A Impact of directed graph feature fusion methods

In this study, we explore the impact of different fusion methods for the concept embeddings generated by the out-degree and in-degree graphs during the user behavior graph modeling process. Specifically, we conduct a comparative analysis of the following five fusion strategies:

- **Addition**: The embeddings generated by the out-degree and in-degree graphs are added element-wise, directly combining the information from both.
- **Concat**: The embeddings from the out-degree and in-degree graphs are concatenated, and then passed through an MLP to reduce the concatenated embedding to the same dimension, achieving a comprehensive fusion of the information.
- **Maximum**: The embeddings generated by the out-degree and in-degree graphs are combined by taking the element-wise maximum, retaining the maximum information from each dimension.
- **Average**: The embeddings generated by the out-degree and in-degree graphs are combined by taking the element-wise average, balancing the information from each dimension.
- **Minimum**: The embeddings generated by the out-degree and in-degree graphs are combined by taking the element-wise minimum, retaining the minimum information from each dimension.

Table 1 shows the impact of directed graph feature fusion methods on our model. On the UCD dataset, the element-wise addition method achieved a 1.49% improvement in ACC and a 0.42% improvement in F1 compared to other suboptimal methods, although the AUC was slightly lower than the suboptimal methods. On the LectureBank dataset, ACC and F1 were higher by 0.83% and 0.41%, respectively, compared to the next best element-wise maximum method. On the MOOC dataset, the element-wise addition method performed the best, with ACC of 0.8756, F1 of 0.8718, and AUC of 0.9236.

In summary, we can conclude the following: (1) The Addition method performs the best in terms of ACC and F1 across all datasets, indicating that the simple vector addition strategy can effectively combine the features from the out-degree and in-degree graphs, making it suitable for different types of datasets. However, its slightly lower AUC compared to other methods may be due to the over-smoothing of some features. The excessive averaging of features may hinder fine-grained differentiation, thus affecting AUC. (2) The Concat+MLP method is a more complex fusion strategy. Although it captures more rich information, its complexity does not significantly improve the performance. (3) The Maximum and Average methods performed moderately, suggesting that while they retain certain features, they fail to capture the global information of both out-degree and in-degree features fully. (4) The Minimum method, which retains the minimum values, caused a significant loss of feature information and negatively impacted the model's performance.

B Baseline method detailed description

For a comprehensive assessment of the performance and effectiveness of our proposed model, this study compares it with several currently popular concept prerequisite relation prediction models.

Traditional Baselines:

- **Classification Models**: We use three traditional classification models: Naive Bayes (NB), Support Vector Machine (SVM), and Random Forest (RF). In these models, we concatenate the features of the source concept and the target concept as input data and train the models using the corresponding prerequisite relation labels to perform the classification task.
- **RefD**: The RefD model effectively measures the prerequisite relations between concepts by utilizing TF-IDF values to set concept weights and employing citation-based metric methods.
- **PREREQ**: The PREREQ model uses a pairwise latent Dirichlet allocation model to obtain the latent representations of concepts and employs a Siamese network to identify concept prerequisite relations.

Graph-based Baselines:

- **GAE**: The GAE model is an effective graph autoencoder based on graph convolutional networks, which predicts links through the reconstruction of the adjacency matrix, enabling concept prerequisite relation learning.
- **VGAE**: As an extension of GAE, the VGAE model introduces a variational inference mechanism, providing an unsupervised learning framework for concept prerequisite relation learning in unstructured data.

Operation	UCD			LectureBank			MOOC		
	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC
Addition (Ours)	**0.8564**	**0.8557**	0.9053	**0.8347**	**0.8246**	0.8795	**0.8756**	**0.8718**	**0.9236**
Concat+MLP	0.8416	0.8416	**0.9074**	0.8099	0.8099	0.8776	0.8557	0.8497	0.9211
Maximum	0.8515	0.8515	0.9048	0.8264	0.8205	0.8724	0.8657	0.8601	0.9190
Average	0.8416	0.8384	0.9011	0.8017	0.7778	**0.8842**	0.8657	0.8643	0.9214
Minimum	0.8515	0.8469	0.9058	0.7851	0.7636	0.8727	0.8507	0.8485	0.9134

Table 1: Impact of directed graph feature fusion methods. The best performance is highlighted in **bold**, and the runner-up is underlined.

Method	UCD			LectureBank			MOOC		
	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC
DGCPL (Ours)	**0.8564**	**0.8557**	**0.9053**	**0.8347**	**0.8246**	**0.8795**	**0.8756**	**0.8718**	**0.9236**
LCPRE	0.8366	0.8216	0.8884	0.8182	0.8000	0.8514	0.8258	0.8223	0.8898
HGAPNet	0.8400	0.8333	0.9044	0.7833	0.7759	0.8672	0.7900	0.7835	0.8573
ConLearn	0.7822	0.7684	0.8529	0.8017	0.7931	0.8541	0.7562	0.7200	0.8472

Table 2: Impact of pre-training language model on performance. The best performance is highlighted in **bold**, and the runner-up is underlined.

- **R-VGAE(T)**: The R-VGAE(T) model combines R-GCN and VGAE, using frequency and inverse document frequency-based features to enhance the model's ability to predict concept prerequisite relations.
- **R-VGAE(P)**: Similar to R-VGAE(T), the R-VGAE(P) model is based on R-GCN and VGAE but uses dense vector representations of phrases as input, further optimizing the concept relation identification process.
- **ConLearn**: The ConLearn model is a context-aware approach for concept prerequisite relation learning, capturing the prerequisite relations between concepts using self-attention and graph neural networks.
- **MHAVGAE**: The MHAVGAE model employs multi-head attention and gating mechanisms to enhance concept representations, and finally predicts concept relations through a variational autoencoder.
- **HGAPNet**: The HGAPNet model proposes a graph neural network-based approach, focusing on extracting prerequisite concepts for a given concept by utilizing node attention mechanisms on heterogeneous graphs.
- **LCPRE**: The LCPRE model represents the relations between concepts as directed edges, building learning paths based on graphs. It also uses graph neural networks to extract feature interactions between concepts and captures concept prerequisite relations.

C Impact of pre-training language model on model performance

In this study, we explore the impact of pre-trained language models on model performance. To ensure fairness in the experiments and comparability of results, our model, DGCPL, uses the same pre-trained model, BERT, to generate initial embeddings as the three advanced baseline models: LCPRE, HGAPNet, and ConLearn. By using the same initial embedding generation method across all models, we ensure consistency in the basic representations, allowing for a more accurate evaluation of the specific impact of the initial embeddings on model performance.

As shown in Table 2, our DGCPL achieves the best performance in terms of ACC, F1, and AUC metrics. On the UCD dataset, compared to the HGAPNet model, DGCPL achieves an ACC improvement of 1.64% and an F1 improvement of 2.24%. On the LectureBank dataset, DGCPL also performs best, with an ACC improvement of 1.65% over the second-best model, LCPRE, and an F1 improvement of 2.46%. Additionally, DGCPL's AUC is 1.23% higher than that of HGAPNet. On the MOOC dataset, DGCPL achieves an ACC of 0.8756, an F1 of 0.8718, and an AUC of 0.9236.

From the comprehensive experimental results across the three datasets, we can conclude: (1) The DGCPL model outperforms the existing baseline models in terms of ACC, F1, and AUC, validating its generalization ability when handling diverse datasets and tasks. (2) The pre-trained language model BERT provides some performance improvement, but it is not the key factor influencing model performance. Despite using the same pre-trained language model across all four models, DGCPL still outperforms all others across all datasets. This indicates that DGCPL's model design has a distinct advantage in capturing concept relations and feature representations.