Review of CSE427

Sheng Wang
A rich hierarchy of biological subsystems at multiple scales: genotypic variations in nucleotides (1 nm scale) -> proteins (1–10 nm) -> protein complexes (10–100 nm), cellular processes (100 nm) -> phenotypic behaviors of cells (1–10 μm), tissues (100 μm to 100 mm), -> complex organisms (>1 m).

How a computer scientist study comp bio?
Understand the input and output first

Gene (1 nm) ➞ Protein complexes (function) (10-100nm) ➞ Cell (1-10 μm) ➞ Tissue (100 μm to 100 mm) ➞ Complex organism (>1 cm)

Biologists: which input should I use for this problem? Gene expression? Tissue images?

Computer scientists: Given the input we have, which method should we use to solve this problem?
Data structure for each scale: protein

Gene (1 nm) → Protein complexes (function) (10-100nm) → Cell (1-10 μm) → Tissue (100 μm to 100 mm) → Complex organism (>1 cm)

A sequence of amino acids/nucleic acids -> A sequence of word/character
NLP methods (edit distance, LSTM, BERT)

Computational challenge: modeling the order in the sequence
Data structure for each scale: network

A network of proteins/genes -> Social network
Graph analysis methods (random walk, pagerank, graph neural network)

Computational challenge: interaction, synergistic effect
Data structure for each scale: cell

Gene (1 nm)

Protein complexes (function) (10-100nm)

Cell (1-10 μm)

Tissue (100 μm to 100 mm)

Complex organism (>1 cm)

A cell by gene matrix -> vector/matrix (high-dimensional, no spatial information)
Dimensionality reduction methods (PCA, t-SNE, variety of embedding methods)

High-dimensional, noisy, large-scale
Data structure for each scale: tissue

Gene (1 nm) → Protein complexes (function) (10-100nm) → Cell (1-10 μm) → Tissue (100 μm to 100 mm) → Complex organism (>1 cm)

Tissue image -> image analysis
Image analysis (segmentation, detection, CNN)

Image analysis, lack of high-quality annotations
Data structure for each scale: organism

- Gene (1 nm)
- Protein complexes (function) (10-100 nm)
- Cell (1-10 μm)
- Tissue (100 μm to 100 mm)
- Complex organism (>1 cm)

Disease mechanisms -> Multimodality
Integration of information from sequences, networks, images and matrixes

Multi-modality and heterogeneous
How did they do this?

DNA sample

Sequencing machine
~2000 dollars

Your entire genome sequence
*.fastq file

Our job as a computer scientist: analyze *.fastq file
What does a fastq file look like?

Very large! ~300000000 lines
Quality: ASCII chars

What should we do? Map each short sequence (we call it read) to the entire human genome
What does a fastq file look like?

Reference genome: “average” human genome.
Most widely used human genome GRCh38: derived from 13 thirteen anonymous volunteers.

Image from Nebula Genomics
### Processed data

#### countData

<table>
<thead>
<tr>
<th>gene</th>
<th>ctrl_1</th>
<th>ctrl_2</th>
<th>exp_1</th>
<th>exp_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>geneA</td>
<td>10</td>
<td>11</td>
<td>56</td>
<td>45</td>
</tr>
<tr>
<td>geneB</td>
<td>0</td>
<td>0</td>
<td>128</td>
<td>54</td>
</tr>
<tr>
<td>geneC</td>
<td>42</td>
<td>41</td>
<td>59</td>
<td>41</td>
</tr>
<tr>
<td>geneD</td>
<td>103</td>
<td>122</td>
<td>1</td>
<td>23</td>
</tr>
<tr>
<td>geneE</td>
<td>10</td>
<td>23</td>
<td>14</td>
<td>56</td>
</tr>
<tr>
<td>geneF</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

#### colData

<table>
<thead>
<tr>
<th>gene</th>
<th>treatment</th>
<th>sex</th>
</tr>
</thead>
<tbody>
<tr>
<td>geneA</td>
<td>ctrl_1</td>
<td>control</td>
</tr>
<tr>
<td>geneB</td>
<td>ctrl_2</td>
<td>control</td>
</tr>
<tr>
<td>geneC</td>
<td>exp_1</td>
<td>treatment</td>
</tr>
<tr>
<td>geneD</td>
<td>exp_2</td>
<td>treatment</td>
</tr>
</tbody>
</table>

Sample names: `ctrl_1, ctrl_2, exp_1, exp_2`
Finding alignments: trace back

Arrows = (ties for) max in F(i,j); 3 LR-to-UL paths = 3 optimal alignments

<table>
<thead>
<tr>
<th>i</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>C</td>
<td>A</td>
<td>T</td>
<td>G</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Arrows indicate the direction of the optimal alignment.
Global Alignment vs. Local alignment

**Needleman-Wunsch algorithm**

- **Initialization:** $F(0, 0) = 0$
- **Iteration:**
  
  $F(i, j) = \max \begin{cases} 
  F(i - 1, j) - d \\
  F(i, j - 1) - d \\
  F(i - 1, j - 1) + s(x_i, y_j) 
  \end{cases}$
- **Termination:** Bottom right

**Smith-Waterman algorithm**

- **Initialization:** $F(0, j) = F(i, 0) = 0$
- **Iteration:**

  $F(i, j) = \max \begin{cases} 
  F(i - 1, j) - d \\
  F(i, j - 1) - d \\
  F(i - 1, j - 1) + s(x_i, y_j) 
  \end{cases}$

- **Termination:** Anywhere
What is protein function prediction?

Human body = country

Single cell = town

Protein = brick, window, carpet, etc.

Protein function = fireproof, soundproof, etc.

**Goal:** classify each protein into its protein functions (multi-label)

**Solution:** find proteins with similar sequences
Problem setting for protein function prediction

Feature extraction

Protein 1
MAEAPQVVEIDP......RPRSGTWPLP

Protein 2
SVLLRSGLGPLG......VWAGFELAWQ

Protein 3
MAEAPQVVEIDP......TWPLPRPEFS

Classifier

Label modeling

Known association

Unknown association
Converting proteins to numeral features

source: Deep learning for drug repurposing: methods, databases, and applications
Ontology of great cats

Lion

Leopard

Jaguar

Snow leopard

Tiger

Cat
Ontology of great cats
Ontology of great cats

- Lion
- Leopard
- Jaguar
- Snow leopard
- Tiger
- Cat

Ontological Classification of Unseen Animals
Ontology of great cats

- Lion
- Leopard
- Jaguar
- Snow leopard
- Tiger
- Cat
Ontology of great cats

- Lion
- Leopard
- Jaguar
- Snow leopard
- Tiger
- Cat
Ontology of great cats

- Lion
- Leopard
- Jaguar
- Snow leopard
- Tiger
- Cat

Ontological classification of unseen animals
Ontology of great cats

Lion

Leopard

Jaguar

Snow leopard

Tiger

Cat
solution: use gene ontology as side information

Each node is a function. 23k functions in total.
solution: use gene ontology as side information
solution: use gene ontology as side information

Seen functions

Protein embedding space
solution: use gene ontology as side information
solution: use gene ontology as side information

Protein embedding space
Exact Subgraph Enumeration

K=3

Node set (currently in the motif)
Candidate set (neighbors of node set)
Nodes in the candidate set must have larger node id than nodes in the node set to avoid duplicate computing
Graph Isomorphism

- If you ignore the node types/features and edge types/features, you will find some subgraphs are topologically equivalent.

\[ \begin{array}{c}
\text{A} & 4 \\
\text{B} & 7 \\
\text{C} & 1 \\
\text{D} & 3 \\
\text{E} & 5 \\
\text{F} & 8 \\
\text{G} & 2 \\
\text{H} & 9 \\
\text{I} & 6 \\
\end{array} \]

- Graphs G and H are isomorphic if there exists a bijection \( f: V(G) \rightarrow V(H) \) such that: Any two nodes \( u \) and \( v \) of \( G \) are adjacent in \( G \) iff \( f(u) \) and \( f(v) \) are adjacent in \( H \).
Graph Isomorphism Detection Algorithm

- **McKay’s Canonical Graph Labeling Algorithm**: Nauty, Trace, Bliss all have their own implementations of Mckay’s algorithm. [McKay 1981]
- Time complexity $\exp(O(n^{2/3}))$
- Intuition: First hash two graphs as two strings and then compare two strings.
- Label each node according to their degrees first. Iterate over each edge.
- Put a “1” if there is an edge between those two nodes, a "0" if not.

\[
\begin{align*}
(1,2) & (1,3) & (1,4) & (1,5) & (1,6) & \quad 00100 \\
(2,3) & (2,4) & (2,5) & (2,6) & \quad 0001 \\
(3,4) & (3,5) & (3,6) & \quad 010 \\
(4,5) & (4,6) & \quad 01 \\
(5,6) & \quad 1 \\
\end{align*}
\]

$N$ node: $N \times (N-1)/2$ edges

5 node: 15 edges
But the order of the edge matters in this hash coding. To solve that problem we want to enumerate all the orderings.

- We first sort the all the nodes according to their degrees.
- Within each degree bin, we enumerate all the orderings.

Graph Isomorphism Detection Algorithm

```
{A, B, C} | {D, E} | {F}
{A, B, C} | {D, E} | {F}  {A, C, B} | {D, E} | {F}  {B, A, C} | {D, E} | {F}  {B, C, A} | {D, E} | {F}
```

This hash code matches the previous one!
Random walk interpretation

The vector $r$ can be reinterpreted as a probability vector to visit each website

- Imagine a random web surfer
  - At any time $k$, surfer has a probability vector $r^k$ to visit a web page following the out-link.
  - Process repeats indefinitely

$$
r = Mr$$
Overview of network-based stratification

Source: Network-based stratification of tumor mutations
Performing random walk with restart for each patient

\[ F_{t+1} = \alpha F_t A + (1-\alpha)F_0 \]

The new mutation status for patient 1, gene j

First consider the mutations for all the genes of patient 1

Then consider all the neighbors of gene j

The original patient mutation matrix

Source: Network-based stratification of tumor mutations
Random walk has stationary distribution when the graph is irreducible and aperiodic.

- **Irreducible**: There is a path from every node to every other node.

![Irreducible vs Not Irreducible Graphs](image)

Source: Random walk on graphs: an overview
Aperiodic: The GCD of all cycle lengths is 1. The GCD is also called period.

Periodicity is 3

Aperiodic

The greatest common divisor of a set of whole numbers is the largest integer which divides them all.

Example: The greatest common divisor of 12 and 15.

\[ \gcd(12, 15). \]

Divisors of 12: 1, 2, 3, 4, 6, 12.

Divisors of 15: 1, 3, 5, 15.

Common divisors: 1, 3.

Greatest common divisor is 3.

\[ \therefore \gcd(12, 15) = 3. \]
Solution: jump to a random node

At each time step, the random surfer has two options

- With prob. $\beta$, follow a link at random
- With prob. $1 - \beta$, jump to a random page
- Common values for $\beta$ are in the range 0.8 to 0.9

$$r_j = \sum_{i \rightarrow j} \beta \frac{r_i}{d_i} + (1 - \beta) \frac{1}{N}$$

$$r_1 = \beta \frac{1}{d_1}, 0, \ldots$$
$$r_2 = \beta \frac{1}{d_1}, \frac{1}{d_2}, \ldots$$
$$\vdots$$
$$r_j = \beta 0, \frac{1}{d_2}, \ldots$$
$$\vdots$$
$$r_N = \beta \frac{1}{d_2}, \ldots$$

+ $(1 - \beta)$

$$1/N$$
$$1/N$$
$$\vdots$$
$$1/N$$
Difference from random walk

Random walk

\[
\begin{align*}
\mathbf{r}_1 &= \beta \\
\mathbf{r}_2 &= \\
\vdots &\vdots \\
\mathbf{r}_j &= \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{d_1}, 0, \ldots \\
\frac{1}{d_1}, \frac{1}{d_2}, \ldots \\
\vdots &\vdots \\
0, \frac{1}{d_2}, \ldots \\
\end{align*}
\]

\[
\begin{align*}
\mathbf{r}_1 &= \beta \\
\mathbf{r}_2 &= \\
\vdots &\vdots \\
\mathbf{r}_j &= \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{N}, \frac{1}{N}, \ldots \\
(1 - \beta)
\end{align*}
\]

Random walk with restart

\[
\begin{align*}
\mathbf{r}_1 &= \beta \\
\mathbf{r}_2 &= \\
\vdots &\vdots \\
\mathbf{r}_j &= \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{d_1}, 0, \ldots \\
\frac{1}{d_1}, \frac{1}{d_2}, \ldots \\
\vdots &\vdots \\
0, \frac{1}{d_2}, \ldots \\
\end{align*}
\]

\[
\begin{align*}
\mathbf{r}_1 &= \beta \\
\mathbf{r}_2 &= \\
\vdots &\vdots \\
\mathbf{r}_j &= \\
\end{align*}
\]

\[
\begin{align*}
\frac{c_1}{c_j}, \frac{c_2}{c_j}, \ldots \\
(1 - \beta)
\end{align*}
\]

\[
\begin{align*}
\frac{0}{0}, 0, \ldots \\
0, 1, \ldots \\
\vdots &\vdots \\
0, 0, \ldots \\
\end{align*}
\]
From advanced matrix to random walk probability matrix

Image adapted from MultiVERSE: a multiplex and multiplex-heterogeneous network embedding approach
Somatic mutation profile

- Compare the mutations of tumors
- Sparse

Supplementary Figure 1

Source: Network-based stratification of tumor mutations
Precision medicine: the right patient, the right drug, the right time, the right dose.

Precision medicine ensures delivery of the right intervention to the right patient at the right time.

We don’t have so many “drugs”

- Discovery new drug?
  - Often not in the scope of precision medicine
  - New patient cannot wait for a new drug
- Drug repurposing
  - Drug A, which is used to treat disease X, is later used to treat disease Y
  - Well-documented side effects and less restriction from FDA
- Drug combination
  - Drug A is not effective. Drug B is not effective. Drug A and B used together is effective.
- Personalized dosage
  - Widely used in clinics. Use genomics data to determine dosage (regression).
Use gene expression after treatment

Drugs target on similar proteins or have similar Mode of Actions have similar (after treatment) expression.

Iorio et al. Discovery of drug mode of action and drug repositioning from transcriptional responses
Synthetic lethality: Gene A OR Gene B

Question: how to leverage SL in drug combination discovery?

Source: wikipedia, Jerby-Arnon et al. Predicting Cancer-Specific Vulnerability via Data-Driven Detection of Synthetic Lethality
Drug treatment based on synthetic lethality

Goal: We want to make normal cells survive and kill cancer cells (BRCA deficient cancer cells)

Prior knowledge: PARP1 (off) + BRCA1 (off) -> cell death

Solution: Turn off PARP1 using Olaparib

Results:
- Normal cells: PARP1 (off) + BRCA1 (on) -> cell survive
- Cancer cells: PARP1 (off) + BRCA1 (off) -> cell death

Gilad et al. Drug Combination in Cancer Treatment—From Cocktails to Conjugated Combinations
Drug combination prediction

\[ E_{A,B} < E_A + E_B \]
\[ E_{A,B} = E_A + E_B \]
\[ E_{A,B} > E_A + E_B \]

E(A) is the efficacy of using drug A (e.g., IC50)