Formulation of a Research Problem

By: Neel Haria
Prof. Hang Liu

Link Prediction Based on Graph Neural Networks

Statement of the Problem

Link prediction is a key problem for network-structured data. Link prediction heuristics use some score functions, such as common neighbors and Katz index, to measure the likelihood of links. They have obtained wide practical uses due to their simplicity, interpretability, and for some of them, scalability. However, every heuristic has a strong assumption on when two nodes are likely to link, which limits their effectiveness on networks where these assumptions fail. In this regard, a more reasonable way should be learning a suitable heuristic from a given network instead of using predefined ones. By extracting a local subgraph around each target link, we aim to learn a function mapping the subgraph patterns to link existence, thus automatically learning a "heuristic" that suits the current network.

Introduction

Link prediction is to predict whether two nodes in a network are likely to have a link. Given the ubiquitous existence of networks, it has many applications such as friend recommendation, movie recommendation, knowledge graph completion, and metabolic network reconstruction. One class of simple yet effective approaches for link prediction is called heuristic methods. Heuristic methods compute some heuristic node similarity scores as the likelihood of links. Existing heuristics can be categorized based on the maximum hop of neighbors needed to calculate the score. For example, common neighbors (CN) and preferential attachment (PA) are first-order heuristics, since they only involve the one-hop neighbors of two target nodes. Adamic-Adar (AA) and resource allocation (RA) are second-order heuristics, as they are calculated from up to two-hop neighborhood of the target nodes. We define h-order heuristics to be those heuristics that require knowing up to the h-hop neighborhood of the target nodes. There are also some high-order heuristics that require knowing the entire network. Examples include Katz, rooted PageRank (PR) [9], and SimRank (SR).

Project Timeline

The Project will be implemented over a time period of 12 weeks. I plan to put in around 10-12 hours each week to work on the same. Since this is a vast topic, the implementation will be focused more on research and learning more about the subject. I will try to implement the heuristic methods and obtain comparable results.

Week 1: Literature Review – Read papers related to our problem and analyze the pros and cons of various approaches to a similar problem for the first two weeks.

Week 3: Gathering Dataset – Analyze and gather datasets from various sources, to train the model and see the difference between different methods.

Week 5: Experimental Implementation – Try to implement heuristic methods and see performance on different datasets.

Week 7: Experimental Implementation II – Continue with implementation of heuristic methods before moving to SEAL implementation

Will keep updating the timeline as weeks progress

Expected Results

Learning link prediction heuristics automatically is a new field. In this project, I aim to learn about a novel link prediction framework, SEAL, to simultaneously learn from local enclosing subgraphs, embeddings and, attributes based on graph neural networks. Experimentally I aim to see that SEAL will achieve unprecedentedly strong performance in comparison to various heuristics, latent feature methods, and network embedding algorithms. I hope SEAL can not only inspire link prediction research but also open up new directions for other relational machine learning problems such as knowledge graph completion and recommender systems.