Exploring deep learning models for curriculum analytics

Gabriel Astudillo
Computer Science Department
Pontifical Catholic University of Chile
Santiago, Chile
gastudillo@uc.cl

Isabel Hilliger
Engineering Design Unit
Pontifical Catholic University of Chile
Santiago, Chile
ihillige@uc.cl

Jorge Baier
Computer Science Department
Pontifical Catholic University of Chile
Santiago, Chile
jbaier@uc.cl

***Abstract*—While curricular flexibility in engineering programs enhances student autonomy, it introduces significant complexity in academic planning for students and institutions. Ineffective decision-making regarding course sequences can hinder academic progression and increase the risk of dropout. Consequently, analyzing curricular trajectories is essential for supporting student success. This study explores the application of deep learning models to model course enrollment sequences at the School of Engineering of the Pontifical Catholic University of Chile, using a dataset of 11,729 students between 2013 and 2025. We compare two neural network architectures—Bidirectional Long Short-Term Memory (Bi-LSTM) and Bidirectional Encoder Representations from Transformers (BERT)—against a K-Nearest Neighbors (KNN) baseline in a masked course prediction task. Our results demonstrate that BERT significantly outperforms the alternative models. This superiority is attributed to BERT's ability to model concurrent course enrollments within a single academic period through positional encoding. These findings suggest that BERT-based architectures are highly effective for understanding complex curricular trajectories, offering a robust foundation for course recommendation systems and student progression analytics. This approach opens new avenues for identifying "bottlenecks," predicting timely graduation, and mitigating dropout risk in flexible educational environments.**

***Index Terms*—Curriculum Analytics, Deep learning, Engineering Education**

I. Introduction

During the last decades, engineering programs in Latin America and Chile have moved toward more flexible curricula—that is, granting students a higher degree of autonomy in decisions regarding what, how, and when to learn [1], [2]. This helps study plans become more adaptable to the learning needs of different groups of students [3].

However, as the breadth of the curricular offer increases, so does the complexity of the decisions students must make: they do not necessarily have reliable information about the contents addressed, required knowledge, difficulty, or other aspects of the courses [4], [5]. This not only makes it difficult for students to take decisions about their learning, but also has consequences for their subsequent academic development: performance and risk of dropout. Authors such as Heileman et al. [6] have argued that progression within a study plan constitutes the basis of academic success, because obstacles in that process can delay graduation or lead to dropping out of programs. From this point of view, it is important for educational institutions to analyze the established curricular sequences and how these facilitate or obstruct academic success, for example, through "bottlenecks" in the curriculum. For teachers, understanding the role their courses occupy within the study program offers relevant information they should consider when designing their courses [4], [7].

Hence, research on curricular analytics seeks to analyze data on students' academic trajectories through the program to support decision-making for all related stakeholders, especially those linked to planning: for students, which courses to enroll in and when; for instructors, which subjects to teach; and for program administrators, which interventions are necessary at the program level [4], [6], [8], [9].

This study explores the use of different machine learning models to understand the academic trajectories of Engineering students in an Engineering program with a highly flexible curriculum, through the prediction of course enrollment at a specific point in the curricular progression.

A. Related work

In the literature, at least three different methodologies have been explored for the analysis of curricular sequences: statistics, process mining, and deep learning models. Within the first approach, Heileman et al. [6] propose measuring different obstructions to curricular progression such as delay, blocking, and course reach.

Srivastava et al. [4] use a combination of approaches to analyze students' curricular trajectories. Through the temporal position of a course and the temporal distance between courses, they analyze the temporal factor, while they analyze course passing profiles to measure difficulty and understand how different student profiles follow different curricular trajectories. Furthermore, they use the L-statistic to analyze in terms of probability how past courses influence future courses.

Other works have explored process mining techniques such as Directly-Follows Graph [10] or Petri Nets [11]. These models, based on descriptive data, represent frequent trajectories in the form of directed nodes that allow the human analyst to identify relationships with different outcomes (e.g., successful graduation, dropout). There, course recurrence (edges that return to the same node from which they originated) allows for the quick identification of bottlenecks in the curriculum, while deviations from expected trajectories can contribute to identifying students who are experiencing delays in their curricular progression.

These models have two limitations. On one hand, they depend heavily on human interpretation for pattern discovery; on the other, they lose analytical power in curricular contexts that are too flexible because the diversity of trajectories increases, which does not necessarily provide relevant information.

A third group of works has employed deep learning neural network models. There are two architectures that have been objects of interest in curricular analytics due to their ability to learn sequential relationships and implicit rules in the data without the need to declare them explicitly: Long-Short Term Memory (LSTM) and Transformer.

LSTM models are characterized by (1) representing sequences as a succession of hidden states that are updated at each time step, and (2) having memory units that make information from previous steps accessible [12]. On this basis, bidirectional variants (Bi-LSTM, [13]) allow for the incorporation of information about both previous and subsequent states, in order to make a prediction about an intermediate state.

In curricular analytics, experimentation has been conducted with both variants to analyze academic trajectories with better performance than other architectures used in sequential problems, such as Recurrent Neural Networks or GRU [14], although a Bi-LSTM model seems to outperform the simple LSTM [17]. However, these architectures introduce a distortion when trying to represent curricular sequences. The internal states of an LSTM or Bi-LSTM network can only represent a single element for each position, which breaks the curricular logic where students enroll in several courses per academic period. In other words: there are several elements in the same position of the sequence.

Given this limitation, the Transformer architecture [15] offers an interesting alternative because it uses a positional encoding method that allows for the incorporation of information about the order of elements in the sequence. In this way, multiple elements can occupy the same position, which is more suitable for modeling the problem of curricular sequences. Additionally, the Transformer uses an attention mechanism to capture distant dependencies throughout the sequence.

Devlin et al. [16] used this architecture in their Bidirectional Encoder Representations from Transformers (BERT) model, designed to encode the semantic information of text. BERT is trained on a task of predicting random words masked in the sequence, for which semantic understanding of the context is necessary.

The work of Shao et al. [17] performed experiments with different variants of a BERT-type architecture, incorporating future reference elements and complementary information about courses and students. They observed that these BERT variants far outperformed both the K-Nearest-Neighbors (KNN) baseline and other deep learning models such as LSTM and Bi-LSTM.

Despite all the methodological advances to understand curricular sequences, this aspect has not been fully explored in engineering education. In this field, curricular analytics work has emphasized the focus on measuring the achievement of competencies in a given curricular sequence [18]–[20], over the analysis of the curricular sequence followed by each student subject to the curricular design of their institution. In Chile, taking into consideration efforts to shorten engineering study plans [21]–[23], this line of work can have great potential for transferring findings to curricular discussions in practice.

II. Methods

The objective of this work is to explore the application of deep learning models to understand the curricular sequences of students in an Engineering school with high curricular flexibility.

A. Data

The data used correspond to course enrollments by undergraduate students from all Engineering specialties at the Pontifical Catholic University of Chile, covering the period between 2013 and the first semester of 2025. Both mandatory and elective Engineering courses are included, while elective courses taught by other faculties are excluded, as well as students from other programs who have enrolled in Engineering courses. This constitutes a total volume of 355,751 student-course records, distributed in 11,729 unique sequences corresponding to students. In order to evaluate the generalization capacity of the models, the data were divided into a training partition (70%) and a testing partition (30%) used for the calculation of all evaluation metrics for each model.

B. Experimental design

Following the standard in the field of natural language processing, a prediction task of masked elements in the sequence—in this case, courses—is defined. Just as in BERT training [16], for the model to correctly predict the masked element, it must be able to represent the essential information of the sequence. For this task, 15% of the courses in each student's sequence were randomly masked.

As a baseline, a classic machine learning algorithm was used: K-Nearest Neighbors (KNN, [24]), against which the results of two deep learning architectures are compared: Bi-LSTM and BERT.

KNN is a machine learning technique where, from the K nearest neighbors in a given feature space, an output variable is predicted [24]: in this case, masked courses. In the implementation for this work, course enrollment sequences were transformed into a matrix of presence or absence of all courses for each student (known as one-hot encoding). This representation omits the order of elements in the sequence. To make the prediction of the masked element, the column corresponding to the masked course is discarded in this representation. Through the Jaccard coefficient, similarity is estimated between one observation and all others. From the k observations with the highest similarity coefficient, the prediction is made based on the most frequent courses in those near neighbors.

For the training of the deep learning models, an AdamW optimizer [25] was used, with a 15% dropout and different learning rates: 3e-5 for Bi-LSTM, and 5e-5 for BERT. The only particularity to highlight is that for the BERT model, positional encoding associated with the student's relative semester was used, such that all courses from the same period are represented in the same position by the model. Each architecture was trained with multiple hyperparameter setups (latent vector dimensionality, number of layers, head attentions, etc.) and the results section reports the performance obtained by the best configuration of each model.

For each of these models, we experimented with different hyperparameter configurations, the details of which are presented in Table I. We tested two KNN models with 5 and 10 neighbors (K). For the Bi-LSTM and BERT architectures, we tested three models by increasing the number of parameters through the hidden dimensionality (ranging from 32 to 256) and the number of layers (from 1 to 4). Additionally, the Transformer architecture features an extra hyperparameter: the number of attention heads (H). We experimented with between 1 and 8 attention heads, which does not increase the model size; instead, it divides the latent vector into subspaces.

TABLE I
HYPERPARAMETERS SETTINGS

Architecture	K	Layers	Hidden Dim.	Attention heads
KNN	5	-	-	-
KNN	10	-	-	-
Bi-LSTM	-	1	32	-
Bi-LSTM	-	2	64	-
Bi-LSTM	-	3	128	-
Bi-LSTM	-	4	256	-
BERT	-	1	32	1
BERT	-	2	64	2
BERT	-	3	128	4
BERT	-	4	256	8

Following Shao et al. [17], the evaluation of masked course predictions across all models is based on Recall and Normalized Discounted Cumulative Gain (NDCG). Both metrics are estimated over the top 10 elements to which the model assigns the highest probability (denoted as @10).

- Recall@10. Proportion of masked elements that were present within the 10 elements predicted by the model, relative to the total number of masked elements.
- NDCG@10. Measures the ability of the predictions to rank relevant versus irrelevant elements. It takes a higher value when the masked element appears with higher probability and decreases as the element appears with lower probability.

III. RESULTS

The performance of the various models in predicting masked courses is summarized in Table II. The baseline KNN models (K=5 and K=10) achieved a Recall@10 of approximately 28%. Interestingly, the smaller Bi-LSTM configurations (Layers 1 and 2) performed similarly to the KNN baseline, with performance peaking at two layers (Recall@10 = 0.2910). Subsequently, as the number of parameters continues to increase, a decrease in Recall is observed, reaching 0.2685 in the 4-layer model. Similarly, the NDCG@10 shows that the KNN models have a discriminative capacity similar to that of the Bi-LSTM models ($\approx$ 0.13).

TABLE II
PREDICTIVE PERFORMANCE ACROSS DIFFERENT HYPERPARAMETER CONFIGURATIONS AND ARCHITECTURES

Model	Layers	H. Dim.	Recall@10	NDCG@10
KNN (K=5)	-	-	0.2800	0.1309
KNN (K=10)	-	-	0.2839	0.1353
Bi-LSTM	1	32	0.2833	0.1343
Bi-LSTM	2	64	0.2910	0.1372
Bi-LSTM	3	128	0.2792	0.1295
Bi-LSTM	4	256	0.2685	0.1245
BERT (H=1)	1	32	0.3190	0.1608
BERT (H=2)	2	64	0.4050	0.1916
BERT (H=4)	3	128	0.3918	0.1891
BERT (H=8)	4	256	**0.4204**	**0.2089**

Note: H represents the number of attention heads in the BERT architecture, and K the number of neighbors for KNN. **Bold** values indicate the best performance for each metric.

However, the BERT architecture significantly outperformed both KNN and Bi-LSTM across all configurations. The best-performing BERT model (4 layers, 256 hidden dimensions, 8 attention heads) achieved a Recall@10 of 0.4204 and an NDCG@10 of 0.2089. Compared to previous models, this implies not only a greater capacity to predict masked courses but also a significantly higher discriminative power for relevant elements.

IV. DISCUSSION

The objective of this work was to explore the application of deep learning models to understand the curricular sequences of students in an Engineering program with high curricular flexibility. To this end, a methodology for predicting random masked courses within students' curricular sequences was implemented, and two deep neural network architectures (Bi-LSTM and BERT) were compared against a baseline of a classic machine learning model, as it is K-Nearest Neighbors. Considering the predictive performance demonstrated by the different models (refer to Table II), the BERT Model was the best setup to understand course sequences within a flexible engineering curriculum. Substantively, this implies that the model offers the best capacity to capture the latent information in the student's curricular sequence. Not only do these results align with what was reported by Shao et al. [17], but also have practical implications to inform curriculum analysis in different contexts.

By contrast, the Bi-LSTM architecture showed an improvement in performance when moving from 1 to 2 layers; however, as more parameters were added to the model, its performance began to degrade. Normally, the most likely reason for this decline would be overfitting, where the model stops learning general patterns and loses its capacity for generalization. Nevertheless, given that the BERT architecture does not show this decline and continues to improve its performance as the number of parameters increases, it is probable that this difference is due specifically to how both architectures process sequences: for the Bi-LSTM, the order of courses within the same semester is relevant, whereas BERT models process patterns in a way that is more faithful to the reality of our problem, where the order within a single position (semester) does not matter—order is only relevant between positions within the sequence.

From a broader engineering education perspective, the superior performance of BERT can be interpreted not only as a technical advantage but as an epistemic alignment with contemporary notions of curricular flexibility. Lucena [26] conceptualizes flexibility in engineering education as a response to historically rigid curricular structures that privilege linear progression, engineering sciences, and predefined problem-solving methods. In highly flexible curricula, students' trajectories no longer conform to a single ordered pathway but instead reflect multiple, partially ordered combinations of learning experiences. Models that assume a strictly sequential structure impose epistemological constraints that mirror traditional pipeline-oriented curricula.

This evidence is especially relevant in the context of any engineering program with a highly flexible curriculum, where the curricular offer also implies difficulties for the decision-making of students, managers, and student support units [4], [5]. A BERT model can be a useful tool to support the decisions and planning of all actors involved. The masked course prediction mechanism could be directly utilized as a course recommendation system for students and support professionals.

Furthermore, akin to BERT models in Natural Language Processing [16], [27], to the extent that the model is capable of understanding a student's curricular sequence, the latent representation produced could be repurposed for other prediction tasks associated with the curriculum and students' individual trajectories. Regarding the former, it is possible to identify which circumstances constitute "bottlenecks" for timely graduation [6]; while regarding student sequences, it can be leveraged for the prediction of timely graduation or dropout [4], [6].

Related to the above is the main limitation of this study: the potential of these models to simultaneously address several curricular analytics tasks is still a theoretically plausible but unexplored promise. As a consequence, there are two directions for future work that open up with this research. First, evaluating the ability of these models to understand delays at the student level, bottlenecks in program design [4], [6], the academic trajectories of different student profiles [4], or how curricular sequences influence different outcomes of the educational process—successful graduation, dropout— [10].

Second, developing tools that facilitate end-users (students, student support professionals, and managers) in making effective use of the information encoded by the model to make decisions either at the individual-student level or through institutional interventions at the study-plan level.

Acknowledgment

References

[1] Chhatria, S., Pani, B., Digal, A. K. M. S., & Sangeeta, M. (2024). Principles and Approaches to Curriculum Development. . . Journal of Higher Education Theory and Practice, 24(1).

[2] Tucker, R., & Morris, G. (2011). Anytime, anywhere, anyplace: Articulating the meaning of flexible delivery in built environment education. British Journal of Educational Technology, 42(6), 904–915. https://doi.org/10.1111/j.1467-8535.2010.01138.x

[3] Jonker, H., März, V., & Voogt, J. (2020). Curriculum flexibility in a blended curriculum. Australasian Journal of Educational Technology. https://doi.org/10.14742/ajet.4926

[4] Srivastava, N., Nawaz, S., Tsai, Y.-S., & Gašević, D. (2024). Curriculum Analytics of Course Choices: Journal of Learning Analytics, 11(1), 116–131. https://doi.org/10.18608/jla.2024.8095

[5] Sutton, K. L., & Sankar, C. S. (2011). Student Satisfaction with Information Provided by Academic Advisors. Journal of STEM Education: Innovations and Research, 12(7), 71–85.

[6] Heileman, G. L., Abdallah, C. T., Slim, A., & Hickman, M. (2018). Curricular Analytics: A Framework for Quantifying the Impact of Curricular Reforms and Pedagogical Innovations. ArXiv Preprint.

[7] Dennehy, D., Conboy, K., & Babu, J. (2023). Adopting Learning Analytics to Inform Postgraduate Curriculum Design: Recommendations and Research Agenda. Information Systems Frontiers, 25(4), 1315–1331. https://doi.org/10.1007/s10796-021-10183-z

[8] Greer, J., Molinaro, M., Ochoa, X., & McKay, T. (2016). Learning analytics for curriculum and program quality improvement (PCLA 2016). Proceedings of the Sixth International Conference on Learning Analytics & Knowledge - LAK '16, 494–495. https://doi.org/10.1145/2883851.2883899

[9] Ochoa, X. (2016). Simple metrics for curricular analytics. In J. Greer, M. Molinaro, X. Ochoa, & T. McKay (Eds.), Proceedings of the First Learning Analytics for Curriculum and Program Quality Improvement Workshop, co-located with the Sixth International Learning Analytics and Knowledge Conference (LAK 2016), 25 April 2016, Edinburgh, UK (pp. 20–26, Vol. 1590).

[10] Salazar-Fernandez, J. P., Sepúlveda, M., Munoz-Gama, J., & Nussbaum, M. (2021). Curricular Analytics to Characterize Educational Trajectories in High-Failure Rate Courses That Lead to Late Dropout. Applied Sciences, 11(4), 1436. https://doi.org/10.3390/app11041436

[11] Martinez, P., Montañes, O., Serralta, J. M., & Tansini, L. (2020). Modelling Computer Engineering Student Trajectories with Process Mining. IV LATIN AMERICAN CONFERENCE ON LEARNING ANALYTICS.

[12] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[13] Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 45(11), 2673–2681. https://doi.org/10.1109/78.650093

[14] Yu, R., Jiang, D., & Warschauer, M. (2018). Representing and predicting student navigational pathways in online college courses. Proceedings of the Fifth Annual ACM Conference on Learning at Scale, 1–4. https://doi.org/10.1145/3231644.3231702

[15] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need.

[16] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers) (pp. 4171-4186).

[17] Shao, E., Guo, S., & Pardos, Z. A. (2021). Degree Planning with PLAN-BERT: Multi-Semester Recommendation Using Future Courses of Interest. Proceedings of the AAAI Conference on Artificial Intelligence, 35(17), 14920–14929. https://doi.org/10.1609/aaai.v35i17.17751

[18] Barthakur, A., Kovanović, V., Dawson, S., & Deneen, C. C. (2024). The application of curriculum analytics for improving assessments and quality assurance in higher education. Australasian Journal of Educational Technology. https://doi.org/10.14742/ajet.9383

[19] Hernández-Campos, M., Hilliger, I., & García-Peñalvo, F.-J. (2025). Evaluating Learning Outcomes Through Curriculum Analytics: Actionable Insights for Curriculum Decision-making. Proceedings of the 15th International Learning Analytics and Knowledge Conference, 384–394. https://doi.org/10.1145/3706468.3706518

[20] Hilliger, I., Aguirre, C., Miranda, C., Celis, S., & Pérez-Sanagustín, M. (2022). Lessons learned from designing a curriculum analytics tool for improving student learning and program quality. Journal of Computing in Higher Education, 34(3), 633–657. https://doi.org/10.1007/s12528-022-09315-4

[21] Said, C. (2019, April 9). Universidades avanzan en acortar la duración de carreras de ingeniería. La Tercera.

[22] Zanetta, R. (2019, May 2). Debate sobre duración de las carreras universitarias: ¿En qué está la UC? https://www.uc.cl/.

[23] Admisión UTEM. (2023, January 12). Nueva duración de carreras universitarias de Ingeniería y de Ciencias de la Construcción. https://admision.utem.cl/.

[24] Cunningham, P., & Delany, S. J. (2021). k-Nearest Neighbour Classifiers - A Tutorial. ACM Comput. Surv., 54(6).

[25] Loshchilov, I., & Hutter, F. (2019, January 4). Decoupled Weight Decay Regularization. International Conference on Learning Representations.

[26] Lucena, J. C. (2003). Flexible Engineers: History, Challenges, and Opportunities for Engineering Education. Bulletin of Science, Technology & Society, 23(6), 419-435. https://doi.org/10.1177/0270467603259875

[27] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.