Yale

Unsupervised Cross-Domain Prerequisite Chain Learning using Variational Graph Autoencoders

CPSC 490 Project by Vanessa Yan, in collaboration with Irene Li and advised by Dragomir Radev

Yale University

LILY Lab

Introduction

Learning prerequisite chains in a graph helps people efficiently acquire knowledge in both known and unknown domains. While existing work on prerequisite chain learning mainly focuses on a single domain, we hypothesize that information from one domain can be leveraged to improve the prediction of prerequisite relations in another domain, as long as the two domains share some common concepts. We propose the task of unsupervised cross-domain prerequisite chain learning and release a new dataset, LectureBankCD. A sample use case for the task would be an expert in the Natural Language Processing (NLP) domain attempting to determine a learning path for concepts in an unfamiliar domain such as Computer Vision (CV). We develop two models that substantially outperform baselines on this task. Our Cross-Domain Variational Graph Autoencoder (CD-VGAE) and Domain-Adversarial Variational Graph Autoencoder (DAVGAE) models learn to effectively transfer prerequisite relations from a source domain to a target domain.

Materials and Methods

Task Formulation: We define a prerequisite relation *p -> q* as the notion that concept p must be learned prior to concept q. We then formulate the cross-domain prerequisite chain learning task as a binary classification problem. Given a source domain and a target domain, there are a number of concept pairs (p,q) in both domains. The label for each pair is 1 if concept p is a prerequisite of concept q and 0 otherwise. We manually annotate every pair of concepts in each of three domains— NLP, CV, and Bioinformatics (BIO). We release the annotations as well as university lecture slide resources (in free text format) for all three domains, in a new dataset called LectureBankCD.

Cross-Domain Concept-Resource Graph: For the CD-VGAE model, we build a graph G_{cr}=(X,A) that includes resource nodes as well as concept nodes from both the source and target domains. The adjacency matrix A consists of four edge types: $A^{c,s}$: edges between source concept nodes; $A^{r,c}$: 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 an unsupervised setting, $A^{c,s}$ are known, and the task is to predict $A^{c,t}$. For A^{rc} and A^{r}, we calculate cosine similarities based on BERT or Phrase2Vec (P2V) node embeddings.

Cross-Domain Concept Graph: For the DAVGAE model, we build a graph using concept nodes only. This graph contains a maximum of 523 nodes, whereas the concept-resource graph contains a maximum of 3281 nodes. Building a model on the concept graph is ~6 times more scalable in terms of space complexity.

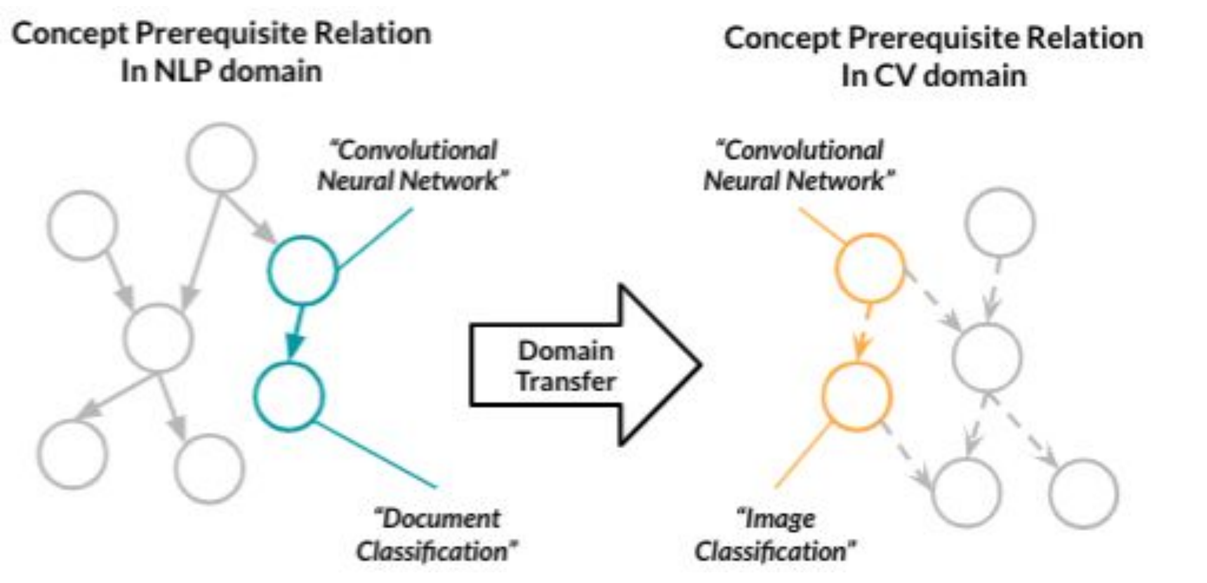
Concept Prerequisite Relation
In NLP domain
"Convolutional Neural Network"
"Document Classification"
Domain Transfer
Concept Prerequisite Relation
In CV domain
"Convolutional Neural Network"
"Image Classification"

Figure 1. An illustration of the cross-domain prerequisite chain learning task. Solid edges in the diagram on the left represent labeled prerequisite relations between concepts in the source domain. Dotted edges in the diagram on the right represent prerequisite relations in the target domain that our models seek to predict. Both domains share common concepts such as *Convolutional Neural Network*. Knowing that *Convolutional Neural Network -> Document Classification* in the source domain helps our models determine that *Convolutional Neural Network -> Image Classification* in the target domain.

Cross-Domain Concept-Resource Graph

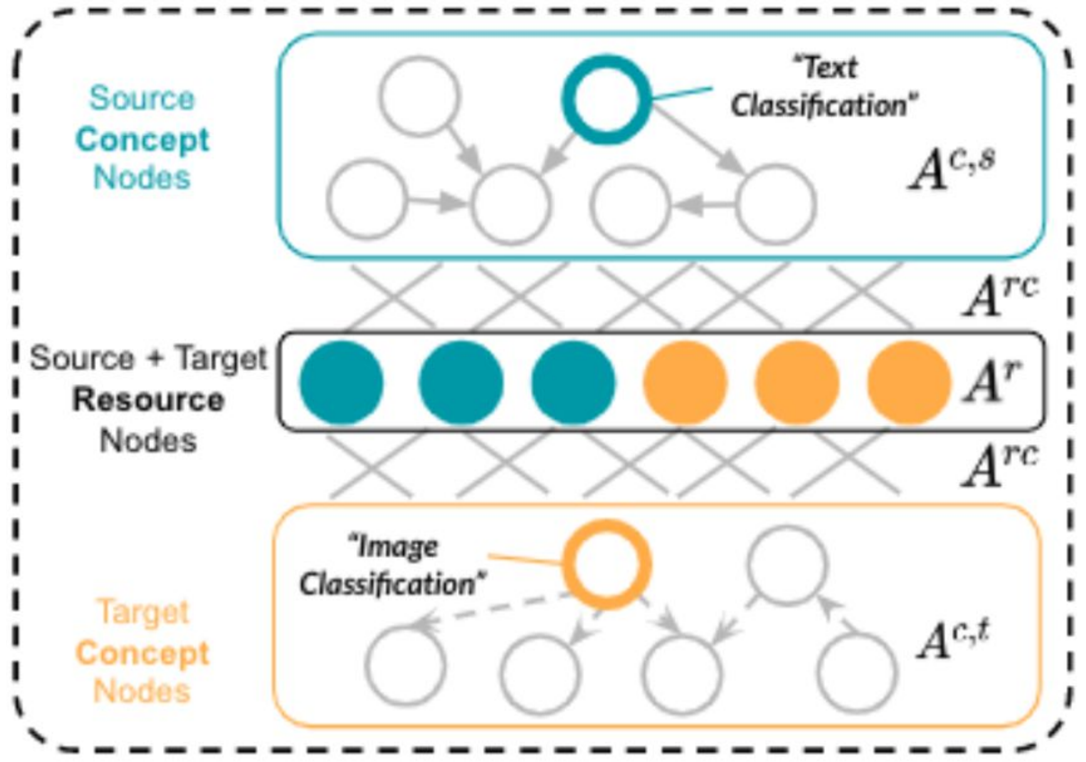

Figure 2. A heterogeneous Cross-Domain Concept-Resource Graph built for the CD-VGAE model.

CD-VGAE Model: CD-VGAE is based on the VGAE model from (Kipf and Welling, 2016). VGAE consists of a graph convolutional network (GCN) encoder and an inner product decoder. The loss of a VGAE is as follows: $L_{vgae} = E_{q(Z|X,A)}[\log p(A|Z)] - KL[q(Z|X,A) \,||\, p(Z)]$, where the first term represents a reconstruction loss and the second term represents the KL divergence between the hidden layer representation Z and a normal distribution.

In a GCN, the hidden representation of a node in the next layer is computed using only the information of direct neighbors and the node itself. We adapt the GCN to additionally consider the domain neighbors for each node. These domain neighbors are a set of common or semantically similar concepts from the other domain, such as those labeled in Figure 1. We determine domain neighbors by computing cosine similarities between concepts in the source domain and those in the target domain. The hidden layer representation Z is now calculated using

$h_i^{(l+1)} = \sigma(\Sigma_{j \in N\_i} W^{(l)} h_j^{(l)} + W^{(l)} h_i^{(l)} + \Sigma_{k \in N\_i\text{^}D} W_D^{(l)} h_k^{(l)})$.

In addition, we change the inner product decoder in VGAE to a DistMult decoder (Yang et al, 2015). Specifically, we take the output features from the last layer and recover the adjacency matrix Â by learning a trainable weight matrix R: $\hat{A} = \breve{X}^{T} R \breve{X}$. Having recovered the matrix A, we use a sigmoid function to predict positive or negative labels.

DAVGAE Model: We add a domain discriminator module to the graph encoder so that it can learn domain-invariant features of the concept nodes in both the source and target domains. The module predicts which domain each node in the hidden layer representation belongs to, using a two-layer neural network. The output of this module is 1 if the node comes from the source domain and 0 otherwise. Overall the total loss of the DAVGAE is $L = L_{vgae} + L_{dis}$, where L_{dis} is a cross-entropy loss for domain prediction.

Baselines: In addition, we develop baseline models using machine learning classifiers as well as graph-based methods. For example, we adapt GraphSAGE (Hamilton et al, 2017) for cross-domain prerequisite chain learning.

To initialize GraphSAGE, we provide the BERT/P2V embeddings of the source and target domain concepts, in addition to an adjacency matrix. The adjacency matrix is constructed from the annotations of the source domain prerequisite relations as well as the cosine similarities of BERT/P2V embeddings of target domain concepts. The concept node embeddings generated by GraphSAGE are then passed into the DistMult decoder.

Table 1. Condensed results (see writeup for full results)

Method	NLP->CV			NLP->BIO		
Baselines	F1	Precision	Recall	F1	Precision	Recall
CLS	0.4881	0.6106	0.4070	0.3930	0.7481	0.2727
GraphSAGE	0.5085	0.5515	0.5342	0.5177	0.5287	0.5283
VGAE	0.6202	0.5368	0.7349	0.6177	0.6521	0.6091
Our models						
CD-VGAE	0.6754	0.5468	0.8837	0.6512	0.6667	0.6364
DAVGAE	0.6321	0.5661	0.7195	0.6421	0.5932	0.7130
Pre-trained DAVGAE	0.6771	0.5734	0.8322	0.6738	0.6559	0.6957

Results

Our CD-VGAE and DAVGAE models both outperform baseline models by a large margin. In particular, our models yield higher recall. This means we have a higher chance of recovering all the concepts required in a learning path, though we may predict some extra concepts compared to the ground truth. On principle, higher recall is preferred in our application setting because we would rather students learn additional concepts than miss key concepts.

With pre-training, DAVGAE improves in performance and outperforms CD-VGAE. Furthermore, DAVGAE saves space complexity by six times compared to CD-VGAE because DAVGAE does not necessitate resource nodes in the graph.

Conclusion

We propose the cross-domain prerequisite learning task, a new dataset LectureBankCD, as well as two novel models that outperform previous machine learning and graph-based models. Our work has been published in ACL 2021.