"Hierarchical graph embedding in vector space by graph pyramid," S. Mousav, et al.

This work was supported in part by the Center for Brain Inspired Computing (C-BRIC), one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.
Research Goal

- Cognition involves accessing and performing inference over complex, structured knowledge
  - Capture knowledge, represent it, perform inference over it
  - Graphs are a natural representation of such sparse information

- Research goal: explore the mapping of graphs to basic cortical like arrays
  - Start with a simplified cortical model - incrementally add cortical characteristics
  - Sparse, distributed data representations
  - Competing objective functions:
    - Increased mapping quality amb performance on standard graph queries (here label inference)
    - Connectivity amb number and length of connections

- Most real-world graphs (semantic, knowledge, social networks, ...) have small world/scale free characteristics
  - A scale-free network is a network whose degree distribution follows a power law
  - In a small-world network the typical distance between two randomly chosen nodes grows according to log(N)
  - Has been observed in neural circuits
What role does cortical interconnect architecture play in cortical functionality?

Connections are expensive in biology

Cortex has, most likely found the “minimum” connectivity - number and length of connections - required by cortex to do its job


- Neocortex has a similar layered architecture in species over a wide range of brain sizes
- Larger brains => longer fibers to communicate between distant cortical areas
- White matter volume increases disproportionally (4/3 power) faster than gray matter volume
- Power law accounts for empirical data spanning several orders of magnitude in brain sizes for various mammalian species

\[ W = \frac{c}{T} G^{4/3} \]
Current Approaches (Deep Neural Networks)

- Deep Network architectures for “graph embedding” Graph Neural networks (GNNs): Node2Vec, Graph2Vec, TransE, RecGNNs, Graph Convolutional Neural Networks

- Gradient descent results in broad data distributions
  - Representing sparse graphical relationships can be less efficient

- Other disadvantages: long training times, scalability, the need to retrain when there is new information

- Biological inspired algorithms address some of these disadvantages
  - Numenta’s “complementary sparseness”: sparse interconnect and sparse activation

"Representation Learning on Graphs: Methods and Applications", W. Hamilton, et al., Department of Computer Science Stanford University.
Cognitive Graph (COG)

- A source graph is read into the system and an internal data structure, COG, is created.

- Graph data sets used (labeled directed graphs)
  - **adjnoun** (A network of word adjacencies of common adjectives and nouns in the novel "David Copperfield" by Charles Dickens)
  - **webkb-wisc** (A dataset that includes web pages from computer science departments of various universities)
  - **CORA** (A dataset of scientific publication citations classified into one of seven classes)
  - **Human Protein Interaction** (A protein-protein interaction network for Homo Sapiens, where nodes represent proteins, and edges indicate the biological interaction between a pair of proteins)

CORA
https://graphsandnetworks.com/the-cora-dataset/
Cortical Graph (COR)

- The COR is constructed via a random walk through the COG (rapid learning)
  - Random walks (from Node2Vec*) with a weighted balance DFS (depth first search) and BFS (breadth first search)
- The cortical network consists of a 2D XY grid of “columns”
- Each column has some number of minicolumns (COR nodes)
  - During the random walk k minicolumns (k-SDR) recruited for each COG node (constrained random)
  - Axons are recruited randomly: a minicolumn can connect, on average, to $P_a$ (axon connection density) other minicolumns (we use 1-5%)
  - Single layer columns, single level XY grid
- The COR representations of the COG nodes constitute a very sparse “graph embedding”
- Synaptic (edge) weights for the COR nodes are determined by simple Hebbian learning during the random walk
  - Are labels correlated or anti-correlated (can be + or -)
- The simulator allows for a range of COR allocation and interconnect techniques
  - SISO – Similar Input Similar Output (Representation)

*node2vec: Scalable Feature Learning for Networks. A. Grover, J. Leskovec, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016.
COR Node Allocation
Graph Query: Label Inference

- COR nodes in each COG node’s COR k-SDR share the same label

- For label inference, 50% of the nodes in each graph have their labels removed
  - These are then inferred from the remaining labels

- Label likelihoods (“beliefs”) are propagated through the network
  - Likelihood propagation via message passing
  - Nodes accumulate label log likelihoods from neighbor nodes
  - COR: subsets of identically labeled nodes form cell assemblies, reinforcing each other “resonance”

- A standard label inference (node classification) algorithm
  - Node Classification in Social Networks, S. Bhagat et al.
  - No “high order information” (homophily) factored into process
## Preliminary Results

<table>
<thead>
<tr>
<th>Data Set</th>
<th>adjnoun</th>
<th>webkb</th>
<th>CORA</th>
<th>Human Protein (PPI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cognitive graph (nodes/edge/labels)</td>
<td>112/425/2</td>
<td>265/822/5</td>
<td>3264/4536/7</td>
<td>3890/76584/50</td>
</tr>
<tr>
<td>Cortical graph (nodes/edge)</td>
<td>1024/1349</td>
<td>16384/1687</td>
<td>16384/3456</td>
<td>262K/49K</td>
</tr>
<tr>
<td>Cognitive graph, Macro-F</td>
<td>99%</td>
<td>77%</td>
<td>90%</td>
<td>10%</td>
</tr>
<tr>
<td>Node2Vec, Macro-F</td>
<td>83%</td>
<td>63%</td>
<td>75%</td>
<td>18%</td>
</tr>
<tr>
<td>Cortical graph, Macro-F</td>
<td>95%</td>
<td>88%</td>
<td>86%</td>
<td>83%</td>
</tr>
<tr>
<td>Cortical graph. Connectivity</td>
<td>5%</td>
<td>1%</td>
<td>5%</td>
<td>1%</td>
</tr>
<tr>
<td>Connection Effectiveness</td>
<td>0.49</td>
<td>0.84</td>
<td>0.31</td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td>0.33</td>
<td>0.38</td>
<td>1.43</td>
<td></td>
</tr>
</tbody>
</table>

- adjnoun k=4, all others k=8
- Connection effectiveness, which is only computed for the Cortical graph, is the network performance (Macro-F1 score) divided by the mean connections per node times the mean connection length.

Example of cell assembly resonance
Next Steps

- COR architecture:
  - More “physical” diffusion model of label belief
  - Numenta-like dendritic segments that capture “higher order” information (homophily)
  - Multi-level hierarchies that capture more graph structure
  - Belief propagation by spikes
  - More complex multi-layered columns
  - Analysis to drive architecture
  - Investigate functional implications of Small World and Scale Free networks

- Expand the tasks, a wider variety of graph queries
  - Label inference (node classification, NC), given a partly labeled graph, infer or classify the unlabeled nodes
  - Edge inference (link prediction or graph completion), remove edges from a learned graph, infer missing edges
  - PageRank (PR) Used extensively in graph search