Universal lossless compression of graph-indexed data

Venkat Anantharam

EECS Department
University of California, Berkeley

June 2, 2017

Information Modelling and Control of Complex Systems Workshop
Ohio State University
(Joint work with Payam Delgosha)
Outline

1. Universal data compression
2. Graph-indexed (graphical) data
3. The framework of local weak convergence
4. Universal compression of graphical data
Payam Delgosha
Outline

1. Universal data compression
2. Graph-indexed (graphical) data
3. The framework of local weak convergence
4. Universal compression of graphical data
The data compression problem

- $X_i$: an i.i.d. sequence of random variables taking values in $\mathcal{X}$
The data compression problem

- $X_i$: an i.i.d. sequence of random variables taking values in $\mathcal{X}$
- Lossless block compression involves a source encoder and decoder:

$$f_n : \mathcal{X}^n \rightarrow \{0,1\}^m \quad \text{and} \quad g_n : \{0,1\}^m \rightarrow \mathcal{X}^n$$
The data compression problem

- $X_i$: an i.i.d. sequence of random variables taking values in $\mathcal{X}$
- Lossless block compression involves a source encoder and decoder:
  
  $$ f_n : \mathcal{X}^n \to \{0,1\}^m \quad \text{and} \quad g_n : \{0,1\}^m \to \mathcal{X}^n $$

- Compression rate is $m/n$. 
The data compression problem

- $X_i$: an i.i.d. sequence of random variables taking values in $\mathcal{X}$
- Lossless block compression involves a source encoder and decoder:
  \[ f_n : \mathcal{X}^n \rightarrow \{0,1\}^m \quad \text{and} \quad g_n : \{0,1\}^m \rightarrow \mathcal{X}^n \]
- Compression rate is $m/n$.
- A rate $R$ is achievable if we can find a sequence of encoders/decoders such that $m/n \rightarrow R$ and also
  \[ \mathbb{P} \left( g_n(f_n(X_1, \ldots, X_n)) \neq (X_1, \ldots, X_n) \right) \rightarrow 0. \]
The data compression problem

- $X_i$: an i.i.d. sequence of random variables taking values in $\mathcal{X}$
- Lossless block compression involves a source encoder and decoder:
  \[
  f_n : \mathcal{X}^n \to \{0, 1\}^m \quad \text{and} \quad g_n : \{0, 1\}^m \to \mathcal{X}^n
  \]
- Compression rate is $m/n$.
- A rate $R$ is achievable if we can find a sequence of encoders/decoders such that $m/n \to R$ and also
  \[
  \mathbb{P}(g_n(f_n(X_1, \ldots, X_n)) \neq (X_1, \ldots, X_n)) \to 0.
  \]
- For stationary sources, the information theoretic limit is the entropy rate of the source, i.e. $R$ is achievable if $R > H(X)$, not achievable if $R < H(X)$, where
  \[
  H(X) := \lim_{n \to \infty} \frac{1}{n} H(X_1, \ldots, X_n).
  \]
Huffman Coding
Huffman Coding
Huffman Coding

\[ \begin{array}{c}
\text{.4} \\
\text{.35} \\
\text{.2} \\
\text{.05} \\
\text{.25} \\
\text{.6} \\
\end{array} \]
Huffman Coding
Huffman Coding
Optimality of Huffman Coding

\[ H(X_1) \leq \mathbb{E} \text{[code length]} \leq H(X_1) + 1 \]
Optimality of Huffman Coding

- \( H(X_1) \leq \mathbb{E} \text{ [code length]} \leq H(X_1) + 1 \)
- Treating the whole block \( X_1, \ldots, X_n \) as a single source,

\[
\frac{1}{n} L_n \leq \frac{1}{n} \left( H(X_1, \ldots, X_n) + 1 \right),
\]

hence achieving the entropy rate of the process.
Typical Sequences

- Given a source distribution $p_{X}$ on alphabet $\mathcal{X}$, a sequence $x^n = (x_1, \ldots, x_n)$ in $\mathcal{X}^n$ is called $\epsilon$-typical if

$$|\phi_{x_0}(x^n) - p_X(x_0)| \leq \epsilon \quad \forall x_0 \in \mathcal{X}$$

where $\phi_{x_0}(x^n)$ is the fraction of indices with value $x_0$. 
Typical Sequences

- Given a source distribution $p_X$ on alphabet $\mathcal{X}$, a sequence $x^n = (x_1, \ldots, x_n)$ in $\mathcal{X}^n$ is called $\epsilon$-typical if

  $$|\phi_{x_0}(x^n) - p_X(x_0)| \leq \epsilon \quad \forall x_0 \in \mathcal{X}$$

  where $\phi_{x_0}(x^n)$ is the fraction of indices with value $x_0$.

- If $\mathcal{T}_\epsilon^n$ denotes the set of $\epsilon$-typical sequences of length $n$ and $X^n$ is i.i.d. with distribution $p_X$,

  $$|\mathcal{T}_\epsilon^n| \leq 2^{n(H(X)+\delta(\epsilon))}$$

  and

  $$\mathbb{P}(X^n \in \mathcal{T}_\epsilon^n) \to 1$$
Typical Sequences

- Given a source distribution $p_X$ on alphabet $\mathcal{X}$, a sequence $x^n = (x_1, \ldots, x_n)$ in $\mathcal{X}^n$ is called $\epsilon$-typical if
  $$|\phi_{x_0}(x^n) - p_X(x_0)| \leq \epsilon \quad \forall x_0 \in \mathcal{X}$$

  where $\phi_{x_0}(x^n)$ is the fraction of indices with value $x_0$.

- If $\mathcal{T}_\epsilon^n$ denotes the set of $\epsilon$-typical sequences of length $n$ and $X^n$ is i.i.d. with distribution $p_X$,
  $$|\mathcal{T}_\epsilon^n| \leq 2^{n(H(X)+\delta(\epsilon))} \quad \text{and} \quad \mathbb{P}(X^n \in \mathcal{T}_\epsilon^n) \to 1$$

- Another optimal encoding: knowing the source distribution, focus on the typical sequences and express a sequence via its index among typical sequences.
Universal data compression

- In practice, it might be the case that we do not know the statistics of the source.
Universal data compression

- In practice, it might be the case that we do not know the statistics of the source.
- Assume a sequence $X_1, X_2, \ldots$ is given to us which is generated from an ergodic stationary stochastic process, but we do not know the statistics of the source.
Universal data compression

- In practice, it might be the case that we do not know the statistics of the source.
- Assume a sequence $X_1, X_2, \ldots$ is given to us which is generated from an ergodic stationary stochastic process, but we do not know the statistics of the source.
- Upon receiving the first $n$ symbols, we convert it to a sequence of zeros and ones with length $l_n$. 
Universal data compression

- In practice, it might be the case that we do not know the statistics of the source.
- Assume a sequence $X_1, X_2, \ldots$ is given to us which is generated from an ergodic stationary stochastic process, but we do not know the statistics of the source.
- Upon receiving the first $n$ symbols, we convert it to a sequence of zeros and ones with length $l_n$.
- This is optimal when with probability one

$$\limsup_n \frac{1}{n} l_n \leq \text{entropy rate}$$
Universal data compression

- In practice, it might be the case that we do not know the statistics of the source.
- Assume a sequence $X_1, X_2, \ldots$ is given to us which is generated from an ergodic stationary stochastic process, but we do not know the statistics of the source.
- Upon receiving the first $n$ symbols, we convert it to a sequence of zeros and ones with length $l_n$.
- This is optimal when with probability one

$$\limsup \frac{1}{n} l_n \leq \text{entropy rate}$$

- By optimal universal compression we mean achieving the entropy rate without knowing the statistics of the source.
<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
</table>

Lempel Ziv
Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Venkat Anantharam

Large networks

June 2, 2017 10 / 35
Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>dictionary</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>output</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
</table>
Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| output |
Lempel Ziv

input 0 1 1 0 0 1 0 1 1

dictionary

1 2 3 4 5 6 7 8

0 1 01

output 1
Lempel Ziv

input: 0 1 1 0 0 1 0 1 1

dictionary:

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

output: 1
Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>01</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

output: 1,2
Lempel Ziv

input

0 1 1 0 0 1 0 1 1

dictionary

1 2 3 4 5 6 7 8

0 1 01 11

output

1 ,2
### Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>01</td>
<td>11</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>output</td>
<td>1</td>
<td>,2</td>
<td>,2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Venkata Anantharam
Large networks
June 2, 2017 10 / 35
Lempel Ziv

input
0 1 1 0 0 1 0 1 1

dictionary

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>01</td>
<td>11</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

output
1 ,2 ,2
Lempel Ziv

input

0 1 1 0 0 1 0 1 1

dictionary

1 2 3 4 5 6 7 8 9

0 1 01 11 10 00

output

1 ,2 ,2 ,1
Lempel Ziv

input

0 1 1 0 0 1 0 1 1

dictionary

1 2 3 4 5 6 7 8

0 1 01 11 10 00

output

1 ,2 ,2 ,1
Lempel Ziv

input 0 1 1 0 0 1 0 1 1

dictionary

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>01</td>
<td>11</td>
<td>10</td>
<td>00</td>
<td>010</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

output 1 , 2 , 2 , 1 , 3
# Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>dictionary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

| output | 1 ,2 ,2 ,1 ,3 |
Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0 1 1 0 0 1 0 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td></td>
</tr>
<tr>
<td>1 2 3 4 5 6 7 8</td>
<td></td>
</tr>
<tr>
<td>0 1 01 11 10 00 010 011</td>
<td></td>
</tr>
<tr>
<td>output</td>
<td>1 ,2 ,2 ,1 ,3 ,3</td>
</tr>
</tbody>
</table>
Lempel Ziv

input 0 1 1 0 0 1 0 1 1

dictionary

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>01</td>
<td>11</td>
<td>10</td>
<td>00</td>
<td>010</td>
<td>011</td>
</tr>
</tbody>
</table>

output 1, 2, 2, 1, 3, 3
## Lempel Ziv

<table>
<thead>
<tr>
<th>input</th>
<th>0 1 1 0 0 1 0 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>dictionary</td>
<td>1 2 3 4 5 6 7 8</td>
</tr>
<tr>
<td></td>
<td>0 1 01 11 10 00 010 011</td>
</tr>
<tr>
<td>output</td>
<td>1 ,2 ,2 ,1 ,3 ,3 ,2</td>
</tr>
</tbody>
</table>
Universality of typical sequence compression

Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$. 
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$.
- We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.

$k/n$ is a proxy for $p$. There are $\binom{n}{k}$ sequences with these many ones.

$\log (\binom{n}{k}) \approx n H \left( \frac{k}{n} \right) \approx n H(p)$. Even without knowing $p$, we have managed to achieve the correct entropy rate $\Rightarrow$ universal compression.
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$.
- We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.
- $k/n$ is a proxy for $p$. 

$$\log (n^k) \approx n H(k/n) \approx n H(p).$$

Even without knowing $p$, we have managed to achieve the correct entropy rate $\Rightarrow$ universal compression.
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$.
- We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.
- $k/n$ is a proxy for $p$.
- There are $\binom{n}{k}$ sequences with these many ones.
Universality of typical sequence compression

Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.

We want to encode this block, but we do not know $p$.

We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.

$k/n$ is a proxy for $p$.

There are $\binom{n}{k}$ sequences with these many ones.

We can specify the input sequence by $\log \binom{n}{k}$ bits.
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$.
- We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.
- $k/n$ is a proxy for $p$.
- There are $\binom{n}{k}$ sequences with these many ones.
- We can specify the input sequence by $\log \binom{n}{k}$ bits.
- $\log \binom{n}{k} \approx nH(k/n) \approx nH(p)$. 

Even without knowing $p$, we have managed to achieve the correct entropy rate $\Rightarrow$ universal compression.
Universality of typical sequence compression

- Let $X^n = (X_1, \ldots, X_n)$ be a sequence of i.i.d. Bernoulli $p$ random variables.
- We want to encode this block, but we do not know $p$.
- We can first encode the number of ones in the sequence, say $k$, by $\log n$ bits.
- $k/n$ is a proxy for $p$.
- There are $\binom{n}{k}$ sequences with these many ones.
- We can specify the input sequence by $\log \binom{n}{k}$ bits.
- $\log \binom{n}{k} \approx nH(k/n) \approx nH(p)$.
- Even without knowing $p$, we have managed to achieve the correct entropy rate $\Rightarrow$ universal compression.
Outline

1. Universal data compression
2. Graph-indexed (graphical) data
3. The framework of local weak convergence
4. Universal compression of graphical data
Big Graphical Data

Web

≈ 47 billion webpages

FB

1.8 billion active users
Big Graphical Data

Web

≈ 47 billion webpages

Other examples: Biological data

FB

1.8 billion active users
Big Graphical Data

Web

\[ \approx 47 \text{ billion webpages} \]

- Other examples: Biological data

- Need: Analyzing, storing, compression

FB

1.8 billion active users
Big Graphical Data

Web

≈ 47 billion webpages

- Other examples: Biological data
- Need: Analyzing, storing, compression
- Desirable properties: Query the compressed form

FB

1.8 billion active users
Stochastic processes as a model for data samples

\[ \mathcal{X} = (X_n)^\infty_{n=-\infty} \]
Stochastic processes as a model for data samples

- $X = (X_n)_{n=-\infty}^{\infty}$
- $P_{X_0}, P_{X_0, X_1}, P_{X_0, X_1, X_2}, \ldots$
Stochastic processes as a model for data samples

- $\mathcal{X} = (X_n)_{n=-\infty}^{\infty}$
- $P_{X_0}, P_{X_0, X_1}, P_{X_0, X_1, X_2}, \ldots$
- $(X_n)_{n=-\infty}^{\infty}$
Stochastic processes as a model for data samples

- \( \mathcal{X} = (X_n)_{n=-\infty}^{\infty} \)
- \( P_{X_0}, P_{X_0,X_1}, P_{X_0,X_1,X_2}, \ldots \)
- \( (x_n)_{n=-\infty}^{\infty} \)
Stochastic processes as a model for data samples

- $\mathcal{X} = (X_n)_{n=\infty}^{-\infty}$
- $P_{X_0}, P_{X_0, X_1}, P_{X_0, X_1, X_2}, \ldots$
- $(x_n)_{n=\infty}^{-\infty}$
Stochastic processes as a model for data samples

- \( \mathcal{X} = (X_n)_{n=-\infty}^{\infty} \)
- \( P_{X_0}, P_{X_0,X_1}, P_{X_0,X_1,X_2}, \ldots \)
- \( (x_n)_{n=-\infty}^{\infty} \)

\[
\frac{1}{2(N+1) - L} \sum_{i=-N}^{N-L+1} \delta_{x_i, \ldots, x_{i+L-1}} \Rightarrow P_{X_0, \ldots, X_{L-1}}.
\]
“Empirical distribution” of a marked graph

\[ G \]

\[ U_2(G) \]
Rooted marked graph process from a marked graph

\[ G \]

\[ U(G) \]
Rooted marked graph process from a marked graph

\( G \): space of unlabelled marked rooted graphs

\( U(G) \)
Rooted marked graph process from a marked graph

\[ G \]

- \( G_* \): space of unlabelled marked rooted graphs
- A process with values in rooted marked graphs: \( \mu \in \mathcal{P}(G_*) \)

\[ U(G) \]
Large Erdős Rényi graphs

$G(n, \alpha/n)$
Large Erdös Rényi graphs

\[ G(n, \alpha/n) \]

\[(n - 1) \text{Ber}(\alpha/n) \approx \text{Poi}(\alpha) \]
La rge Erdös Rényi graphs

\[ G(n, \alpha/n) \]

\[(n - 1)\text{Ber}(\alpha/n) \approx \text{Poi}(\alpha)\]
Large Erdős Rényi graphs

\[ G(n, \alpha/n) \]

\[(n - 1) \text{Ber}(\alpha/n) \approx \text{Poi}(\alpha)\]

\[(n - 3) \frac{\alpha^2}{n^2} = O(1/n)\]
Poisson Galton-Watson tree

- The local environment of a typical vertex in an Erdös - Rényi graph converges to a Poisson Galton-Watson tree as $M \to \infty$.

**Poisson Galton-Watson tree**:
- Start with a root.
- Pick a Poisson ($\lambda$) number of neighbors (at depth 1).
- For each of these, independently pick a Poisson ($\lambda$) number of neighbors (at depth 2).
- etc.
Outline

1. Universal data compression
2. Graph-indexed (graphical) data
3. The framework of local weak convergence
4. Universal compression of graphical data
The objective method

- $G^*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
The objective method

- $G_*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
- The distance between two elements of $G_*$ is $\frac{1}{1+r}$, where $r$ is the largest depth of a neighborhood around the root up to which they agree.
The objective method

- $\mathcal{G}_*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
- The distance between two elements of $\mathcal{G}_*$ is $\frac{1}{1+r}$, where $r$ is the largest depth of a neighborhood around the root up to which they agree.
- This distance makes $\mathcal{G}_*$ into a complete separable metric space.
The objective method

- $G_*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
- The distance between two elements of $G_*$ is $\frac{1}{1+r}$, where $r$ is the largest depth of a neighborhood around the root up to which they agree.
- This distance makes $G_*$ into a complete separable metric space.
- A fixed finite graph $G$ corresponds to a probability distribution on $G_*$ by picking the root at random from the vertices of $G$. 
The objective method

- $G_*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
- The distance between two elements of $G_*$ is $\frac{1}{1+r}$, where $r$ is the largest depth of a neighborhood around the root up to which they agree.
- This distance makes $G_*$ into a complete separable metric space.
- A fixed finite graph $G$ corresponds to a probability distribution on $G_*$ by picking the root at random from the vertices of $G$.
- A sequence of finite graphs is said to converge in the sense of local weak convergence if the corresponding probability distributions on $G_*$ converge weakly.
The objective method

- $G_*$ denotes the set of locally finite connected rooted graphs considered up to rooted isomorphism.
- The distance between two elements of $G_*$ is $\frac{1}{1+r}$, where $r$ is the largest depth of a neighborhood around the root up to which they agree.
- This distance makes $G_*$ into a complete separable metric space.
- A fixed finite graph $G$ corresponds to a probability distribution on $G_*$ by picking the root at random from the vertices of $G$.
- A sequence of finite graphs is said to converge in the sense of local weak convergence if the corresponding probability distributions on $G_*$ converge weakly.

The definitions extend naturally to marked graphs, i.e., graphs where each edge carries an element of some other separable metric space, as does each vertex.
The objective method (continued)

- $G^{**}$ denotes the set of locally finite connected graphs with a distinguished oriented edge, considered up to isomorphism (preserving the distinguished oriented edge).
The objective method (continued)

- $G^{**}$ denotes the set of locally finite connected graphs with a distinguished oriented edge, considered up to isomorphism (preserving the distinguished oriented edge).
- $G^{**}$ can be metrized to give a complete separable metric space, just as for $G^*$. 
The objective method (continued)

- \( G^{**} \) denotes the set of locally finite connected graphs with a distinguished oriented edge, considered up to isomorphism (preserving the distinguished oriented edge).
- \( G^{**} \) can be metrized to give a complete separable metric space, just as for \( G^{*} \).
- A function \( f: G^{**} \mapsto \mathbb{R} \) gives rise to a function \( \partial f: G^{*} \mapsto \mathbb{R} \) via
  \[
  \partial f(G, o) = \sum_{i \sim o} f(G, i, o).
  \]
- A probability distribution \( \mu \) on \( G^{*} \) gives rise to a measure \( \vec{\mu} \) on \( G^{**} \) via
  \[
  \int_{G^{**}} f d\vec{\mu} = \int_{G^{*}} \partial f d\mu,
  \]
  for all bounded continuous \( f \).
- Note that
  \[
  \vec{\mu}(G^{**}) = \deg(\mu) := \int_{G^{*}} \deg(\text{root}) d\mu.
  \]
The objective method (continued)

- $\mathcal{G}^{**}$ denotes the set of locally finite connected graphs with a distinguished oriented edge, considered up to isomorphism (preserving the distinguished oriented edge).
- $\mathcal{G}^{**}$ can be metrized to give a complete separable metric space, just as for $\mathcal{G}^*$.
- A function $f : \mathcal{G}^{**} \mapsto \mathbb{R}$ gives rise to a function $\partial f : \mathcal{G}^* \mapsto \mathbb{R}$ via
  $$\partial f(G, o) = \sum_{i \sim o} f(G, i, o).$$
- A probability distribution $\mu$ on $\mathcal{G}^*$ gives rise to a measure $\vec{\mu}$ on $\mathcal{G}^{**}$ via
  $$\int_{\mathcal{G}^{**}} fd\vec{\mu} = \int_{\mathcal{G}^*} \partial f d\mu,$$
  for all bounded continuous $f$. 
The objective method (continued)

- $G^{**}$ denotes the set of locally finite connected graphs with a distinguished oriented edge, considered up to isomorphism (preserving the distinguished oriented edge).
- $G^{**}$ can be metrized to give a complete separable metric space, just as for $G^*$.
- A function $f : G^{**} \mapsto \mathbb{R}$ gives rise to a function $\partial f : G^* \mapsto \mathbb{R}$ via
  $$\partial f(G, o) = \sum_{i \sim o} f(G, i, o).$$
- A probability distribution $\mu$ on $G^*$ gives rise to a measure $\vec{\mu}$ on $G^{**}$ via
  $$\int_{G^{**}} fd\vec{\mu} = \int_{G^*} \partial fd\mu,$$
  for all bounded continuous $f$.
- Note that $\vec{\mu}(G^{**}) = \text{deg}(\mu) := \int_{G^*} \text{deg(root)}d\mu$.
Unimodularity

Given $f : \mathcal{G}^{**} \mapsto \mathbb{R}$, define $f^* : \mathcal{G}^{**} \mapsto \mathbb{R}$ via

$$f^*(G, i, o) = f(G, o, i).$$

A probability distribution $\mu$ on $\mathcal{G}_*$ is called unimodular if

$$\int_{\mathcal{G}^{**}} f d\bar{\mu} = \int_{\mathcal{G}^{**}} f^* d\bar{\mu}, \text{ for all bounded continuous } f.$$

It is known that the local weak limit of any sequence of finite graphs is unimodular \textit{(Aldous and Lyons)}. 
Unimodular Galton-Watson trees

- Given a probability distribution \( \{ \pi(i), \ i \geq 0 \} \) on the nonnegative integers, with finite mean \( \sum_i i \pi(i) \), define

\[
\hat{\pi}(i) := \frac{(i + 1) \pi(i + 1)}{\sum_i i \pi(i)}, \quad i \geq 0.
\]

\( \{ \hat{\pi}(i), \ i \geq 0 \} \) is also a probability distribution.
Unimodular Galton-Watson trees

Given a probability distribution \( \{ \pi(i), \ i \geq 0 \} \) on the nonnegative integers, with finite mean \( \sum_i i \pi(i) \), define

\[
\hat{\pi}(i) := \frac{(i + 1)\pi(i + 1)}{\sum_i i \pi(i)}, \quad i \geq 0.
\]

\( \{ \hat{\pi}(i), \ i \geq 0 \} \) is also a probability distribution.

The unimodular Galton-Watson tree, \( \text{UGWT}(\pi) \) is the random tree constructed as follows: Start with a root and give it a random number of children (at depth 1) with the number of children distributed as \( \pi \). For each child, give it a random number of children (at depth 2), the number distributed as \( \hat{\pi} \), independently. Repeat (using \( \hat{\pi} \) from now on).
Unimodular Galton-Watson trees

- Given a probability distribution \( \{\pi(i), \ i \geq 0\} \) on the nonnegative integers, with finite mean \( \sum_i i \pi(i) \), define
  \[
  \hat{\pi}(i) := \frac{(i+1) \pi(i+1)}{\sum_i i \pi(i)}, \quad i \geq 0.
  \]
  \( \{\hat{\pi}(i), \ i \geq 0\} \) is also a probability distribution.

- The unimodular Galton-Watson tree, \( \text{UGWT}(\pi) \) is the random tree constructed as follows: Start with a root and give it a random number of children (at depth 1) with the number of children distributed as \( \pi \). For each child, give it a random number of children (at depth 2), the number distributed as \( \hat{\pi} \), independently. Repeat (using \( \hat{\pi} \) from now on).

- Many standard sequences of bipartite graph models, such as the pairing model based on half edges and fixed degree distributions which shows up in the theory of LDPC codes, have a unimodular Galton-Watson tree as their local weak limit.
Outline

1 Universal data compression

2 Graph-indexed (graphical) data

3 The framework of local weak convergence

4 Universal compression of graphical data
The BC entropy: counting typical graphs

- $\Xi$: edge marks, $\Theta$: vertex marks, both finite
The BC entropy: counting typical graphs

- $\Xi$: edge marks, $\Theta$: vertex marks, both finite
- $\mathcal{G}_{m_n,u_n}^{(n)}$: set of graphs on $n$ vertices with $m_n(x)$ many edges with mark $x \in \Xi$ and $u_n(t)$ many vertices with mark $t \in \Theta$. 
The BC entropy: counting typical graphs

- $\Xi$: edge marks, $\Theta$: vertex marks, both finite
- $\mathcal{G}^{(n)}_{m_n,u_n}$: set of graphs on $n$ vertices with $m_n(x)$ many edges with mark $x \in \Xi$ and $u_n(t)$ many vertices with mark $t \in \Theta$.
- $\mathcal{G}^{(n)}_{m_n,u_n}(\mu, \epsilon) = \{ G \in \mathcal{G}^{(n)}_{m_n,u_n} : U(G) \in B(\mu, \epsilon) \}$. 
The BC entropy: counting typical graphs

- $\Xi$: edge marks, $\Theta$: vertex marks, both finite
- $G_{m_n,u_n}^{(n)}$: set of graphs on $n$ vertices with $m_n(x)$ many edges with mark $x \in \Xi$ and $u_n(t)$ many vertices with mark $t \in \Theta$.
- $G_{m_n,u_n}^{(n)}(\mu, \epsilon) = \{ G \in G_{m_n,u_n}^{(n)} : U(G) \in B(\mu, \epsilon) \}$.
- For $\mu \in \mathcal{P}(G_*)$ and $x \in \Xi$, $\deg_x(\mu)$: expected number of edges connected to the root with mark $x$,
The BC entropy: counting typical graphs

- $\Xi$: edge marks, $\Theta$: vertex marks, both finite
- $\mathcal{G}^{(n)}_{m_n, u_n}$: set of graphs on $n$ vertices with $m_n(x)$ many edges with mark $x \in \Xi$ and $u_n(t)$ many vertices with mark $t \in \Theta$.
- $\mathcal{G}^{(n)}_{m_n, u_n}(\mu, \epsilon) = \{ G \in \mathcal{G}^{(n)}_{m_n, u_n} : U(G) \in B(\mu, \epsilon) \}$.
- For $\mu \in \mathcal{P}(\mathcal{G}_*)$ and $x \in \Xi$, $\deg_x(\mu)$: expected number of edges connected to the root with mark $x$.
- $t \in \Theta$, $\Pi_t(\mu)$: probability of root having mark $t$. 
Fix sequences \( m_n, u_n \) such that \( m_n(x)/n \to \deg_x(\mu)/2 \) and \( u_n(t)/n \to \Pi_t(\mu) \) for all \( x \in \Xi \), \( t \in \Theta \).
The BC entropy: counting typical graphs
(continued)

- Fix sequences $m_n, u_n$ such that $m_n(x)/n \to \deg_x(\mu)/2$ and $u_n(t)/n \to \Pi_t(\mu)$ for all $x \in \Xi, t \in \Theta$.

- $\log |G_{m_n,u_n}^{(n)}| = \|m_n\|_1 \log n + cn + o(n)$ where $\|m_n\|_1 = \sum_{x \in \Xi} m_n(x)$. 

$\Sigma(\mu) := \lim \epsilon \downarrow 0 \limsup_{n \to \infty} \log |G_{m_n,u_n}^{(n)}(\mu,\epsilon)| - \|m_n\|_1 \log n$
The BC entropy: counting typical graphs (continued)

- Fix sequences $m_n, u_n$ such that $m_n(x)/n \to \deg_x(\mu)/2$ and $u_n(t)/n \to \prod_t(\mu)$ for all $x \in \Xi$, $t \in \Theta$.

- $\log |\mathcal{G}_{m_n,u_n}^{(n)}| = \|m_n\|_1 \log n + cn + o(n)$ where $\|m_n\|_1 = \sum_{x \in \Xi} m_n(x)$.

$$
\bar{\Sigma}(\mu) := \lim_{\epsilon \downarrow 0} \lim_{n \to \infty} \frac{\log |\mathcal{G}_{m_n,u_n}^{(n)}(\mu, \epsilon)| - \|m_n\|_1 \log n}{n}
$$

$$
\underline{\Sigma}(\mu) := \lim_{\epsilon \downarrow 0} \lim_{n \to \infty} \frac{\log |\mathcal{G}_{m_n,u_n}^{(n)}(\mu, \epsilon)| - \|m_n\|_1 \log n}{n}
$$
The BC entropy: counting typical graphs (continued)

- Fix sequences $m_n, u_n$ such that $m_n(x)/n \to \deg_x(\mu)/2$ and $u_n(t)/n \to \prod_t(\mu)$ for all $x \in \Xi$, $t \in \Theta$.

- $\log |G_{m_n,u_n}^{(n)}| = \|m_n\|_1 \log n + cn + o(n)$ where $\|m_n\|_1 = \sum_{x \in \Xi} m_n(x)$.

- \[ \bar{\Sigma}(\mu) := \lim_{\epsilon \downarrow 0} \lim_{n \to \infty} \sup \frac{\log |G_{m_n,u_n}^{(n)}(\mu, \epsilon)| - \|m_n\|_1 \log n}{n} \]

- \[ \underline{\Sigma}(\mu) := \lim_{\epsilon \downarrow 0} \lim_{n \to \infty} \inf \frac{\log |G_{m_n,u_n}^{(n)}(\mu, \epsilon)| - \|m_n\|_1 \log n}{n} \]

- If they are equal, define the common value as $\Sigma(\mu)$ (generalization of the BC entropy of Bordenave and Caputo).
Our target for the graph regime

- **Goal**: design $f_n : G_n \to \{0,1\}^*$ and $g_n : \{0,1\}^* \to G_n$
- $g_n \circ f_n = \text{Id}$
- $\mu \in \mathcal{P}(G_*)$ a process
- **Target**: typical graphs
- **Optimal** if $G_n \xrightarrow{lwc} \mu$

\[
\limsup_{n \to \infty} \frac{l(f_n(G_n)) - m_n \log n}{n} \leq \Sigma(\mu),
\]

where $m_n$ is the total number of edges in $G_n$. 
A First Step Coding Scheme: Example

\[ A_{k_n, \Delta_n} = \{ [G, o] \in G_* : \text{depth} \leq k_n, \text{max deg} \leq \Delta_n \} \]

\[ n = 4, \ k_n = 1 \]

\[ \Delta_n = 2 \]
A First Step Coding Scheme: Example

\[ \mathcal{A}_{k_n, \Delta_n} = \{ [G, o] \in G_* : \text{depth} \leq k_n, \max \text{deg} \leq \Delta_n \} \]

\( n = 4, k_n = 1 \)

\[ \Delta_n = 2 \]

\[ W_n := \text{the set of graphs with the same sequence} \]
A First Step Coding Scheme: Example

\[ \mathcal{A}_{kn, \Delta n} = \{ [G, o] \in \mathcal{G}_*: \text{depth} \leq k_n, \text{max deg} \leq \Delta_n \} \]

\[ n = 4, \; k_n = 1 \]

\[ \Delta_n = 2 \]

\[ W_n := \text{the set of graphs with the same sequence} \]
A First Step Coding Scheme: Example

\[ A_{k_n, \Delta_n} = \{ [G, o] \in G_* : \text{depth} \leq k_n, \text{max deg} \leq \Delta_n \} \]

\[ n = 4, k_n = 1 \]

\[ \Delta_n = 2 \]

\[ W_n := \text{the set of graphs with the same sequence} \]
Analysis Outline

- $l(f_n(G_n))$, the total number of bits we use:
  - $\log n$ bits for $\Delta_n$,
  - $|A_{k_n,\Delta_n}| \log n$ bits for specifying how many times each pattern appears in the graph
  - $\log |W_n|$ bits to specify the input graph among the graphs with the same pattern counts.

- We need to show that if $G_n \xrightarrow{\text{lwc}} \mu$,

$$
\frac{l(f_n(G_n)) - m_n \log n}{n} \leq \bar{\Sigma}(\mu).
$$

- If $|A_{k_n,\Delta_n}| = o\left(\frac{n}{\log n}\right)$, we only need to consider the $\log |W_n|$ term.

- Graphs in $W_n$ are typical $\Rightarrow$ yields $\bar{\Sigma}(\mu)$ as an upper bound.
First step algorithm: Main Result

Proposition

If parameters $k_n$ and $\Delta_n$ are such that $|A_{k_n,\Delta_n}| = o\left(\frac{n}{\log n}\right)$ and $k_n \to \infty$ as $n \to \infty$, for any sequence $G_n$ with maximum degree no more than $\Delta_n$ and local weak limit $\mu \in \mathcal{P}(G_\ast)$ such that $\overline{\Sigma}(\mu) > -\infty$ we have

$$\limsup_{n \to \infty} \frac{l(f_n(G_n)) - m_n \log n}{n} \leq \overline{\Sigma}(\mu),$$

(1)

where $m_n$ is the number of edges in $G_n$. 
General Algorithm

\[ \Delta_n = 5 \rightarrow \tilde{G}_n \]

\[ T_n = \{ \text{endpoints of removed edges} \} \]

Compress \( \tilde{G}_n \) using the first step scheme, then compress removed edges.

\[ \Delta_n = \log \log n \]

\[ |T_n| / n \rightarrow 0 \]

\[ |A_{\Delta_n}| = o \left( \frac{n}{\log n} \right) \]

\[ G_n \text{lwc} \rightarrow \mu \Rightarrow \tilde{G}_n \text{lwc} \rightarrow \mu \]

Venkata Anantharam

Large networks

June 2, 2017 31 / 35
General Algorithm

$\Delta_n = 5 \rightarrow \tilde{G}_n$

$T_n = \{\text{endpoint of removed edges}\}$

Compress $\tilde{G}_n$ using the first step scheme, then compress removed edges.

$\Delta_n = \log \log n$

$k_n = \sqrt{\log \log n}$

$|T_n|/n \rightarrow 0$

$|A_k n, \Delta_n| = o(n/\log n)$

$G_n \rightarrow \mu \Rightarrow \tilde{G}_n \rightarrow \mu$

Venkat Anantharam
Large Networks
June 2, 2017
31 / 35
General Algorithm

\[ \Delta_n = 5 \rightarrow \tilde{G}_n = \{ \text{endpoints of removed edges} \} \]

Compress \( \tilde{G}_n \) using the first step scheme, then compress removed edges.

\[ \Delta_n = \log \log n \rightarrow \frac{|T_n|}{n} \rightarrow 0 \]

\[ |A_k^n, \Delta_n| = o \left( \frac{n}{\log n} \right) \]

Large networks

June 2, 2017
General Algorithm

\[ \Delta_n = 5 \rightarrow \tilde{G}_n \]

\[ T_n = \{\text{endpoint of removed edges}\} \]

\[ \tilde{G}_n \]
General Algorithm

\[ G_n \xrightarrow{\Delta_n=5} \tilde{G}_n \]

\[ T_n = \{ \text{endpoint of removed edges} \} \]

Compress \( \tilde{G}_n \) using the first step scheme, then compress removed edges.
General Algorithm

\[
\Delta_n = 5 \rightarrow \tilde{G}_n
\]

\[
T_n = \{ \text{endpoint of removed edges} \}
\]

Compress \( \tilde{G}_n \) using the first step scheme, then compress removed edges

\[ \Delta_n = \log \log n \quad \quad k_n = \sqrt{\log \log n} \]
General Algorithm

Compress $\tilde{G}_n$ using the first step scheme, then compress removed edges

$\Delta_n = \log \log n$ \hspace{1cm} $k_n = \sqrt{\log \log n}$

$|T_n|/n \to 0$ \hspace{1cm} $|A_{k_n, \Delta_n}| = o(n/\log n)$ \hspace{1cm} $G_n \overset{\text{lwc}}{\rightarrow} \mu \Rightarrow \tilde{G}_n \overset{\text{lwc}}{\rightarrow} \mu$
Theorem

Assume $\mu \in \mathcal{G}_*$ with $\deg_x(\mu) < \infty$ for all $x$ and $\Sigma(\mu) > -\infty$. If $G_n$ is a sequence of marked graphs with local weak limit $\mu$, we have

$$\limsup_{n \to \infty} \frac{l(f_n(G_n)) - m_n \log n}{n} \leq \Sigma(\mu),$$

where $m_n$ is the number of edges in $G_n$. 
Result: Converse

**Theorem**

Assume $\mu \in \mathcal{P}(G_\ast)$ with $\Sigma(\mu) > -\infty$ and $\deg_x(\mu) < \infty$ for all $x \in \Xi$. Then there exists a sequence of graph ensembles $G_n$ converging to $\mu$ such that with probability one for any sequence of compression schemes $f_n$ we have

$$\liminf_{n \to \infty} \frac{l(f_n(G_n)) - m_n \log n}{n} \geq \Sigma(\mu),$$

where $m_n$ is the number of edges in $G_n$. 
Conclusion

- Discussed a notion of graph-indexed stochastic process.
- Formulated the problem of universal compression for graphical data through this language.
- Discussed a notion of entropy based on counting typical graphs.
- Proposed an optimal universal compression scheme.
The End