Random Edge Coding:
One-Shot Bits-Back Coding of Large Labeled Graphs


Daniel Severo


James Townsend


Ashish Khisti


Alireza Makhzani

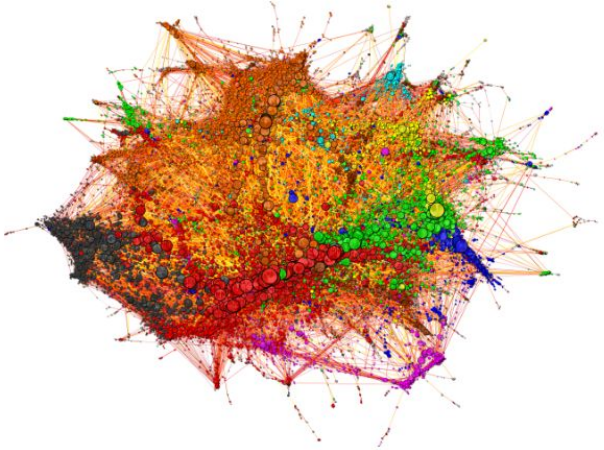
UNIVERSITY OF TORONTO

VECTOR INSTITUTE

UNIVERSITY OF AMSTERDAM

arXiv:2305.09705

Objective: Fast lossless compression of large networks

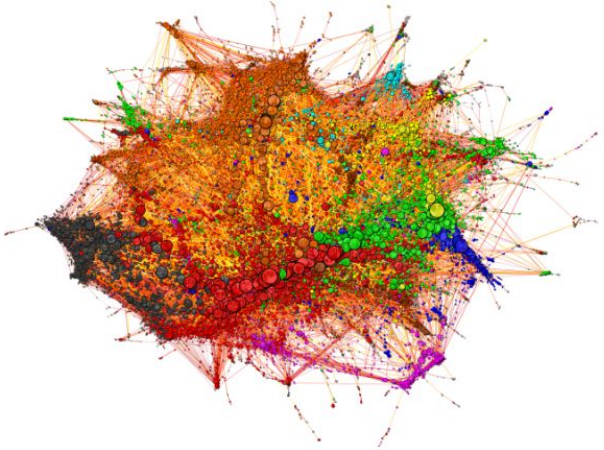


Encode

Decode

01001010010000100

Objective: Fast lossless compression of large networks



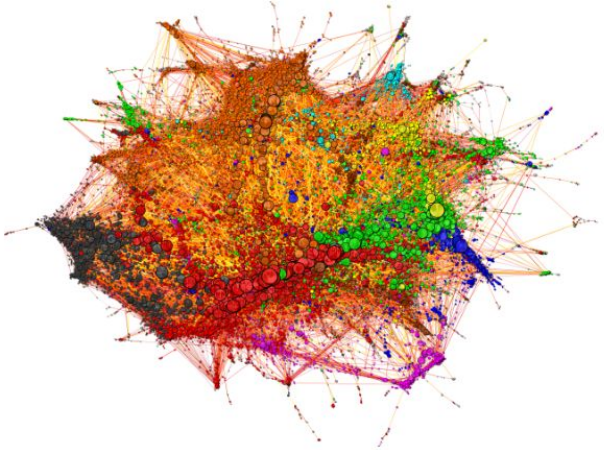
millions of nodes and edges

Encode

Decode

01001010010000100

Objective: Fast lossless compression of large networks



millions of nodes and edges

sparse

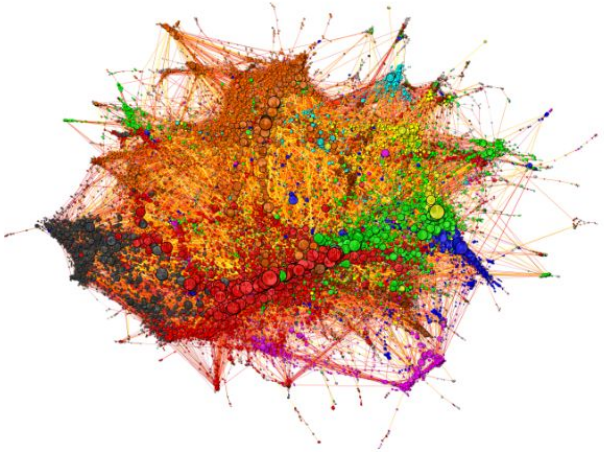
$$|E| \ll |V|^2$$

Encode

Decode

0100101001000010 0

Objective: Fast lossless compression of large networks



millions of nodes and edges

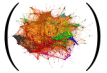
sparse

$$|E| \ll |V|^2$$

Encode

Decode

$$0100101001000010 0$$

$$\log_2 1/\Pr(\text{  })$$

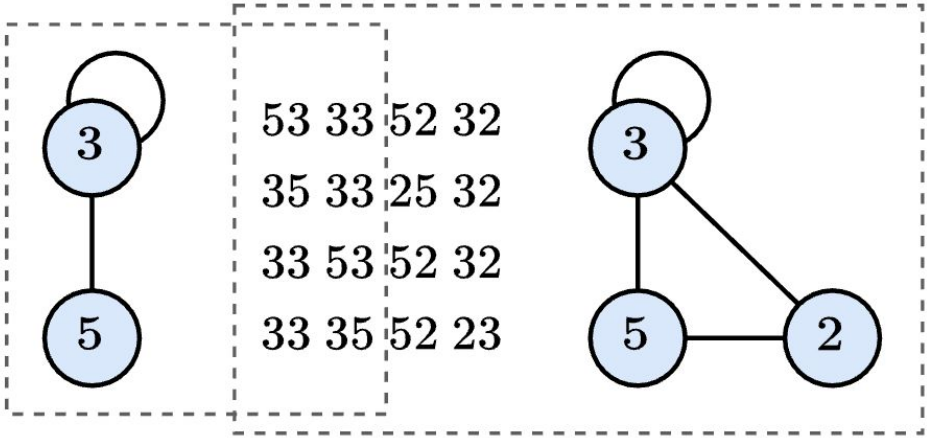
We want to losslessly compress a (large) labeled graph G at the one-shot rate

$$\log 1/\Pr(G)$$

We want to losslessly compress a (large) labeled graph G at the one-shot rate

$$\log 1/\Pr(G)$$

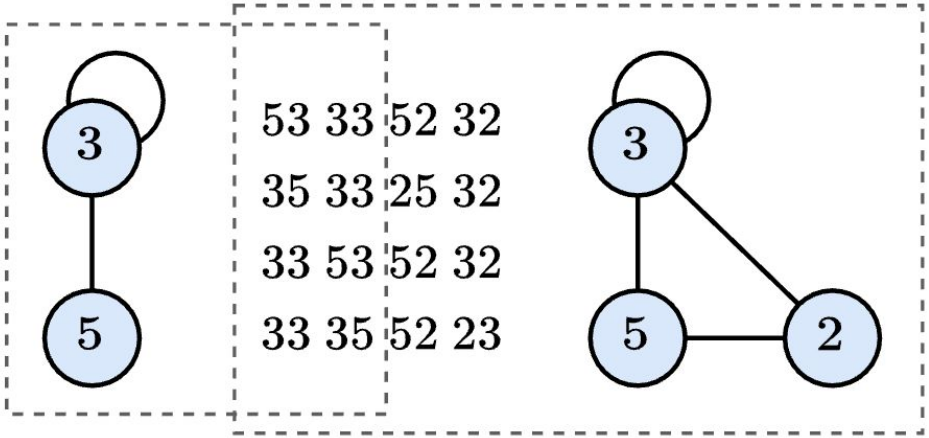
Graphs can be represented as sequences of vertices



We want to losslessly compress a (large) labeled graph G at the one-shot rate

$$\log 1/\Pr(G)$$

Graphs can be represented as sequences of vertices



Equivalent sequences $\mathbf{v} \sim \mathbf{w}$ map to the same graph

$$53\ 33 \sim 33\ 53 \sim 33\ 35 \sim 35\ 33$$

Compressing a single sequence uses too many bits

$$\log 1/\Pr(G) = \log 1/\Pr(\mathbf{v}) - \underbrace{\log(\#\text{ of equivalent sequences})}_{\text{excess bits}}$$

Compressing a single sequence uses too many bits

$$\log 1/\Pr(G) = \log 1/\Pr(\mathbf{v}) - \underbrace{\log(\#\text{ of equivalent sequences})}_{\text{excess bits}}$$

Given $\mathbf{v}$, how many equivalent sequences are there?

Compressing a single sequence uses too many bits

$$\log 1/\Pr(G) = \log 1/\Pr(\mathbf{v}) - \underbrace{\log(\#\text{ of equivalent sequences})}_{\text{excess bits}}$$

Given $\mathbf{v}$, how many equivalent sequences are there? We can,

- Permute edges $\rightarrow |E|!$

Compressing a single sequence uses too many bits

$$\log 1/\Pr(G) = \log 1/\Pr(\mathbf{v}) - \underbrace{\log(\#\text{ of equivalent sequences})}_{\text{excess bits}}$$

Given $\mathbf{v}$, how many equivalent sequences are there? We can,

- Permute edges $\rightarrow |E|!$
- Permute vertices within an edge $\rightarrow 2^{|E|}$

Compressing a single sequence uses too many bits

$$\log 1/\Pr(G) = \log 1/\Pr(\mathbf{v}) - \underbrace{\log(\#\text{ of equivalent sequences})}_{\text{excess bits}}$$

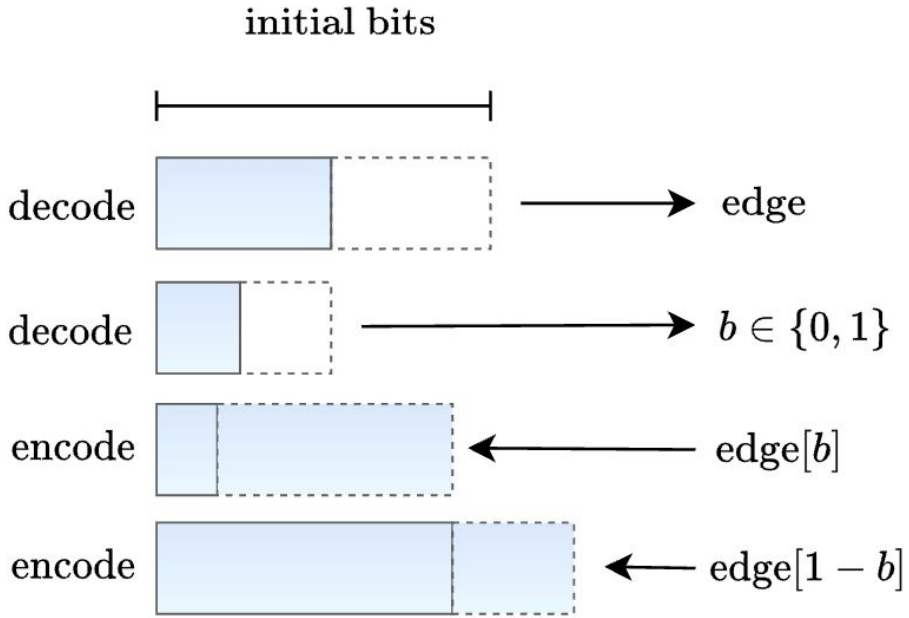
Given $\mathbf{v}$, how many equivalent sequences are there? We can,

- Permute edges $\rightarrow |E|!$
- Permute vertices within an edge $\rightarrow 2^{|E|}$

Excess bits $= \log(|E|!) + |E| =$ the ordering between edges

Random Edge Encoder

$\mathcal{O}(|E|)$ memory
$\mathcal{O}(|E| \log |E|)$ compute



	Social Networks			
	YouTube	FourSq.	Digg	Gowalla
# Nodes	3,223,585	639,014	770,799	196,591
# Edges	9,375,374	3,214,986	5,907,132	950,327
$10^6 \times$Density	1.8	15.75	19.82	50.2
(ours) FHW w/ BB-EXG	**15.01**	9.82	10.49	12.18
Pool Comp.	15.38	**9.23**	11.59	**11.73**
Slashburn	17.03	10.67	**9.82**	11.83
Backlinks	17.98	11.69	12.56	15.56
List Merging	15.80	9.95	11.92	14.88

Random Edge Coding:
One-Shot Bits-Back Coding of Large Labeled Graphs


Daniel Severo


James Townsend


Ashish Khisti


Alireza Makhzani

UNIVERSITY OF TORONTO

VECTOR INSTITUTE

UNIVERSITY OF AMSTERDAM

arXiv:2305.09705