Graph shortest-path distance estimation

This document summarizes a research problems of interest after the discussions and readings.

Motivation: This is a classic problem with ongoing studies being published in top venues (including *Science*). "Answering point-to-point distance queries in graphs is a fundamental building block for many applications in social networks, search, computational biology, computer networks, and road networks."

Formulation:

Definition 1 (Graph shortest-path distance (GSD) problem) *Let $G = \langle V, E \rangle$ be a graph, where V is a set of vertices and E is a set of edges. Let $d_G(u, v)$ denote the shortest-path distance of two vertices u and v $(u, v \in V)$. The graph shortest-path distance problem returns $d_G(u, v)$ given any two vertices u and v $(u, v \in V)$.*

Solutions:

1. [Baseline 1?] We learn an MLP (model structure?) f that maps two vertices u and v to a real value d, such that $|f(u, v) - d_G(u, v)|$ is minimized.
2. [Baseline 2?] We learn an auto-encoder (what kind of AE? model structure?) f that maps a vertex v to a vector $\mathtt{V}$, such that $|f(v) \cdot f(u) - d_G(u, v)|$ is minimized.
3. [Proposed?] We learn an auto-encoder (what kind of AE? model structure?) f that maps a vertex v to a vector $\mathtt{V}$, such that $|d_E(f(v), f(u)) - d_G(u, v)|$ is minimized. Here, $d_E()$ is a function that returns the Euclidean distance between two vectors.

Research Challenges (Contributions):

1. The entire matrix of $d_G(u, v)$ values is too large to be stored (otherwise we can simply store the matrix and look it up at query time). We need to design an order of assessing the mini batches, such that the mini batches used in close rounds can share the computation for the $d_G(u, v)$ values.

   Solution: ? Set up some sort of traversal order?

2. We need to derive a bound on the error of the predicted distance value $\hat{d}_G(u, v)$.

   Solution:

   - Empirical: We can test the learned model on all vertex pairs to compute the maximum error.

- Theoretical: We can use LSH to approximate an $n \times n$ grid and derive a bound on the distance error? German Tank problem? Explore kernel matrix (kernel LSH)?

  Count min sketch (preserve inner product by XX error). Add constraint for LSH into optimization function.

3. We can use the model built to verify the existence of triangular inequality, or integrate this property into the model optimization function. This helps improve the model interpretability.

   Solution: Look up Google Tensorflow update for adding domain specific constraints.

4. Matrix value is non-continuous. How to optimize the model?