Gervet, T., Koedinger, K., Schneider, J., and Mitchell, T. (2020). When is deep learning the best approach to knowledge tracing? *Journal of Educational Data Mining*, 12(3):31–54.

Ghosh, A., Heffernan, N. T., and Lan, A. S. (2020). Context-aware attentive knowledge tracing. *CoRR*, abs/2007.12324.

Graves, A., Wayne, G., and Danihelka, I. (2014). Neural turing machines. In *arXiv:1410.5401*.

Guo, X., Huang, Z., Gao, J., Shang, M., Shu, M., and Sun, J. (2021). Enhancing knowledge tracing via adversarial training. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 367–375.

Kenton, J. D. M.-W. C. and Toutanova, L. K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Liu, Q., Shen, S., Huang, Z., Chen, E., and Zheng, Y. (2021). A survey of knowledge tracing. In *arXiv:2105.15106*.

Lu, Y., Wang, D., Meng, Q., and Chen, P. (2020). Towards interpretable deep learning models for knowledge tracing. In *International Conference on Artificial Intelligence in Education*, pages 185–190. Springer.

Mandalapu, V., Gong, J., and Chen, L. (2021). Do we need to go deep? knowledge tracing with big data. In *35th AAAI Conference on Artificial Intelligence*.

Mao, Y. (2018). Deep learning vs. bayesian knowledge tracing: Student models for interventions. *Journal of educational data mining*, 10(2).

Pandey, S. and Karypis, G. (2019). A self-attentive model for knowledge tracing. In *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019)*.

Pandey, S., Karypis, G., and Srivastava, J. (2021). An empirical comparison of deep learning models for knowledge tracing on large-scale dataset. In *35th AAAI Conference on Artificial Intelligence*.

Pang, T., Yang, X., Dong, Y., Su, H., and Zhu, J. (2021). Bag of tricks for adversarial training. In *International Conference on Learning Representations*.

Pantelimon, F.-V., Bologa, R., Toma, A., and Posedaru, B.-S. (2021). The evolution of ai-driven educational systems during the covid-19 pandemic. *Sustainability*, 13(23):13501.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Pavlik Jr, P., Cen, H., and Koedinger, K. (2009). Performance factors analysis - a new alternative to knowledge tracing. *Frontiers in Artificial Intelligence and Applications*, 200:531–538.

Penteado, B. and Fornazin, M. (2021). Detecção de inovações tecnológicas na evolução da informática educacional no brasil. In *Anais do XXXII Simpósio Brasileiro de Informática na Educação*, pages 157–167, Porto Alegre, RS, Brasil. SBC.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., and Sohl-Dickstein, J. (2015). Deep knowledge tracing. *Advances in neural information processing systems*, 28.

Raposo, A., Maranhão, D., and Neto, C. S. (2020). Analise da capacidade preditiva de técnicas para modelagem do conhecimento aplicadas ao aprendizado de algoritmos. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 1653–1662, Porto Alegre, RS, Brasil. SBC.

Richey, J. E., Andres-Bray, J. M. L., Mogessie, M., Scruggs, R., Andres, J. M., Star, J. R., Baker, R. S., and McLaren, B. M. (2019). More confusion and frustration, better learning: The impact of erroneous examples. *Computers & Education*, 139:173–190.

Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., and Lillicrap, T. (2016). Meta-learning with memory-augmented neural networks. In Balcan, M. F. and Weinberger, K. Q., editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1842–1850, New York, New York, USA. PMLR.

Santos, C., Paillard, G., Moreira, L., Filho, F. R. S., and Coutinho, E. (2020). Uma análise qualitativa sobre atividades remotas em disciplinas no período de isolamento social. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 292–301, Porto Alegre, RS, Brasil. SBC.

Scruggs, R., Baker, R. S., and McLaren, B. M. (2020). Extending deep knowledge tracing: Inferring interpretable knowledge and predicting post-system performance. In *Proceedings of the 28th International Conference on Computers in Education*.

Sukhbaatar, S., Weston, J., Fergus, R., et al. (2015). End-to-end memory networks. *Advances in neural information processing systems*, 28.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, F., Liu, Q., Chen, E., Huang, Z., Chen, Y., Yin, Y., Huang, Z., and Wang, S. (2020). Neural cognitive diagnosis for intelligent education systems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6153–6161.

Yeung, C.-K. and Yeung, D.-Y. (2018). Addressing two problems in deep knowledge tracing via prediction-consistent regularization. In *Proceedings of the Fifth Annual ACM Conference on Learning at Scale*, pages 1–10.

Zeng, J., Zhang, Q., Xie, N., and Yang, B. (2021). Application of deep self-attention in knowledge tracing. In *Arxiv:2105.07909*.

Zhang, J., Shi, X., King, I., and Yeung, D.-Y. (2017). Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774.

Zhu, J., Yu, W., Zheng, Z., Huang, C., Tang, Y., and Fung, G. P. C. (2020). Learning from interpretable analysis: Attention-based knowledge tracing. *Artificial Intelligence in Education*, pages 364–368.