Predicting Learners Need
for Recommendation Using Dynamic
Graph-Based Knowledge Tracing

Abdessamad Chanaa[(✉)] and Nour-Eddine El Faddouli

RIME Team, MASI Laboratory, E3S Research Center,
Mohammadia School of Engineers (EMI), Mohammed V University (UM5),
Rabat, Morocco
`abdessamad.chanaa@gmail.com, nfaddouli@gmail.com`

Abstract. Personalized recommendation as a practical approach to overcoming information overloading has been widely used in e-learning. Based on learners individual knowledge level, we propose a new model that can predict learners needs for recommendation using dynamic graph-based knowledge tracing. By applying the Gated Recurrent Unit (GRU) and the Attention model, this approach designs a dynamic graph over different time steps. Through learning feature information and topology representation of nodes/learners, this model can predict with high accuracy of 80,63% learners with low knowledge acquisition and prepare them for further recommendation.

Keywords: Node classification · Dynamic graph · Knowledge tracing · Recommendation · Gated Recurrent Unit (GRU)

1 Introduction

The personalized recommendation has been widely used in e-learning systems; It has been a practical approach to overcome information overloading by helping learners for better course selection [3,8]. However, the development of recommendation system must not only consider the capability of delivering the suitable learning material to the learner anytime, but also how to actively distinguish learners who need a recommendation at that time based on their past performance.

Knowledge tracing, on the other hand, is the process of modelling student knowledge over time to predict how learners will perform on future interactions accurately [5]. Knowledge tracing can identify suitable learners for a potential recommendation based on their knowledge level, thus providing more effective learning. It can be helpful for both learners and tutors, as predicting recommendation need in the right time can highly decrease drop out rate and increase learners engagement.

Recently, deep learning [2] and graph theory [11] are becoming two actives areas in e-learning. Previous work tries to predict student proficiency by modelling knowledge concepts into nodes using a deep graph neural network [9]. Although the efficiency of this approach, it focuses on knowledge concepts more than the learner. Also, this approach is not entirely taking into consideration the dynamic structure of the graph, which reflects the knowledge acquisition change over time steps.

In our paper, Based on [12], we propose a time-series node classification in a dynamic graph-based knowledge tracing approach. By modelling learners into nodes, we group learners in graphs based on a particular knowledge concept introduced by the tutor. Both nodes and graph topology are transforming over time, matching the knowledge tracing of learners. Through Gated Recurrent Unit (GRU) network [4] and the Attention Neural Network (ANN) [7], we propose to learn feature representation by aggregating the learner (presented by node) and its neighbours, then extract the network topology information at each different time step. The generated dependent temporal information will provide adequate information about the actual need for a future recommendation in the chosen knowledge concept for every individual learner presented in the graph.

2   Proposed Approach

Problem Definition: The problem we consider in this paper is supervised node classification. We suppose that the coursework is structured as $G = (\zeta^1, \zeta^2, ..., \zeta^T)$ where T is the number of time steps. $\zeta^t = (V, A^t, X^t, C)$ is the graph at time step t, where ζ^t denote a graph with nodes set V. Let $N = |V|$ denote the number of learners/nodes in our graph. Those nodes share a knowledge concept C as a dependency relationship, where $C = \{C_1, C_2, ..., C_m\}$ presents a knowledge concept where m is the number of existing knowledge concepts. Let $A^t \in R^{N \times N}$ be the adjacency matrix describing nodes connections where $A_{ij} = 1$ shows a shared knowledge concept C at time t between nodes i and j. A missing connection is signified by $A_{ij} = 0$. $X^t \in R^{N \times f}$ is the node attribute matrix where f is the dimension of the attribute features (the number of features/information presenting each learner). Both A^t and X^t change at different time steps, while V and C are fixed for all time steps.

Dynamic Graph Based Knowledge Tracing: As shown in the Fig. 1, first, the tutor chose an available knowledge concept. The knowledge tracing dataset is transformed into a dynamic graph that changes over time steps, where each node represents a learner with attribute features extracted and aggregated from his previous knowledge. All learners in the generated graphs share the same knowledge concept already chosen by the tutor. The idea behind node classification in a dynamic graph is to integrate both network structure information and node attribute information, using two connected GRU [12], an attribute GRU (A-GRU) and a topology GRU (T-GRU). First, attention neural network capture relevant node information and then aggregate important neighbours of
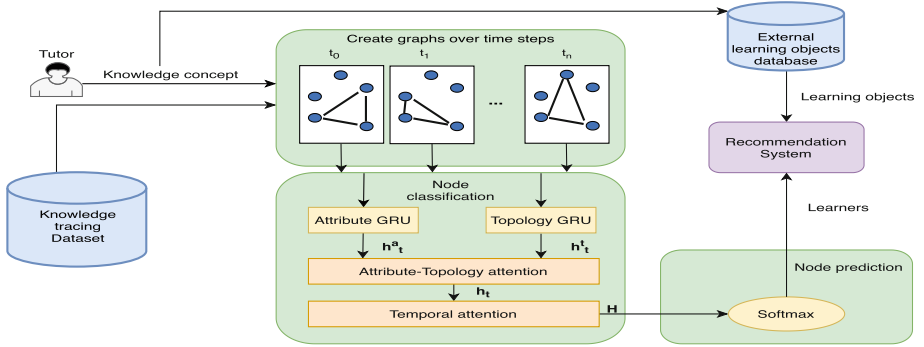
Fig. 1. The global architecture and workflow of the approach

a node. We use this neighbour representation along with node features vector of the previous state at each time step resulting in the new GRU state vector $h_t^A \in R^{d_h}$ that represents the A-GRU, where d_h is the state vector size. As for the T-GRU, it considers the topology context vectors of a node/learner at different time steps, resulting in the GRU state vector $h_t^T \in R^{d_h}$. Both T-GRU and A-GRU share the same calculation process of a standard GRU [1]. The attribute-topology attention determines the importance of attribute and topology at each time step; It receives the state vectors h_t^T and h_t^A and resolves respectively the attention values β_t^A and β_t^T. Therefore, the final state vector at time step t is: $h_t = [(\beta_t^T \times h_t^T)^\top \oplus (\beta_t^A \times h_t^A)^\top]^\top \in R^{2d_h}$. Moreover, temporal attention is added to detect the temporal influence in graph structure over multiple time step. The main objective of the temporal self-attentional layer is to capture the temporal variations in graph structure over multiple time steps. The attention model receives the state h_t and outputs the attention value α_t for each state. Using multiple-head self-attention [10], The final vector representation for the node is $\alpha \times H \in R^{2d_h}$, where $H = [h_1...h_t]$ represents the concatenation of all h_t and $\alpha \in R^T$ is the attention value of all different time steps. Finally, we used the cross-entropy loss and the Softmax function to estimate the node labels. Only the nodes that represent learners with low knowledge acquisition over time steps on the chosen knowledge concepts will be input to the recommendation system, alongside with learning objects matching that knowledge concept.

3 Experiment

3.1 Dataset

In order to evaluate our proposed approach, we adopt the dataset drawn from the ASSISTments learning platform[1] [6]. We reorganized the dataset by extracting

---

[1] https://sites.google.com/site/assistmentsdata/home/2012-13-school-data-with-affect.

and aggregating relevant features and then labelling it. We chose eight different features to represent the learner (time spent, number of correct answers, the hints count, the attempts count, frustration score, boredom score, confusion score and concentration score). Each learner is labelled with a binary value indicating whether the learner has low knowledge acquisition and needs a recommendation. The data was coded by two experts with a good inter-rater agreement. With the new labelled data, we took the example of «Addition and Subtraction Integers» as knowledge concept (the labelled data shows a 42% of learners that have problems and need a recommendation); Then we created a dynamic graph based on the chosen knowledge concept as explained in Table 1. This graph links all learners that pass an assignment with the knowledge concept «Addition and Subtraction Integers» over different time steps. The dataset alongside the generated graph is publicly available[2]. It is important to note that this experiment was conducted in Google Colab[3] with P100-PCIE-16 GB GPU and 25 GB RAM support settings.

Table 1. Reports on the graph data for the considered concept.

Knowledge concept	Assignments	Nodes	Features	Time steps	Labels
Addition and subtraction integers	151061	10732	8	10	2

3.2   Results and Discussion

The results are presented in Table 2. After several experiences, we notice that our model achieves the best performance under those parameters: batch size $= 2048$, learning rate $= 0.001$, number of epochs $= 30$, the state vector size $d_h = 12$. Our model combines the importance of chosen features that represent each learner of the graph, alongside with graph topology that represents the link between learners with the same knowledge concept. Using a dynamic representation of the graph over time steps, this approach will model better the learning acquisition of learners comparing to any static method that relies only on a static snapshot of the graph. The high accuracy also proves the effectiveness of the user attention model. In other words, this model can predict with high accuracy the need for a recommendation for each learner, which will highly decrease the dropout rate.

Table 2. Experiment results.

Knowledge concept	Accuracy	F1 score	AUC
Addition and subtraction integers	0,8063	0,8063	0,8342

Additionally, this approach will also facilitate building an adaptive system for learners with a low acquisition.

4 Conclusion and Future Work

In this work, we exploit the use of node classification in a dynamic graph-based knowledge tracing approach to predict the needs for a recommendation for learners, using mainly the GRU and the Attention models. The experimental results have demonstrated the efficiency of the proposed approach. Future works will focus on building a framework matching the chosen learners for recommendation with suitable learning objects.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate, pp. 1–15 (2014). http://arxiv.org/abs/1409.0473
2. Chanaa, A., El Faddouli, N.E.: Deep learning for a smart e-learning system. In: ACM International Conference Proceeding Series (2018). https://doi.org/10.1145/3289100.3289132
3. Chanaa, A., El Faddouli, N.E.: Context-aware factorization machine for recommendation in Massive Open Online Courses (MOOCs). In: 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems, WITS 2019, pp. 1–6 (2019). https://doi.org/10.1109/WITS.2019.8723670
4. Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: encoder-decoder approaches, pp. 103–111 (2015). https://doi.org/10.3115/v1/w14-4012
5. Corbett, A.T., Anderson, J.R.: Knowledge tracing: modeling the acquisition of procedural knowledge. User Model. User Adap. Interact. **4**(4), 253–278 (1994)
6. Feng, M., Heffernan, N., Koedinger, K.: Addressing the assessment challenge with an online system that tutors as it assesses. User Model. User Adap. Interact. **19**(3), 243–266 (2009). https://doi.org/10.1007/s11257-009-9063-7
7. Kim, Y., Denton, C., Hoang, L., Rush, A.M.: Structured attention networks, pp. 1–21 (2017). http://arxiv.org/abs/1702.00887
8. Klašnja-Milićević, A., Ivanović, M., Nanopoulos, A.: Recommender systems in e-learning environments: a survey of the state-of-the-art and possible extensions. Artif. Intell. Rev. **44**(4), 571–604 (2015). https://doi.org/10.1007/s10462-015-9440-z
9. Nakagawa, H., Iwasawa, Y., Matsuo, Y.: Graph-based knowledge tracing: modeling student proficiency using graph neural network. In: Proceedings of 2019 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2019, pp. 156–163 (2019). https://doi.org/10.1145/3350546.3352513
10. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
11. Vidal, J.C., Lama, M., Otero-García, E., Bugarín, A.: Graph-based semantic annotation for enriching educational content with linked data. Knowl. Based Syst. **55**, 29–42 (2014). https://doi.org/10.1016/j.knosys.2013.10.007
12. Xu, D., et al.: Adaptive neural network for node classification in dynamic networks. In: ICDM. IEEE (2019)