### Unsupervised Graph Embedding via Adaptive Graph Learning

Rui Zhang;Yunxing Zhang;Chengjun Lu;Xuelong Li

IEEE Transactions on Pattern Analysis and Machine Intelligence

Year: 2022 | Early Access Article | Publisher: IEEE

Link: <https://arxiv.org/pdf/2003.04508.pdf>

Hi my old friends, how are you? Long time no see. Let’s do it again!

We have a graph, with nodes



And their features

.

Of course we have the the node-to-node adjacency matrix



We want to learn embeddings for the nodes,

= [z1, …zn]

 For this purpose we use a deep learning model with many layers. For the l-th layer, it is a convolutional graph layer







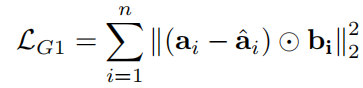
Then after many layers we have the final embedding vectors of the nodes in a embeding matrix Z.



We will use them to reconstruct the adjacency matrix



But how is the reconstruction error? We calculate the L2-norm distance



  0.\



Meanwhile we also want to say if two nodes have a large adjacency in a\_ij, then their embeddings should be close to each other, so we minimize

Min

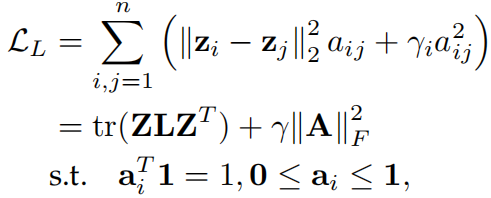
we also hope the adjacency is smooth, so we minimize the squares of the adjacency matrix

Min 

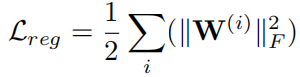
Put together we have a loss function about the adjacency







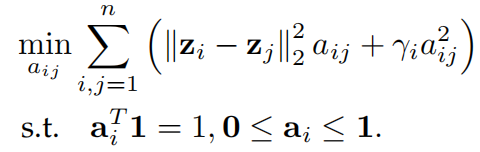


Combining the above two terms with a regularization term, we have the final loss function

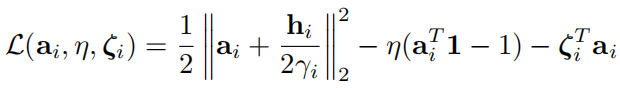


Then how to solve this problem? Again alternate

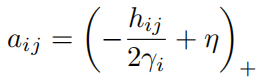
* Fix embedding vectors in Z and only optimize A



With , this is a constrained minimization problem, and its Lagrange equation is



With Karush-Kuhn-Tucker (KKT) conditions, the solution is



So we have a new adjacency matrix, . but the author are actually not really confident with this new matrix, so they again make a trade-off between the new matrix and the original one, 



* Fix A and solve Z. This step is simply a back propagation with a Adam algorithm to update W.

Inspiration:

1. When we talk about graph embedding we always think the graph is given as it is, but actually the embeddings can also help to refine the graph
2. We can also merge some other tasks to the same model, for example, embedding+graph refine+node classification/link prediction
3. Using some more advanced models for embedding, for example, attention network.