Unsupervised Cross-Domain Prerequisite Chain Learning using Variational Graph Autoencoders

Irene Li, Vanessa Yan, Tianxiao Li, Rihao Qu and **Dragomir Radev**
Yale University, USA
{irene.li,vanessa.yan,tianxiao.li,rihao.qu,dragomir.radev}@yale.edu

Abstract

Learning prerequisite chains is an essential task for efficiently acquiring knowledge in both known and unknown domains. For example, one may be an expert in the natural language processing (NLP) domain but want to determine the best order to learn new concepts in an unfamiliar Computer Vision domain (CV). Both domains share some common concepts, such as machine learning basics and deep learning models. In this paper, we propose unsupervised cross-domain concept prerequisite chain learning using an optimized variational graph autoencoder. Our model learns to transfer concept prerequisite relations from an information-rich domain (source domain) to an information-poor domain (target domain), substantially surpassing other baseline models. Also, we expand an existing dataset by introducing two new domains—CV and Bioinformatics (BIO). The annotated data and resources, as well as the code, will be made publicly available.

1 Introduction

With the rapid growth of online educational resources in diverse fields, people need an efficient way to acquire new knowledge. Building a concept graph can help people design a correct and efficient study path (ALSaad et al., 2018; Yu et al., 2020). There are mainly two approaches to learning prerequisite relations between concepts: one is to extract the relations directly from course content, video sequences, textbooks, or Wikipedia articles (Yang et al., 2015b; Pan et al., 2017; Alzetta et al., 2019), but this approach requires extra work on feature engineering and keyword extraction. Our method follows a different approach of inferring the relations within a concept graph (Liang et al., 2018; Li et al., 2019, 2020).

In a concept graph, we define $p \to q$ as the notion that learning concept p is a prerequisite to learning concept q. Existing methods formulate this question as a classification task. A typical method is to encode concept pairs and train a classifier to predict if there is a prerequisite relation (Alzetta et al., 2019; Yu et al., 2020). However, this method requires annotated prerequisite pairs during training. Alternatively, others have used graph-based models to predict prerequisite relations. Gordon et al. (2016) proposed information-theoretic approaches to infer concept dependencies. Li et al. (2019) modeled a concept graph using Variational Graph Autoencoders (VGAE) (Kipf and Welling, 2016), training their model to infer unseen prerequisite relations in a semi-supervised way. While most of the previous methods were supervised or semi-supervised, Li et al. (2020) introduced Relational-VGAE, which enabled unsupervised learning on prerequisite relations.

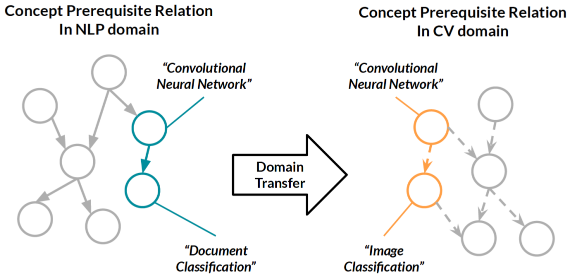
Concept Prerequisite Relation
In NLP domain
Concept Prerequisite Relation
In CV domain
"Convolutional Neural Network"
"Convolutional Neural Network"
Domain Transfer
"Document Classification"
"Image Classification"

Figure 1: Cross-domain prerequisite chains.

Existing work mainly focuses on prerequisite relations within a single domain. In this paper, we tackle the task of cross-domain prerequisite chain learning, by transferring prerequisite relations between concepts from a relatively information-rich domain (source domain) to an information-poor domain (target domain). As an example, we illustrate in Figure 1, a partial concept graph from the Natural Language Processing (NLP) domain and a partial concept graph from the Computer Vision (CV) domain. Prerequisite relations among concepts in the NLP domain are known, and we seek to infer prerequisite relations among concepts in

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Short Papers), pages 1005–1011
August 1–6, 2021. ©2021 Association for Computational Linguistics

the CV domain. These two domains share some concepts, such as *Convolutional Neural Network*. We assume that being aware of prerequisite relations among concepts in the source domain helps infer potential relations in the target domain. More specifically, in the figure, knowing that *Convolutional Neural Network→Document Classification* helps us determine that *Convolutional Neural Network→Image Classification*.

Our contributions are two-fold. First, we propose cross-domain variational graph autoencoders to perform unsupervised prerequisite chain learning in a heterogeneous graph. Our model is the first to do domain transfer within a single graph, to the best of our knowledge. Second, we extend an existing dataset by collecting and annotating resources and concepts in two new target domains. Data and code will be made public in `https://github.com/Yale-LILY/LectureBank/tree/master/LectureBankCD`.

2 Dataset

LectureBank2.0 (Li et al., 2020) dataset contains 1,717 lecture slides (hereon called **resources**) and 322 concepts with annotated prerequisite relations, largely from NLP. We treat this dataset as our information-rich source domain (NLP). Also, we propose an expansion dataset, LectureBankCD, by introducing two new target domains in the same data format: CV and Bioinformatics (BIO). We report statistics on the dataset in Table 1. For each domain, we identify high-quality lecture slides from the top university courses, collected by domain experts, and we choose concepts by crowd-sourcing. We end up with 201 CV concepts and 100 BIO concepts. In each domain, we ask two graduate-level annotators with deep domain knowledge to add prerequisite chain annotations for every possible pair of concepts. The Cohen's kappa agreement scores (McHugh, 2012) are 0.6396 for CV and 0.8038 for BIO. Cohen's kappa between 0.61–0.80 is considered substantial, so our annotations are reliable.

Domain	Files	Pages	Tks/pg	Con.	PosRel
NLP	1,717	65,028	47	322	1,551
CV	1,041	58,32	43	201	871
BIO	148	7,13	135	100	234

Table 1: LectureBankCD statistics on NLP, CV and BIO domain: Tks/pg (Tokens per slide page), Con. (Number of concepts), PosRel (Positive Relations).

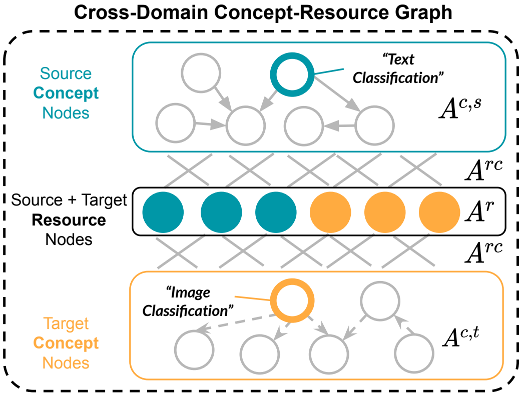

Figure 2: Cross-Domain Concept-Resource Graph: we model the resource nodes (solid nodes) and concept nodes (hollow nodes) from two domains (in blue and orange) in a heterogeneous graph. We show a subset of nodes and edges.

We take the union of the positive annotations for our experiments: 871 positive relations for CV and 234 positive relations for BIO.

3 Methodology

Inspired by Li et al. (2020), we build a cross-domain concept-resource graph $G = (X, A)$ that includes resource nodes and concept nodes from both the source and target domains (Figure 2). To obtain the node feature matrix X, we use either BERT (Devlin et al., 2019) or Phrase2Vec (Artetxe et al., 2018) embeddings. We consider four edge types to build the adjacency matrix A: $A^{c,s}$: edges between source concept nodes; A^{rc}: edges between all resource nodes and concept nodes; A^{r}: edges between resource nodes only; and $A^{c,t}$: edges between target concept nodes. In unsupervised prerequisite chain learning, $A^{c,s}$—concept relations of the source domain—are known, and the task is to predict $A^{c,t}$—concept relations of the target domain. For A^{rc} and A^{r}, we calculate cosine similarities based on node embeddings, consistent with previous works (Li et al., 2019; Chiu et al., 2020).

Cross-Domain Graph Encoder VGAE (Kipf and Welling, 2016) contains a graph neural network (GCN) encoder (Kipf and Welling, 2017) and an inner product decoder. In a GCN, the hidden representation of a node i in the next layer is computed using only the information of direct neighbours and the node itself. To account for cross-domain knowledge, we additionally consider the *domain neighbours* for each node i. These domain neighbours are a set of common or semantically similar concepts from the other domain.[1] We define the cross-domain graph encoder as:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in N_i} W^{(l)} h_j^{(l)} + W^{(l)} h_i^{(l)} \right.$$
$$\left. + \sum_{k \in N_i^D} W_D^{(l)} h_k^{(l)} \right)$$

where N_i denotes the set of direct neighbours of node i, N_i^D is the set of domain neighbours, and W_D and W are trainable weight matrices. To determine the domain neighbors, we compute cosine similarities and match the concept nodes only from source domain to target domain: $cosine(h_s, h_t)$. The values are then normalized into the range of [0,1], and we keep the top 10% of domain neighbors.[2]

DistMult Decoder We optimize the original inner product decoder from VGAE. To predict the link between a concept pair (c_i, c_j), we apply the DistMult (Yang et al., 2015a) method: we take the output node features from the last layer, $\hat{X}$, and define the following score function to recover the adjacency matrix $\hat{A}$ by learning a trainable weight matrix R: $\hat{A} = \hat{X} R \hat{X}$. A Sigmoid function is used to predict positive/negative labels from $\hat{A}$.

4 Evaluation

We evaluate on our new corpus LectureBankCD, treating the NLP domain as the source domain and transferring to the two new target domains: NLP→CV and NLP→BIO. Consistent with Kipf and Welling (2017); Li et al. (2019), we randomly split the positive relations into 85% training, 5% validation, and 10% testing. To account for imbalanced data, we randomly select negative relations such that the training set has the same number of positive and negative relations. We do the same for the validation and test sets. We report average scores over five different randomly seeded splits.

To encode concepts and resources, we test BERT and P2V embeddings. For BERT, we applied a pretrained version from Google[3]. We trained P2V using all the resource data. Both methods only require free-text for training and encoding.

Baseline Models We concatenate the BERT/P2V embeddings of each pair of concepts and feed the result into a classifier (CLS + BERT and CLS + P2V). We train the classifier on the source domain only, then evaluate on the target domain. We report the best performance among Support Vector Machine, Logistic Regression, Gaussian Naïve Bayes, and Random Forest. In addition, we train the VGAE model Li et al. (2019) on the source domain and test on the target domain, initializing the VGAE input with BERT and P2V embeddings separately (VGAE + BERT and VGAE + P2V). Given that GAE is structurally similar to VGAE, we leave this for future work. Other graph-based methods including DeepWalk (Perozzi et al., 2014) and Node2vec (Grover and Leskovec, 2016) are not applicable in this setting as both models require training edges from the target domain in order to generate node embeddings for target concepts.

Proposed Method We report results of our proposed model, CD-VGAE, initialized with BERT and P2V node embeddings separately. Consistent with the work from Li et al. (2019) and Li et al. (2020), P2V embeddings yield better results than BERT embeddings in general. One possible reason for this difference is that BERT embeddings have a large number of dimensions, making it very easy to overfit. The two CLS models yield a negative result, with F1 worse than random guess. A possible reason is that treating concept pairs independently from the source domain may not be beneficial for the target domains. The VGAE models have a better performance when considering the concepts in a large graph. As shown in the table, our method performs better than the chosen baselines on both accuracy and F1 score, by incorporating information from domain neighbors. In particular, it yields much higher recall than all the baseline models. We provide further analysis in a later section.

Upper Bound Performance Finally, we conduct in-domain experiments on CV and BIO (supervised training and testing in the target domain), to show an upper bound for cross-domain performance. We test a variety of methods including traditional classifiers as well as graph-based approaches, including DeepWalk, Node2vec, and GraphSAGE (Hamilton et al., 2017).

5 Analysis

Next, we conduct quantitative analysis and case studies on the target domain concept graphs recovered by our model (CD-VGAE+P2V) and two

[1] In Figure 2, the two labeled nodes are domain neighbors.

[2] Parameter is selected using validation dataset.

[3] `https://github.com/google-research/bert`, (version with L = 12 and H = 768)

	NLP→CV				NLP→BIO			
Method	**F1**	**Acc**	**Pre**	**Rec**	**F1**	**Acc**	**Pre**	**Rec**
Baseline Models								
CLS + BERT	0.4277	0.5480	0.5743	0.3419	0.3930	0.6000	0.7481	0.2727
CLS + P2V	0.4881	0.5757	0.6106	0.4070	0.2222	0.5333	0.6000	0.1364
VGAE + BERT (Li et al., 2019)	0.5885	0.5477	0.5398	0.6488	0.6011	0.6091	0.6185	0.5909
VGAE + P2V (Li et al., 2019)	0.6202	0.5500	0.5368	0.7349	0.6177	0.6273	0.6521	0.6091
Proposed Method								
CD-VGAE + BERT	0.6391	0.5593	0.5441	0.7884	0.6289	0.6273	0.6425	0.6364
CD-VGAE + P2V	**0.6754**	**0.5759**	0.5468	0.8837	**0.6512**	**0.6591**	0.6667	0.6364
Supervised Performance - Upper Bound								
CLS + Node2vec (Grover and Leskovec, 2016)	0.8172	0.8197	0.8223	0.8140	0.8060	0.7956	0.7547	0.8727

Table 2: Evaluation results on two target domains. Underlined scores are the best among the baseline models.

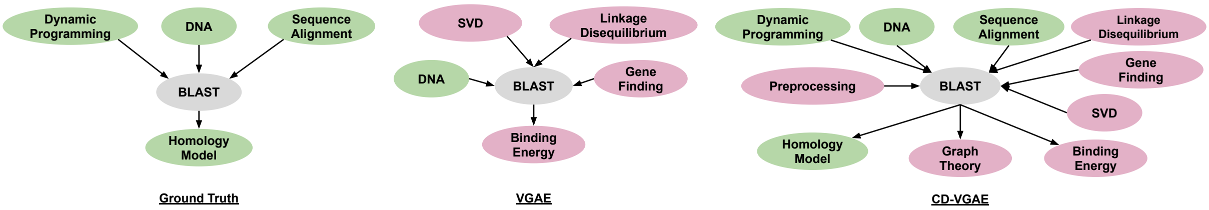

Figure 3: Case Study in BIO: direct neighbors of *BLAST*, including successors and prerequisites, from the ground truth, VGAE, and our proposed CD-VGAE model. SVD stands for Singular Value Decomposition. Correct nodes are marked in blue, incorrect nodes are marked in red. (Best viewed in color!)

baseline models (CLS + P2V, VGAE + P2V), to take a closer look at the results.

Quantitative Analysis We first apply the three trained models to recover the concept graph in the CV domain. Compared to the ground truth with 871 positive relations, the baseline model predicts 527, VGAE predicts 963, and our model predicts 1,209. Similarly, in the BIO domain with 234 positive relations, the baseline model predicts only 128 positive edges, VGAE predicts 261, and our model predicts 303. Since our model tends to predict more positive edges, it has a higher recall. A higher recall is preferred in real-world applications as a system should not miss any relevant concepts when designing a user's study path.

Concept Graph Recovery We now provide case studies of the recovered concept graphs. In Table 3, we show successors of the concept *Image Processing* from the CV domain, i.e. concepts for which *Image Processing* is a prerequisite. Both the baseline model and VGAE miss many successor concepts, whereas our model can recover a correct list without any missing concepts.

We illustrate another case study from the BIO domain in Figure 3 using the concept *BLAST* (short for "basic local alignment search tool "), an algorithm for comparing primary biological sequence information. In the ground truth, BLAST has three prerequisite concepts (*Dynamic Programming*, *DNA* and *Sequence Alignment*), and one successor concept (*Homology Model*). We observe that VGAE predicts only one prerequisite, *DNA*, and misses all the others. In contrast, our model successfully includes all the ground truth relations, although it predicts some extra ones compared to VGAE. A closer look at the extra predictions reveals that these are still relevant topics, even though they are not direct prerequisites. For example, *Sequence Alignment*, *BLAST* and *Graph Theory* are all associated with sequence analysis and share some common algorithms (i.e. De Bruijn Graph).

We provide a case study in the CV domain, shown in Figure 4, by selecting concept node *Object Localization*. The ground truth shows that it has 14 direct neighbors. The VGAE model only predicts five neighbors, while our model predicts more. Our model has two wrong predictions, but it gets 12 correct ones. In contrast, the VGAE model misses up to 10 neighbors, which is not acceptable in an application scenario of an educational platform leading students to miss very useful information.

6 Conclusion

In this paper, we proposed the CD-VGAE model to solve the task of cross-domain prerequisite chain

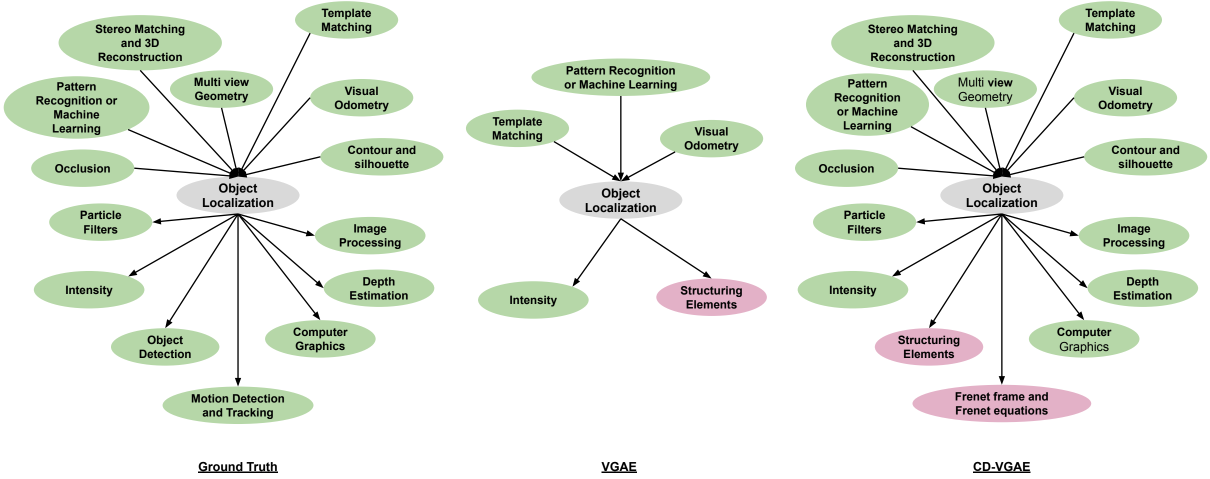

Figure 4: Case Study in CV: direct neighbors of *Object Localization*.

Base	**VGAE**
Image Representation	Image Representation
OCR	Computer graphics
	Eye Tracking
CD-VGAE	**Ground Truth**
Video/Image augmentation	Video/Image augmentation
Image Representation	Image Representation
Face Detection	Face detection
Emotion Recognition	Emotion Recognition
Feature Extraction	Feature Extraction
Feature Learning	Feature Learning
OCR	OCR
Computer Graphics	Computer Graphics
Eye Tracking	Eye Tracking

Table 3: Successors of the concept *Image Processing*, i.e. concepts for which *Image Processing* is a prerequisite (OCR stands for Optical Character Recognition).

learning. Results show that our model outperforms previous unsupervised graph-based models by a large margin, especially with respect to the F1 and recall scores. In addition, we created a new dataset that contains resources and concepts from two domains along with annotated prerequisite relations.

References

Fareedah ALSaad, Assma Boughoula, Chase Geigle, Hari Sundaram, and ChengXiang Zhai. 2018. Mining mooc lecture transcripts to construct concept dependency graphs. *International Educational Data Mining Society*.

Chiara Alzetta, Alessio Miaschi, Giovanni Adorni, Felice Dell'Orletta, Frosina Koceva, Samuele Passalacqua, and Ilaria Torre. 2019. Prerequisite or not prerequisite? that's the problem! an nlp-based approach for concept prerequisite learning. In *CLiC-it*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3632–3642, Brussels, Belgium. Association for Computational Linguistics.

Billy Chiu, Sunil Kumar Sahu, Derek Thomas, Neha Sengupta, and Mohammady Mahdy. 2020. Autoencoding keyword correlation graph for document clustering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3974–3981, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jonathan Gordon, Linhong Zhu, Aram Galstyan, Prem Natarajan, and Gully Burns. 2016. Modeling concept dependencies in a scientific corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–875, Berlin, Germany. Association for Computational Linguistics.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 855–864. ACM.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034.

Thomas N Kipf and Max Welling. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Irene Li, Alexander Fabbri, Swapnil Hingmire, and Dragomir Radev. 2020. R-VGAE: Relational-variational graph autoencoder for unsupervised prerequisite chain learning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1147–1157, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Irene Li, Alexander R. Fabbri, Robert R. Tung, and Dragomir R. Radev. 2019. What should I learn first: Introducing lecturebank for NLP education and prerequisite chain learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6674–6681. AAAI Press.

Chen Liang, Jianbo Ye, Shuting Wang, Bart Pursel, and C. Lee Giles. 2018. Investigating active learning for concept prerequisite learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 7913–7919. AAAI Press.

Mary L McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282.

Liangming Pan, Chengjiang Li, Juanzi Li, and Jie Tang. 2017. Prerequisite relation learning for concepts in MOOCs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1447–1456, Vancouver, Canada. Association for Computational Linguistics.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: online learning of social representations. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 701–710. ACM.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015a. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Yiming Yang, Hanxiao Liu, Jaime G. Carbonell, and Wanli Ma. 2015b. Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM 2015, Shanghai, China, February 2-6, 2015*, pages 159–168. ACM.

Jifan Yu, Gan Luo, Tong Xiao, Qingyang Zhong, Yuquan Wang, Wenzheng Feng, Junyi Luo, Chenyu Wang, Lei Hou, Juanzi Li, Zhiyuan Liu, and Jie Tang. 2020. MOOCCube: A large-scale data repository for NLP applications in MOOCs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3135–3142, Online. Association for Computational Linguistics.

A Supervised Results

Method	Acc	F1	Pre	Rec
GS+BERT	0.7491	0.7513	0.7404	0.7628
GS+P2V	0.7457	0.7423	0.7486	0.7372
CLS+P2V	0.7642	0.757	0.7754	0.7395
CLS+BERT	0.7572	0.7495	0.7677	0.7326
DeepWalk	0.7988	0.791	0.8182	0.7674
Node2vec	**0.8197**	**0.8172**	0.8223	0.8140

Table 4: Supervised evaluation results: CV→CV. GS:GraphSAGE.

Method	Acc	F1	Pre	Rec
GS+BERT	0.7289	0.7355	0.7104	0.7727
GS+P2V	0.7911	0.7904	0.7787	0.8091
CLS+P2V	0.72	0.7367	0.6874	0.8091
CLS+BERT	0.7067	0.7189	0.683	0.7727
DeepWalk	0.7911	0.8079	0.7334	0.9091
Node2vec	**0.7956**	**0.8060**	0.7547	0.8727

Table 5: Supervised evaluation results: BIO→BIO. GS:GraphSAGE.

As a supplementary experiment, we present in-domain results in Table 4, 5: CV→CV and BIO→BIO respectively. While we show in the main paper that CLS + Node2vec yields the best result, which serves as an upper bound on cross-domain performance, we additionally show our experimental results for other supervised methods:

CLS + P2V/BERT We encode concept pairs with P2V/BERT, concatenate the embeddings of both concepts within each possible pair, and then train a binary classifier. We report the best performance among Support Vector Machine, Logistic Regression, Gaussian Naïve Bayes, and Random Forest.

DeepWalk, Node2vec DeepWalk (Perozzi et al., 2014) randomly samples a node and traverses to a neighbor node until it reaches a maximum length, updating the latent representation of each node after each “walk ”to maximize the probability of each node’s neighbors given a node’s representation. Node2Vec (Grover and Leskovec, 2016) improves DeepWalk by providing the additional flexibility of placing weights on random walks. For both methods, we input the training prerequisite relations and obtain concept node embeddings. After generating embeddings for each concept in the target domain, we concatenate the embeddings of both concepts in each concept pair and pass the concatenated representation into a classifier to predict the relation. Again, we report the best performance from the same four classifiers.

GraphSAGE + P2V/BERT GraphSAGE (Hamilton et al., 2017) is an inductive framework to generate node embeddings for unseen data by leveraging existing node features. We first treat it as a node embedding method, as done with DeepWalk and Node2vec. After generating concept node embeddings, we train a classifier to predict concept relations and report in-domain results. In addition, we investigate GraphSAGE for the out-of-domain setting. We assume that, because there are unseen topics when transferring to new domains, such an inductive method like GraphSAGE may fit in our scenario. However, we end up with negative results as the original GraphSAGE may not fit in to this specific application. We leave further investigation for future work.