Consistency Meets Inconsistency: A Unified Graph Learning Framework for Multi-view Clustering

Youwei Liang[1], Dong Huang[1], Chang-Dong Wang[2]

[1]South China Agricultural University, Guangzhou, China

[2]Sun Yat-sen University, Guangzhou, China

Introduction

- **Spectral clustering**
  - **Traditional way:** Pre-define an affinity graph -> Partition it.
  - **Its performance heavily relies on the pre-defined graph!**

- **Graph learning:** Can we adaptively learn the graph?
  - **(Single-view) graph learning (**Nie et al. KDD 2015; Nie et al. AAAI 2016)
  - **Multi-view graph learning** (Zhan et al. TKDE 2018; Nie et al. IJCAI 2017)

Multi-view Graph Learning (Graph Fusion)

- **Zhan et al. (TKDE 2018; TIP 2019) fused multiple graphs into a consistent graph** with a certain number of connected components.

- **Nie et al. (IJCAI 2017)** proposed a self-weighted scheme to **fuse multiple graphs with the importance of each view considered**.

These methods focus on multi-view consistency, yet cannot simultaneously and explicitly consider both multi-view consistency and inconsistency.

The inconsistency is a much broader concept than noise. It may be caused by not just noise/corruptions, but also different kinds of view-specific characteristics.

Consistency & Inconsistency

- In this paper, we propose **a new multi-view graph learning approach for multi-view clustering**.

- We argue that the simultaneous modeling of **multi-view consistency** and **multi-view inconsistency** can significantly benefit the multi-view graph learning process.

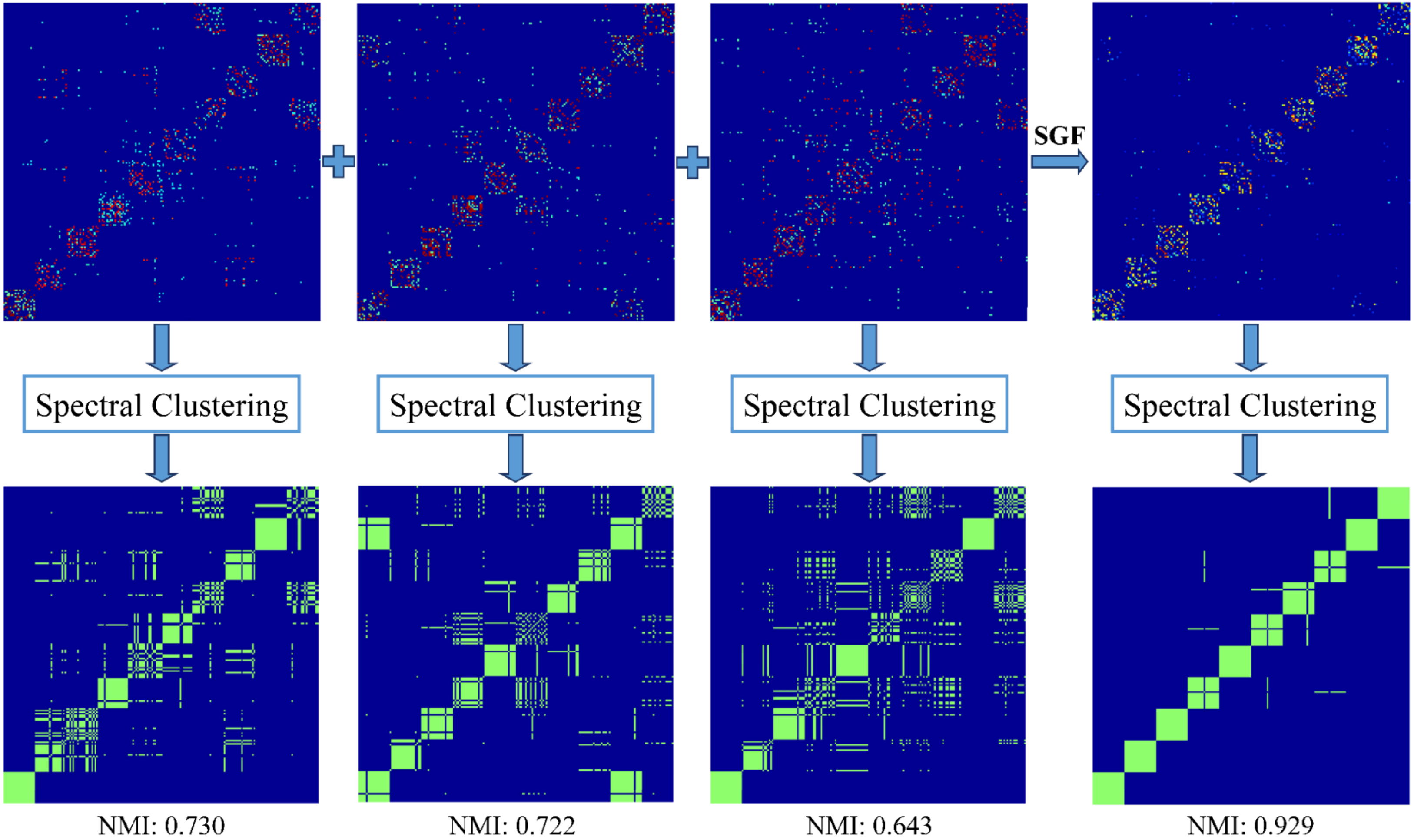
Decomposition & Fusion

- It is assumed that **the graph of each view can be decomposed into two parts, i.e., the consistent part and the inconsistent part.**

- Given graphs of multiple views, our goal is **to learn and remove the inconsistent parts while preserving and fusing the consistent parts**.

  ✓ The graphs can be similarity graphs or distance (dissimilarity) graph.

Illustration



Objective Function

- A naïve objective function:

$$\min_{\boldsymbol{\alpha},\mathbf{S}} \quad \sum_{i=1}^{v} \|\alpha_i \mathbf{W}^{(i)} - \mathbf{S}\|_F^2$$
$$\text{s.t.} \quad \boldsymbol{\alpha}^\top \mathbf{1} = 1, \alpha \geqslant 0, \mathbf{S} \geqslant 0.$$

- We decompose the adjacent matrix $\mathbf{W}^{(i)}$ for the i-th view into two matrices $\mathbf{A}^{(i)}$ and $\mathbf{E}^{(i)}$

$$\mathbf{W}^{(i)} = \overbrace{\mathbf{A}^{(i)}}^{\text{consistent part}} + \overbrace{\mathbf{E}^{(i)}}^{\text{inconsistent part}}$$

Some Other Terms

1. Let **the sum of the products of the inconsistent parts be small**:

Element-wise multiplication

$$\gamma \sum_{\substack{i,j=1 \\ i \neq j}}^{V} \mathrm{sum}\left(\left(\alpha_i \mathbf{E}^{(i)}\right) \circ \left(\alpha_j \mathbf{E}^{(j)}\right) \right)$$

Simply speaking, the inconsistent parts from different views should have little in common!

2. Additionally, we do not want the inconsistent parts to be too large:

$$\beta \sum_{i=1}^{V} \mathrm{sum}\left(\left(\alpha_i \mathbf{E}^{(i)}\right) \circ \left(\alpha_i \mathbf{E}^{(i)}\right) \right)$$

Objective Function

- Considering that

$$\operatorname{sum}\left(\left(\alpha_i \mathbf{E}^{(i)}\right) \circ \left(\alpha_j \mathbf{E}^{(j)}\right)\right) \;=\; \alpha_i \alpha_j \operatorname{Tr}\left(\mathbf{E}^{(i)} \cdot \left(\mathbf{E}^{(j)}\right)^\top\right)$$

We have the unified objective function:

Fuse the consistent parts

Make **different** inconsistent parts to have little in common

$$\min_{\substack{\boldsymbol{\alpha},\mathbf{S},\\ \mathbf{A}^{(1)},\dots,\mathbf{A}^{(v)}}} \sum_{i=1}^{v} \left\| \alpha_i \mathbf{A}^{(i)} - \mathbf{S} \right\|_F^2 + \gamma \sum_{\substack{i,j=1\\ i\neq j}}^{V} \underbrace{\alpha_i \alpha_j \operatorname{Tr}\left(\mathbf{E}^{(i)} \cdot \left(\mathbf{E}^{(j)}\right)^\top\right)}_{\operatorname{sum}\left(\left(\alpha_i \mathbf{E}^{(i)}\right)\circ\left(\alpha_j \mathbf{E}^{(j)}\right)\right)}$$

$$+ \beta \sum_{i=1}^{V} \alpha_i^2 \operatorname{Tr}\left(\mathbf{E}^{(i)} \cdot \left(\mathbf{E}^{(i)}\right)^\top\right)$$

Prevent **each** inconsistent part from being too large

$$\text{s.t.} \quad \boldsymbol{\alpha}^\top \mathbf{1} = 1, \boldsymbol{\alpha} \geqslant 0, \mathbf{S} \geqslant 0,$$

$$\mathbf{W}^{(i)} \geqslant \mathbf{A}^{(i)} \geqslant 0, \quad i = 1, \dots, v$$

Optimization

- **As the objective function is not jointly convex on all variables, we use an alternating minimization scheme to optimize it.**

Alternate between:

- Fix $\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(v)}$, update α and $\mathbf{S}$
- Fix α and $\mathbf{S}$, update $\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(v)}$

until convergence.

- **Specifically, we develop an efficient algorithm based on projection to solve these two sub-problems.**
  - ✓ Basic idea:
    - Solve the sub-problems with no constraints
    - Project the solutions into feasible region so that they meet the constraints

Overall Algorithm

Algorithm 1 Consistent Graph Learning

Input: Adjacency matrices $\{\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(v)}\}$, parameters β and γ, max iteration m

Output: Adjacency matrix of the consistent graph $\mathbf{S}$

```
Initialize A^(i): A^(i) ← W^(i),   i = 1, ..., v
while not converge do
    Obtain α̃ by solving Eq. (16)
    Project α̃ onto the feasible region using Alg. [12]
    Update S using Eq. (12)
    Obtain vec(A^(i)) by Eq. (23)
    Reshape vec(A^(i)) into a matrix A^(i)
    Project A^(i) onto the feasible region using Eq. (24)
    if reach max iteration then
        break
    end if
end while
```

Two Graph Fusion-based Variants: SGF & DGF

- Notice that our multi-view graph learning technique is **applicable to both similarity graphs and dissimilarity graphs**.

- It leads to two graph fusion-based variants:
  - Similarity graph fusion (**SGF**)
  - Distance (dissimilarity) graph fusion (**DGF**)

Experiments: Datasets

TABLE I

STATISTICS OF THE REAL-WORLD DATASETS

dataset	# of instances	# of views	# of clusters
UCI Digits	2000	6	10
NUS-WIDE	2000	5	31
MSRCv1	210	5	7
Flower17	1360	7	17
Caltech101-7	1474	6	7
Caltech101-20	2386	6	20
BBCSport	544	2	5
Reuters	1500	5	6

Experiments: Convergence Analysis

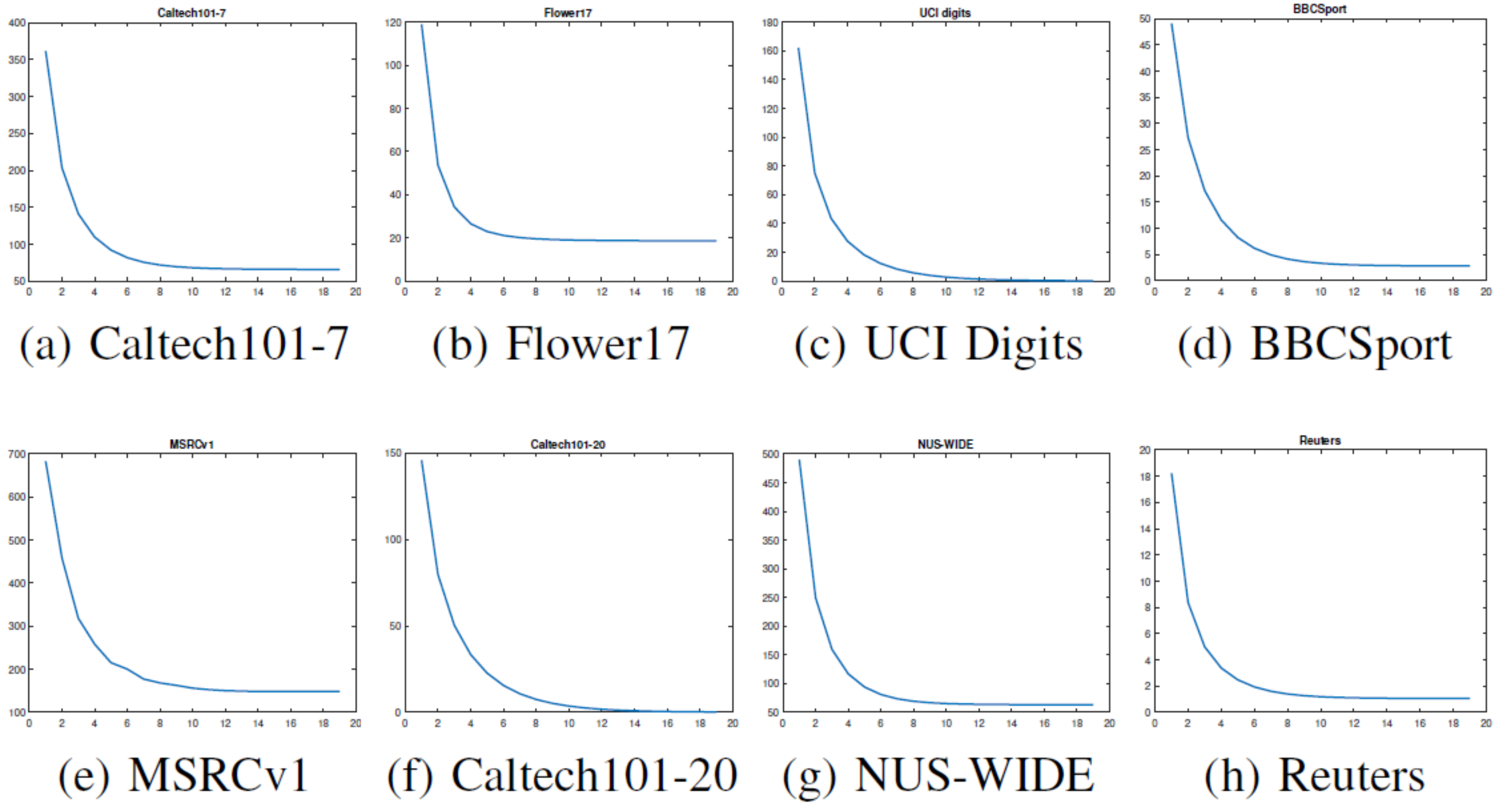


Fig. 2. Convergence curves of the proposed algorithm on the eight datasets. The Y-axis are the objective value, and the X-axis are number of iterations.

Experiments: Parameter Analysis

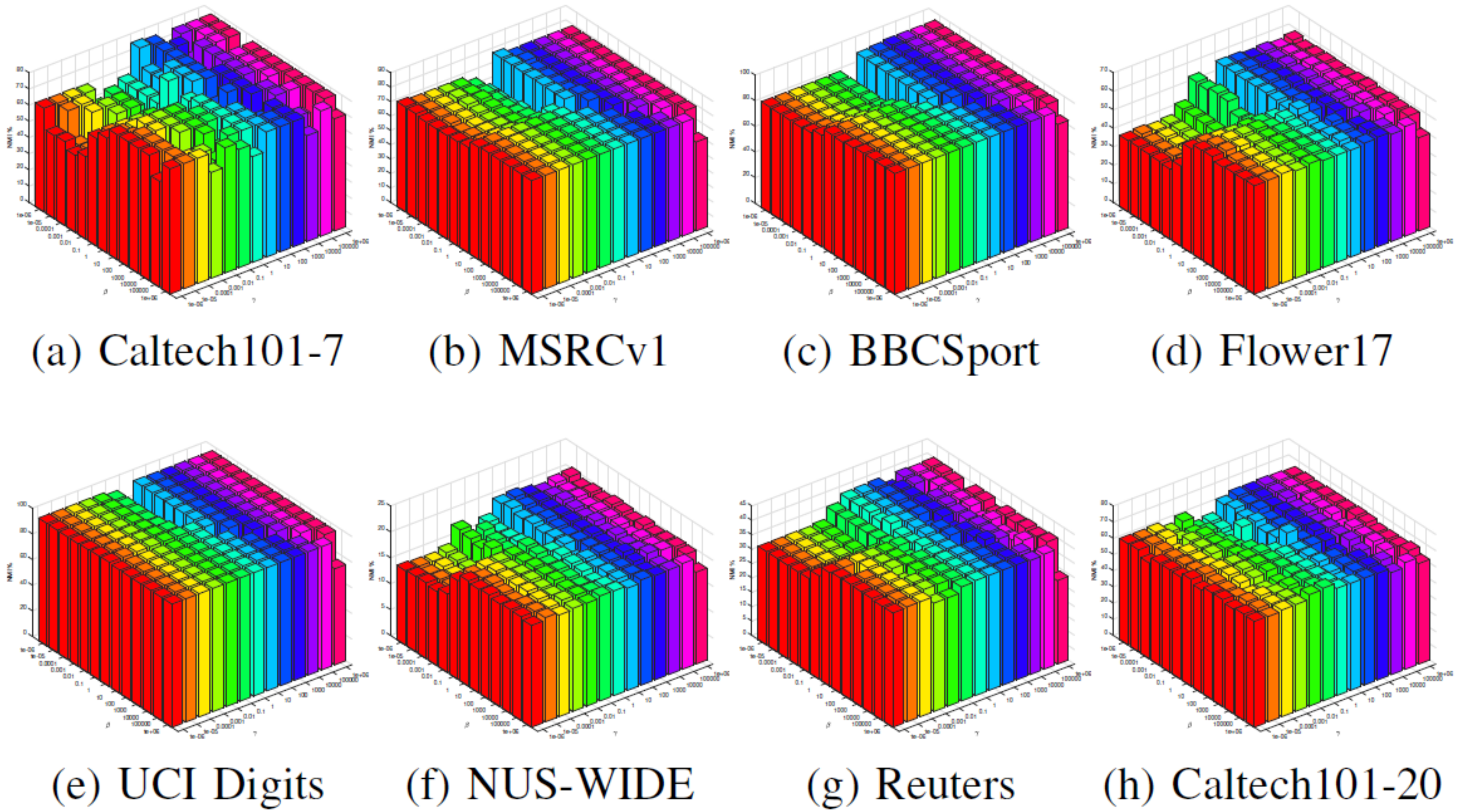


Fig. 3. NMI against parameters β and γ on each dataset.

Experiments: Comparison

TABLE II
AVERAGE PERFORMANCES (W.R.T. NMI (%)) OVER 20 RUNS BY DIFFERENT ALGORITHMS. THE BEST TWO SCORES IN EACH COLUMN ARE IN BOLD.

Method	Caltech101-7	MSRCv1	BBCSport	Flower17	UCI Digits	NUS-WIDE	Reuters	Caltech101-20
SC(best)	$52.42_{\pm 0.97}$	$62.46_{\pm 0.00}$	$81.73_{\pm 0.00}$	$46.94_{\pm 0.42}$	$84.69_{\pm 0.04}$	$17.27_{\pm 0.25}$	$30.32_{\pm 0.04}$	$54.36_{\pm 0.97}$
CoRegSC	$48.21_{\pm 0.00}$	$75.89_{\pm 0.15}$	$\mathbf{93.15}_{\pm 0.00}$	$55.50_{\pm 0.13}$	$93.65_{\pm 0.04}$	$19.20_{\pm 0.27}$	$\mathbf{36.93}_{\pm 0.00}$	$56.79_{\pm 1.06}$
RMSC	$49.45_{\pm 0.14}$	$73.97_{\pm 0.00}$	$91.63_{\pm 3.38}$	$55.57_{\pm 0.35}$	$85.96_{\pm 1.10}$	$19.10_{\pm 0.27}$	$34.38_{\pm 0.00}$	$59.88_{\pm 1.03}$
AASC	$53.85_{\pm 0.07}$	$75.12_{\pm 0.51}$	$90.34_{\pm 0.00}$	$57.98_{\pm 0.21}$	$88.64_{\pm 0.03}$	$19.58_{\pm 0.26}$	$33.44_{\pm 0.00}$	$61.35_{\pm 0.48}$
MVGL	$55.52_{\pm 0.00}$	$70.86_{\pm 0.00}$	$92.47_{\pm 0.00}$	$45.50_{\pm 0.00}$	$88.91_{\pm 0.00}$	$10.32_{\pm 0.00}$	$27.62_{\pm 0.00}$	$59.07_{\pm 0.00}$
MCGC	$51.26_{\pm 0.00}$	$69.62_{\pm 0.00}$	$91.42_{\pm 0.00}$	$50.43_{\pm 0.64}$	$94.22_{\pm 0.00}$	$16.31_{\pm 0.53}$	$30.10_{\pm 0.00}$	$59.59_{\pm 0.00}$
AWP	$48.59_{\pm 1.44}$	$68.98_{\pm 4.32}$	$89.84_{\pm 6.99}$	$51.49_{\pm 1.20}$	$88.65_{\pm 4.18}$	$17.15_{\pm 0.42}$	$30.61_{\pm 2.89}$	$56.86_{\pm 1.75}$
WMSC	$51.22_{\pm 0.00}$	$75.34_{\pm 0.34}$	$92.85_{\pm 0.00}$	$57.93_{\pm 0.50}$	$91.04_{\pm 0.04}$	$19.04_{\pm 0.30}$	$35.02_{\pm 0.70}$	$57.48_{\pm 0.81}$
SGF	$\mathbf{56.07}_{\pm 0.06}$	$\mathbf{76.92}_{\pm 0.14}$	$92.28_{\pm 0.00}$	$\mathbf{64.83}_{\pm 0.21}$	$\mathbf{94.54}_{\pm 0.00}$	$\mathbf{19.61}_{\pm 0.40}$	$35.04_{\pm 0.03}$	$\mathbf{61.58}_{\pm 0.72}$
DGF	$\mathbf{75.55}_{\pm 5.02}$	$\mathbf{81.29}_{\pm 0.00}$	$\mathbf{94.05}_{\pm 0.00}$	$\mathbf{58.13}_{\pm 0.53}$	$\mathbf{96.22}_{\pm 0.00}$	$\mathbf{19.93}_{\pm 0.28}$	$\mathbf{39.52}_{\pm 0.80}$	$\mathbf{65.36}_{\pm 0.92}$

Two basic observations:

1. The two proposed methods show very competitive results.
2. DGF generally outperforms SGF!!!

Summarization

- We for the first time, to our knowledge, simultaneously and explicitly **model multi-view consistency and inconsistency in a unified objective function**.
- To optimize this objective function, we **present an efficient alternating minimization scheme** to obtain an approximate solution.
- **A multi-view clustering framework** based on multi-view graph learning is presented, with **two graph fusion variants**, i.e., SGF and DGF.
- Source Code: https://github.com/youweiliang/ConsistentGraphLearning

Two Interesting Issues in the Future Work

- **Consistency VS Inconsistency**
- **Similarity Fusion VS Dissimilarity Fusion**