

# ПОДБОР УЧЕБНЫХ ПРОГРАММ В ОБРАЗОВАТЕЛЬНЫХ СИСТЕМАХ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ ГЛУБОКОГО ОБУЧЕНИЯ

Папроцкий Э.В.

Белорусский государственный университет информатики и радиоэлектроники, г. Минск, Беларусь, fisheroliver9@gmail.com

Abstract. The growth of the popularity of deep learning algorithms has been determined by the possibilities of using a special type of deep neural network - GNN (Graph Neural Network) in the field of forecasting educational programs in educational systems. In addition, based on the DKT (Deep Knowledge Tracing) method, an improved method of selecting training programs was presented. The results of the improved method indicate that this method of predicting the results of knowledge is superior to previous methods and has a positive impact on the quality of learning.

В настоящее время повышение качества образования – это не только задача государственного уровня. Это, в первую очередь, задача самих учреждений образования, которые по мере развития рынка образовательных услуг и обострения конкуренции вынуждены искать дополнительные конкурентные преимущества.

Существующие методы и подходы оценки качества образования не позволяют объективно оценивать знания каждого учащегося, а также не способны выявлять устойчивой зависимости между эффективностью обучения студентов и качеством организации деятельности учебных заведений.

Указанные недостатки обуславливают необходимость разработки новых методов обучения, которые позволят определять учебные программы для максимально эффективного обучения учащихся на основе их индивидуального уровня знаний и способности к обучению.

Применение нейронной сети GNN [1] в составлении учебной программы предполагает прогнозирование успеваемости учащихся в ходе выполнения программы обучения. Для данного прогнозирования необходимо представить систему обучения в виде графа  $G=(V, E, A)$ ; требование к усвоению учебной программы декомпозируется на  $N$  заданий, известных как  $V=(v_1, \dots, v_N)$ , и эти задания разделяют отношения зависимости, представленные как  $E \subseteq V \times V$ , где степень зависимости определяется матрицей смежности  $A \in R^{N \times N}$ . Каждый учащийся имеет текущее состояние прогресса обучения  $h^t = \{h_{i \in V}^t\}$ , по времени. В процессе обучения студент выполняет задания  $v_i$ , после чего обновляется состояние  $h_{i \in N_i}^t$ , где  $N_i$  – множество соседних заданий [2].

Использование возможностей нейронной сети GNN в модели отслеживания знаний позволяет улучшить производительность и интерпретируемость, в отличие от предыдущих методов, которые не учитывают особенности структур данных построенных на графе [3]. Цель данной работы заключается в следующем:

- продемонстрировать возможности нейронной сети GNN в решениях задач по подбору учебных программ;
- реализация модели предсказаний учебных программ на основе графа.
- провести тестирование разработанного метода подбора учебных программ и сравнить результаты работы с другими методами.

Предложенный метод заключается в использовании GNN в задаче по подбору учебных программ и представлении необходимых данных для модели в виде графа. Архитектура данного метода представлена на рисунке 1.

Архитектура предложенного метода состоит из трёх этапов, а именно:

1. Агрегирование (aggregation). Агрегирует скрытый слой состояний и объединяет  $t$ -тые ответы на задания с соседними заданиями :

$$h_k^t = \begin{cases} h_k^t, x^t E_x, k = i \\ h_k^t, E_c(k), k \neq i \end{cases} \quad (1)$$

где  $x^t \in \{0, 1\}^{2N}$  – входной вектор, который показывает правильность выполнения заданий за период времени  $t$ ;  $E_x \in \mathbb{R}^{2N \times e}$  – матрица индексов учебных заданий и ответов на них;  $E_c \in \mathbb{R}^{N \times e}$  – матрица индексов учебных заданий;  $E_c(k)$  – матрица, где  $k$ -ая строка  $E_c$  и  $e$  – это размерность.

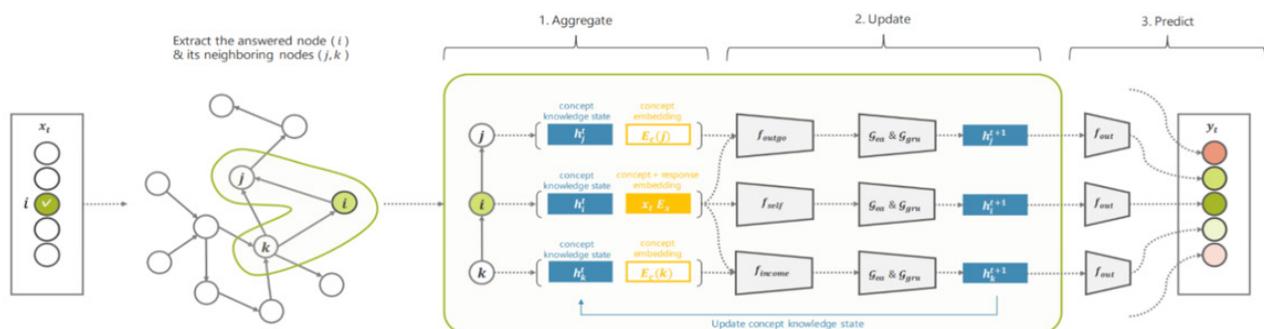


Рисунок 1 – Архитектура рассмотренного метода

2. Обновление (update). Обновление скрытого слоя состояний агрегированных признаков (features) и графа модели:

$$m_k^{t+1} = \begin{cases} f_{self}(h_k^t), k = i \\ f_{neighbor}(h_i^t, h_k^t), k \neq i \end{cases} \quad (2)$$

$$\tilde{m}_k^{t+1} = G_{ea}(m_k^{t+1})$$

$$h_k^{t+1} = G_{gru}(\tilde{m}_k^{t+1}, h_k^t)$$

где  $f_{self}$  – многослойный перцептрон (MLP [3]);  $G_{ea}$  – блок удаления и добавления [5] и  $G_{gru}$  – управляемый рекуррентный блок [6];  $f_{neighbor}$  – функция распространения по соседним узлам графа.

3. Предсказание (predict). Определение прогнозируемой вероятности учащегося, правильно выполняющий каждое задание на шаге:

$$y_k^t = \sigma(W_{out} h_k^{t+1} + b_k) \quad (3)$$

где  $W_{out}$  – матрица весов для всех узлов графа;  $b_k$  – смещение узла  $k$ ,  $\sigma$  – это функция сигмоида. Полученная модель минимизирует оценку функции отрицательного логарифмического правдоподобия (Negative Log-Likelihood) [7].

Для тестирования был выбран один из открытых наборов данных (dataset) PISA [8] об обучения студентов в высших учебных учреждениях. Каждый набор данных включает в себя три выборки: обучение (training), валидация (validation), тестирование (test) в соотношении 8:1:1. В качестве метрики оценки была выбрана площадь под кривой AUC (Area Under Curve) [10].

Метод DKT [9] тестировался на нейронная сети с 200 скрытыми слоями и коэффициентом обучения 0.001. Метод DKVMN (Dynamic Key-Value Memory Network) [5] тестировался на нейронной сети с размером 128 скрытыми слоями и коэффициентом обучения 0.01 [11]. Предложенный метод тестировался на нейронной сети с 180 скрытыми слоями и коэффициентом обучения 0.01.

Таблица 1 – Сравнения результатов предсказаний

Метод		Площадь под кривой (Area Under Curve)	
		PISA 2015	PISA 2018
Базовые	DKT	0.709	0.751
	DKVMN	0.710	0.753
Статистические	Плотный граф	0.722	0.762
	Граф Состояний	0.721	0.769
	Граф DKT	0.723	0.764
Предложенный метод		<b>0.698</b>	<b>0.733</b>

Исходя из результатов тестирования (таблица 1) можно сделать вывод, что предложенный метод оказался эффективнее своих предшественников.

В данной работе рассмотрен метод по подбору учебных программ в образовательных системах, который учитывает структуру учебной программы в виде графа и прогнозирует результаты обучения учащихся.

Проведя тесты, было установлено, что метод эффективен по сравнению со своими предшественниками и помогает лучше прогнозировать результат обучения учащихся по сгенерированной учебной программе.

Сравнения работы методов предсказаний на примере двух аспектов представлено в таблице 2.

Таблица 2 – Сравнения работы методов предсказаний

Метод	Аспект	
	Определение текущего прогресс обучения учащегося $h^t$	Взаимодействие между учебными заданиями в процессе обучения
DKT	Единичный скрытый вектор	Отсутствует
DKVMN	Множество скрытых векторов	Между входными и выходными концептами
Предложенный метод	Пары скрытых векторов	Между произвольными концептами

Результаты данного исследования могут быть полезны в применении в различных сферах образования и могут помочь улучшить построение образовательного процесса.

### Литература

1. Marco Gori, Gabriele Monfardini, Frank Scarselli, A new model for learning in graph domains. In Neural Networks. – 2005.
2. Will Hamilton, Zhitao Ying, Jure Leskovec. Inductive representation learning on large graphs. In Advances in Neural Information Processing Systems. – 2017.
3. Peter Battaglia, Jessica Hamrick, Victor Bapst. Relational inductive biases, deep learning and graph networks. – 2018.
4. Multilayer perceptron [Электронный ресурс]. – Режим доступа: <https://en.wikipedia.org/wiki/Multiplayer-perceptron>.
5. Jiani Zhang, Xingjian Shi, Irwin King, Dit-Yan Yeung. Dynamic Key-Value Memory Network for Knowledge Tracing. – 2016.
6. Dzmitry Bahdanau, Yoshua Bengio, Fethi Bougares. Learning phrase representations using RNN encoder-decoder for statistical machine translation. – 2014.
7. С. Николаенко, А. Кадури, Е. Архангельская. Глубокое обучение. Погружение в мир нейронных сетей. – 2018.
8. PISA (Programme for International Student Assessment) [Электронный ресурс]. – Режим доступа: <https://oecd.org/pisa/data>.
9. Chris Piech, Jonathan Bassen. Deep knowledge tracing. In Advances in Neural Information Processing Systems. – 2015.
10. AUC (Area Under Curve) [Электронный ресурс]. – Режим доступа: <https://u.to/wbnbfG>.
11. Реализация метода DKVMN [Электронный ресурс]. – Режим доступа: <https://github.com/jennyzang0215/dkvmn>.