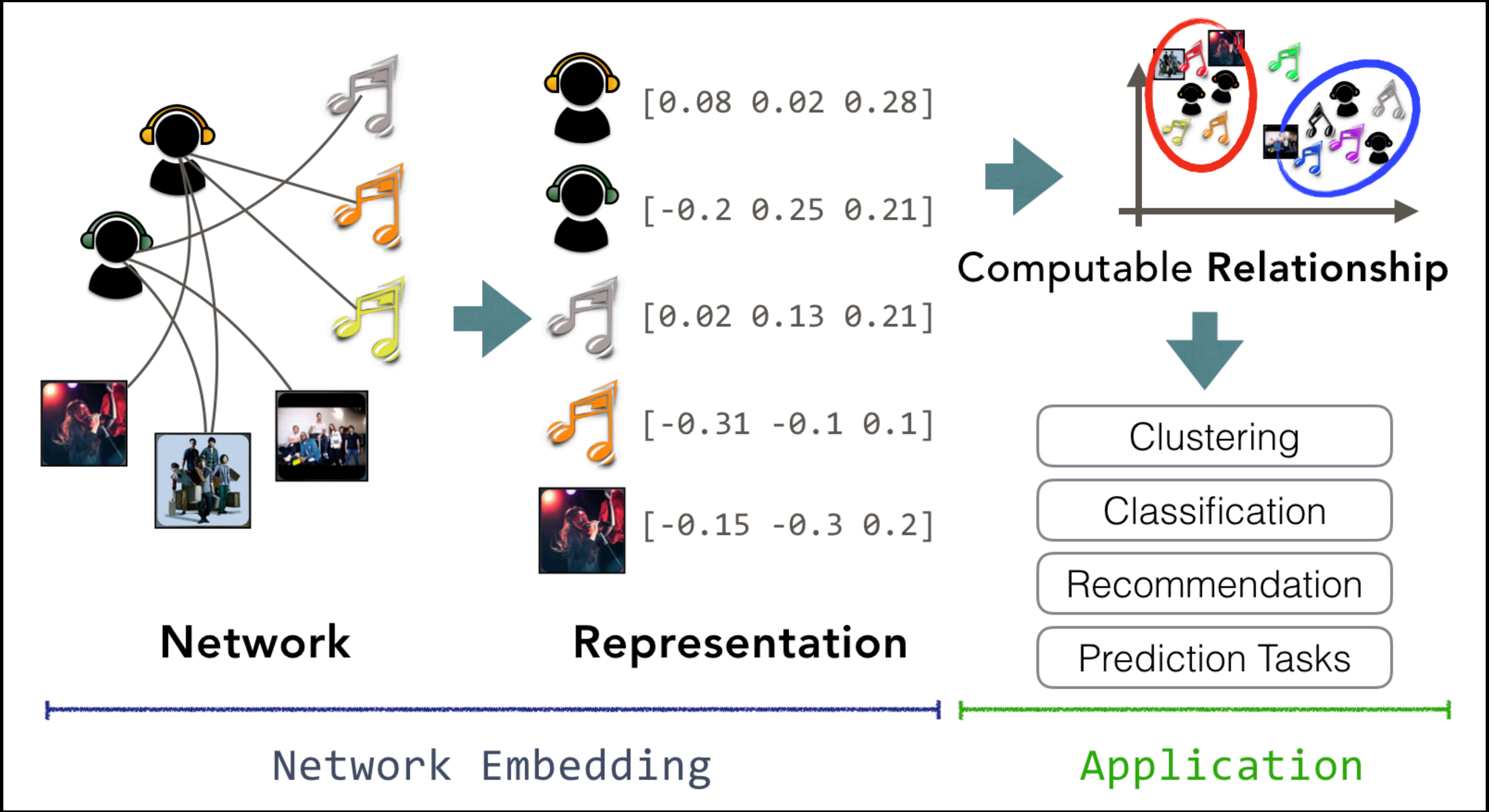
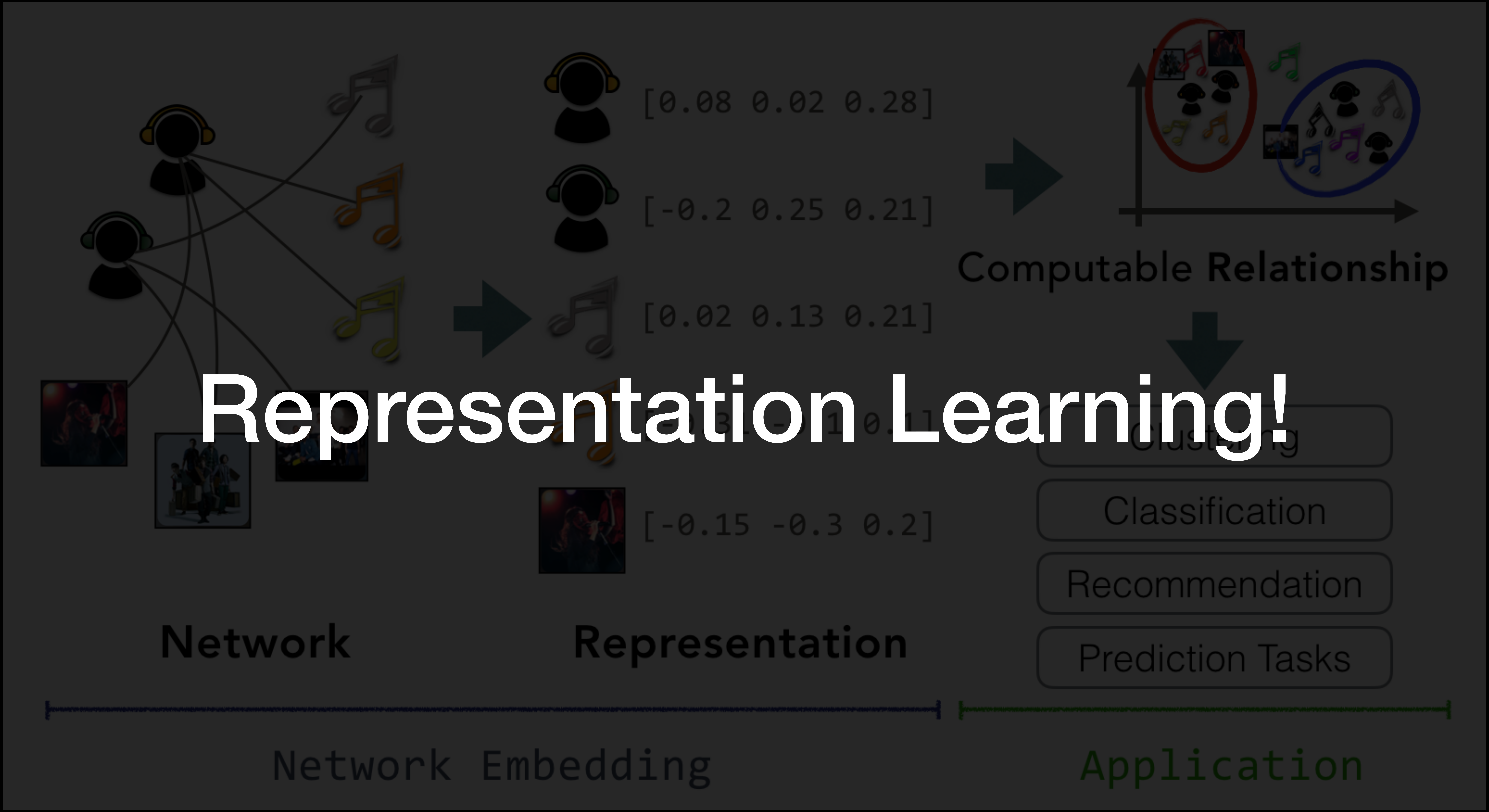
LINE: Large-scale Information Network Embedding

Sharing by Roger

Let’s recap
Network Embedding


[0.08 0.02 0.28]
[-0.2 0.25 0.21]
[0.02 0.13 0.21]
[-0.31 -0.1 0.1]
[-0.15 -0.3 0.2]
Computable Relationship
Clustering
Classification
Recommendation
Prediction Tasks
Network
Representation
Network Embedding
Application

Representation Learning!



2014

DeepWalk

Not Scalable、No Weight and Direction

↑ Solve

2015

LINE: Large-scale Information Network Embedding

Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, Qiaozhu Mei

Microsoft Research + 北京大學 PEKING UNIVERSITY

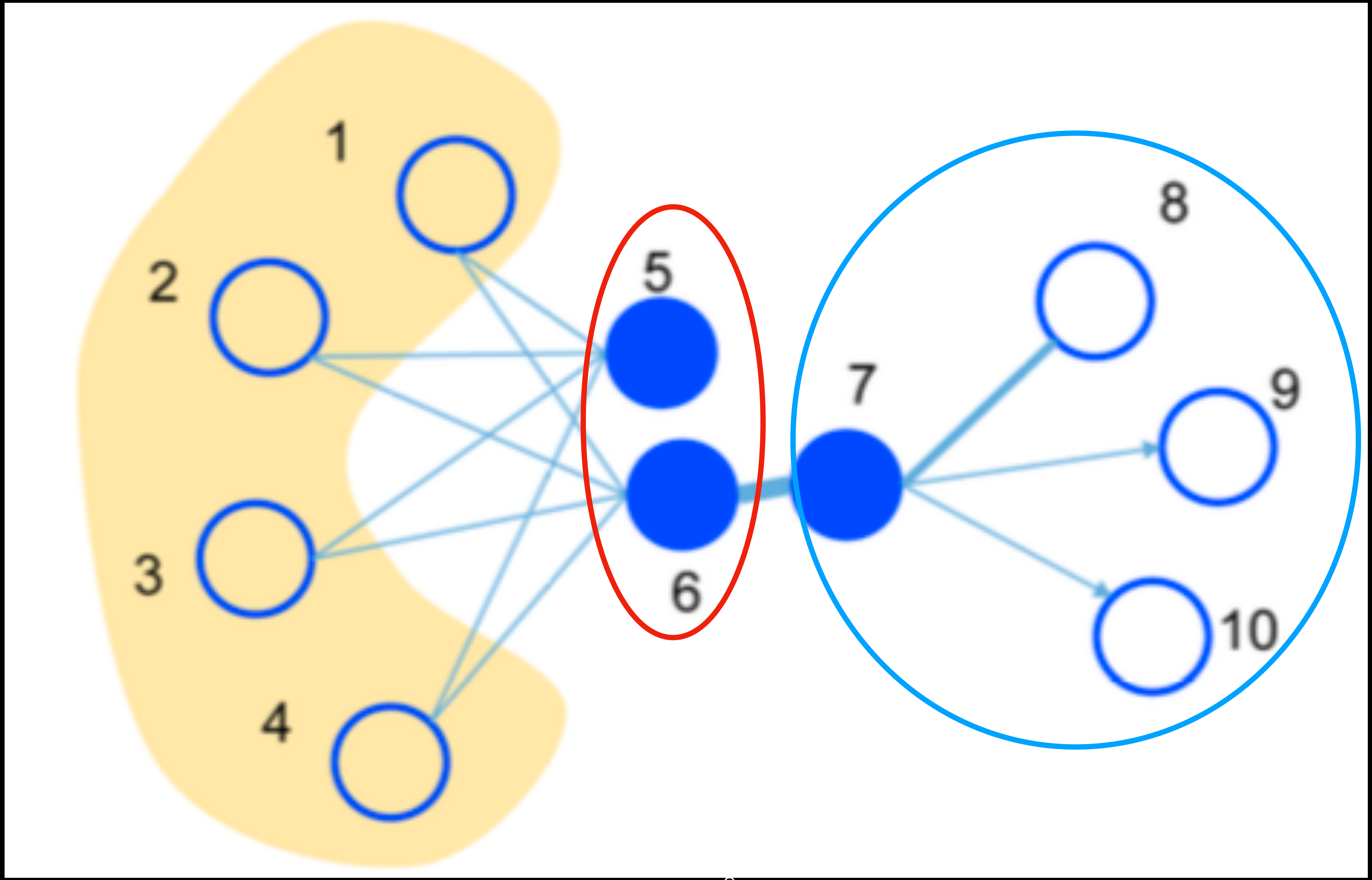
接下來

- 請了解圖學數據結構
- 請了解算法細節與特徵運用
- 請了解算法目的與降維後視覺化意義

LINE關鍵特色

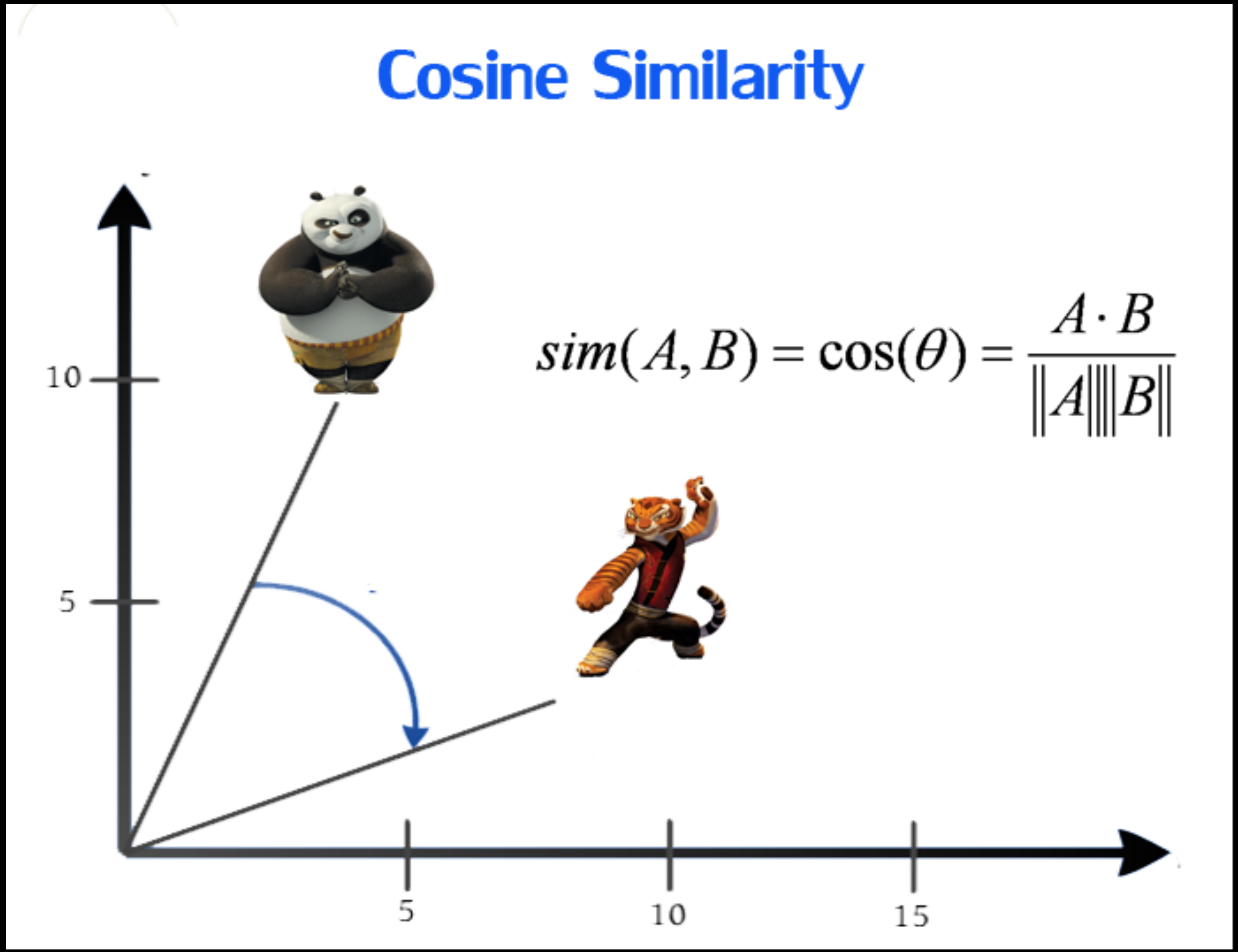
Why choose LINE?

1. 直接關係(1st) + 同義關係(2nd)

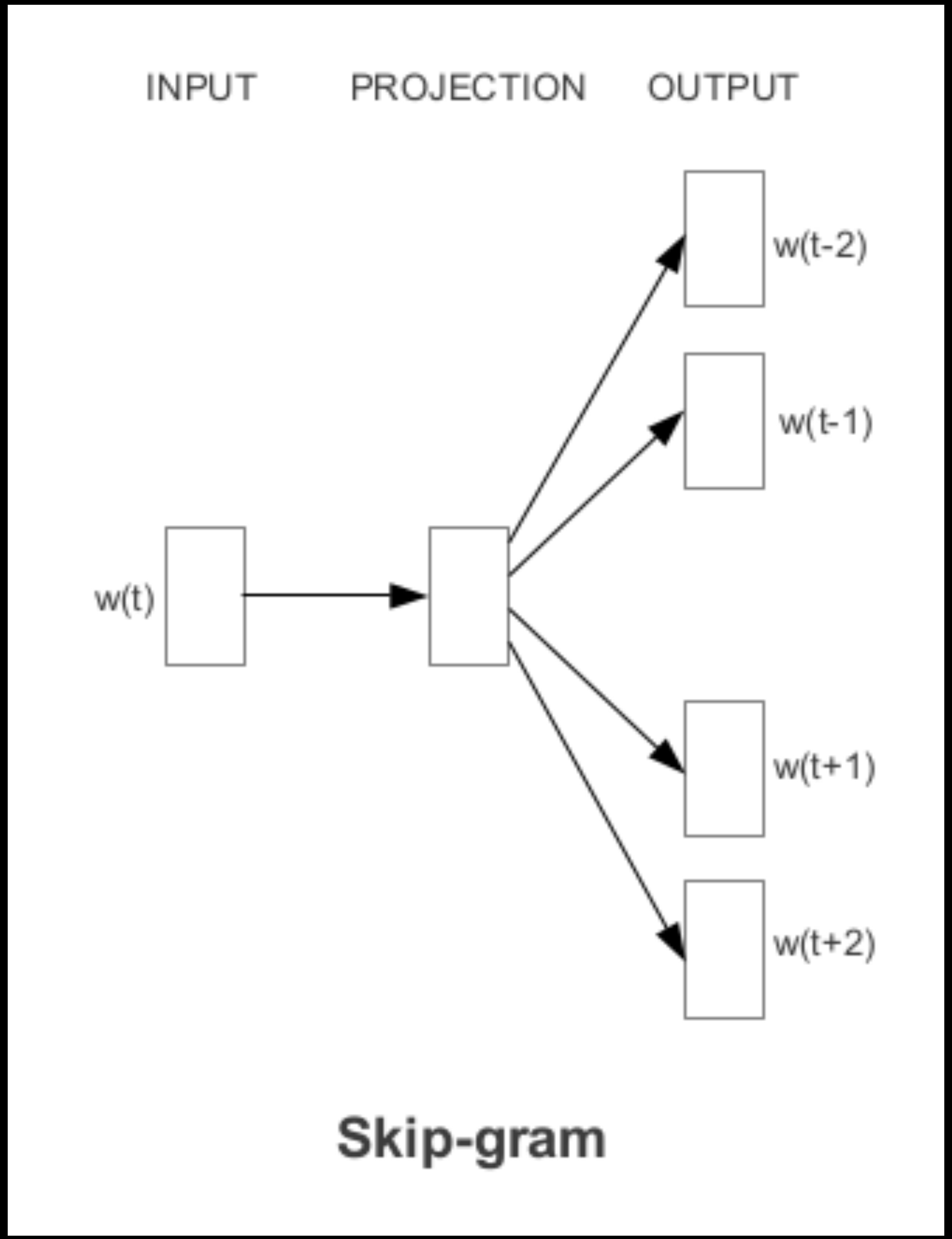


Word	Similarity	Top similar words
good	1st	luck bad faith assume nice
	2nd	decent bad excellent lousy reasonable
information	1st	provide provides detailed facts verifiable
	2nd	infomation informaiton informations nonspammy animecons
graph	1st	graphs algebraic finite symmetric topology
	2nd	graphs subgraph matroid hypergraph undirected
learn	1st	teach learned inform educate how
	2nd	learned teach relearn learnt understand

1st Order



2nd Order



2. Edge Sampling, No random walk

1st Order

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \log p_1(v_i, v_j),$$

2nd Order

$$O_2 = -\sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i).$$

3. 考慮Edge Weight與方向

1st Order

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \log p_1(v_i, v_j),$$

權重

2nd Order

$$O_2 = -\sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i).$$

權重　方向

目標問題定義

數學課

Problem Definition

Entities

V = 頂點集合

E = 有權重邊集合

G = (V, E)

→

學習將V投射d維空間

並保存

1st order

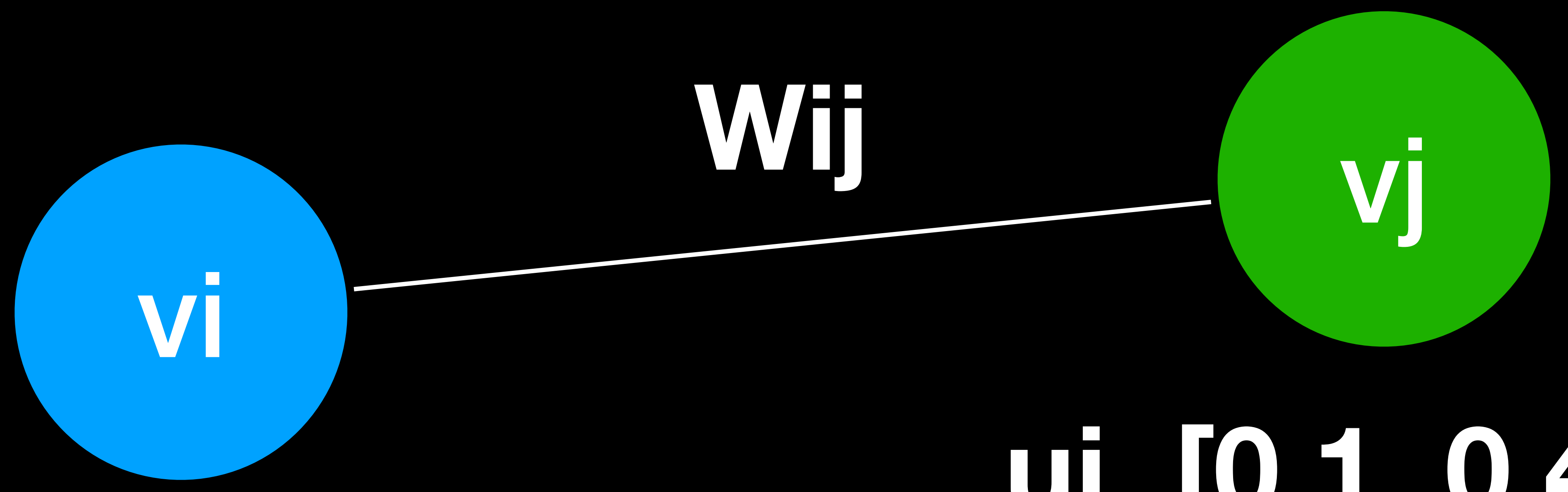
或

2nd order

關係

1st-order Proximity

無向圖限定

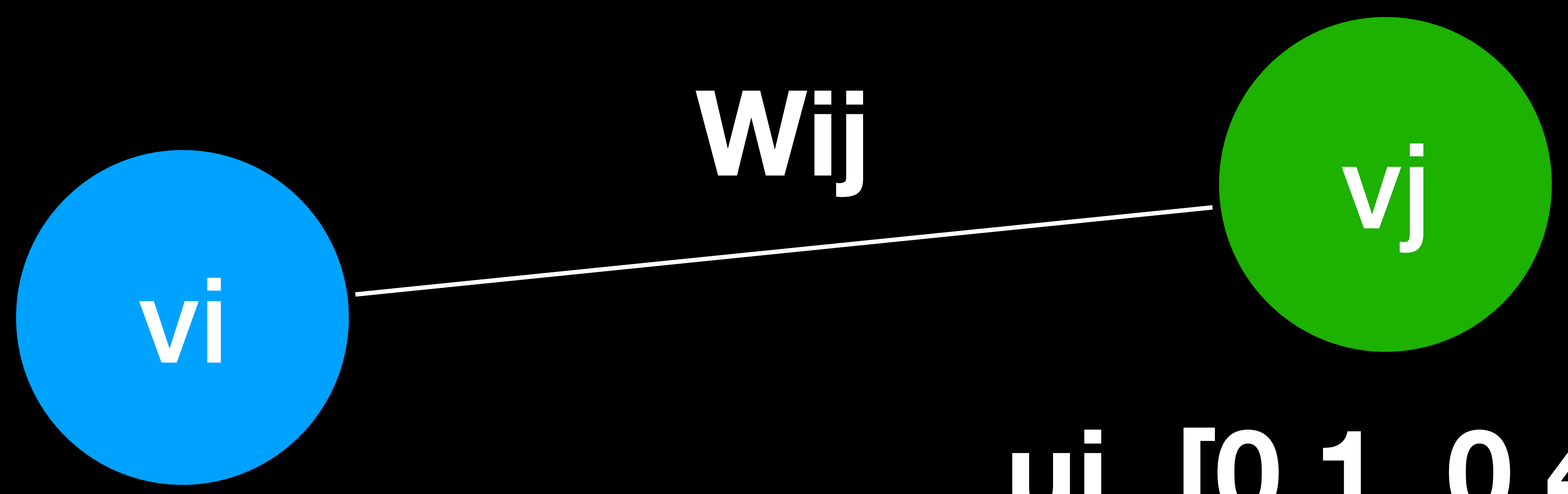


uj [0.1, 0.4, …]

ui [0.3, 0.2, …]

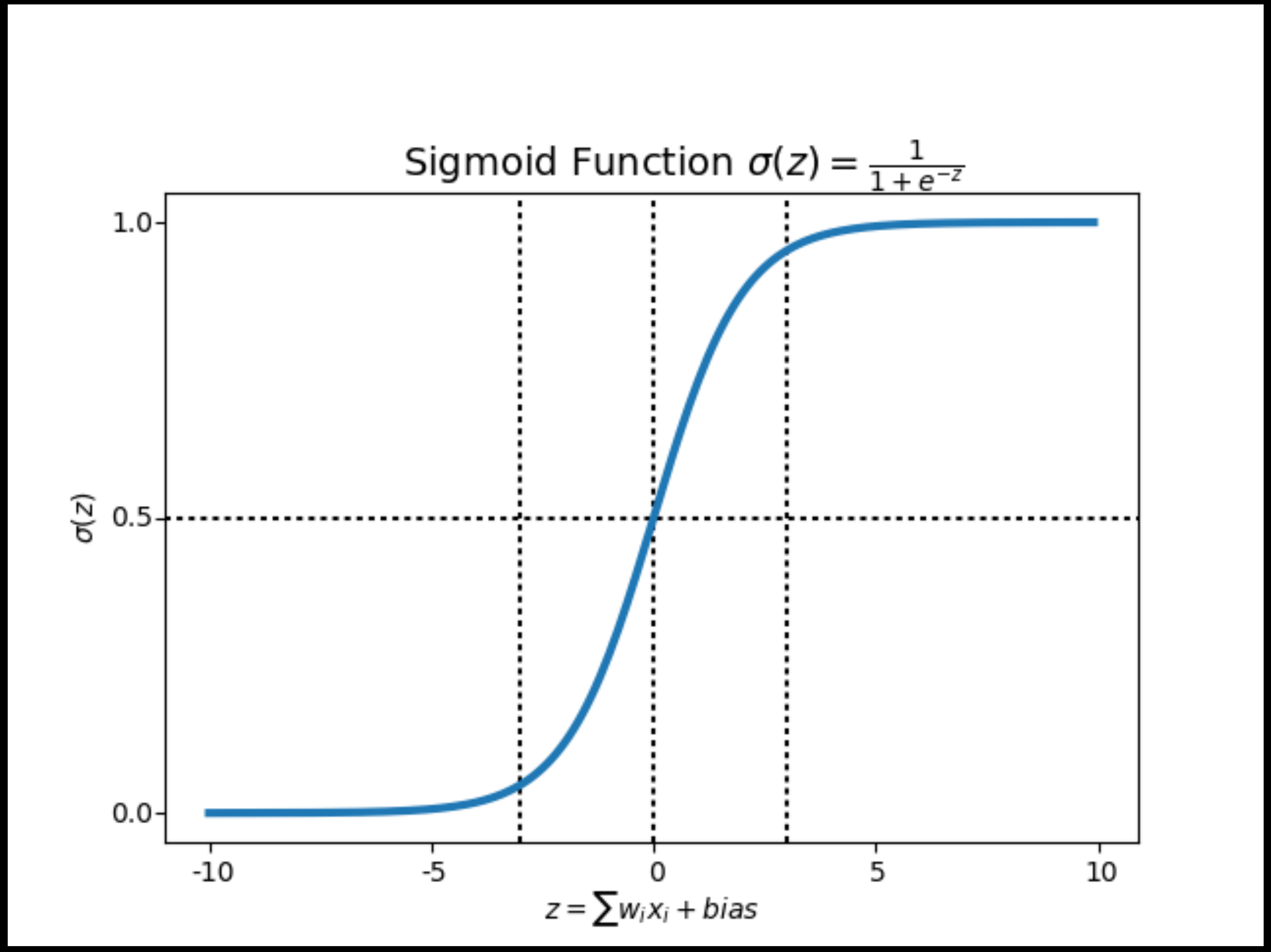
概念: vi與vj之間的向量距離正比於wij/全w合

1st-order Proximity



uj [0.1, 0.4, …]

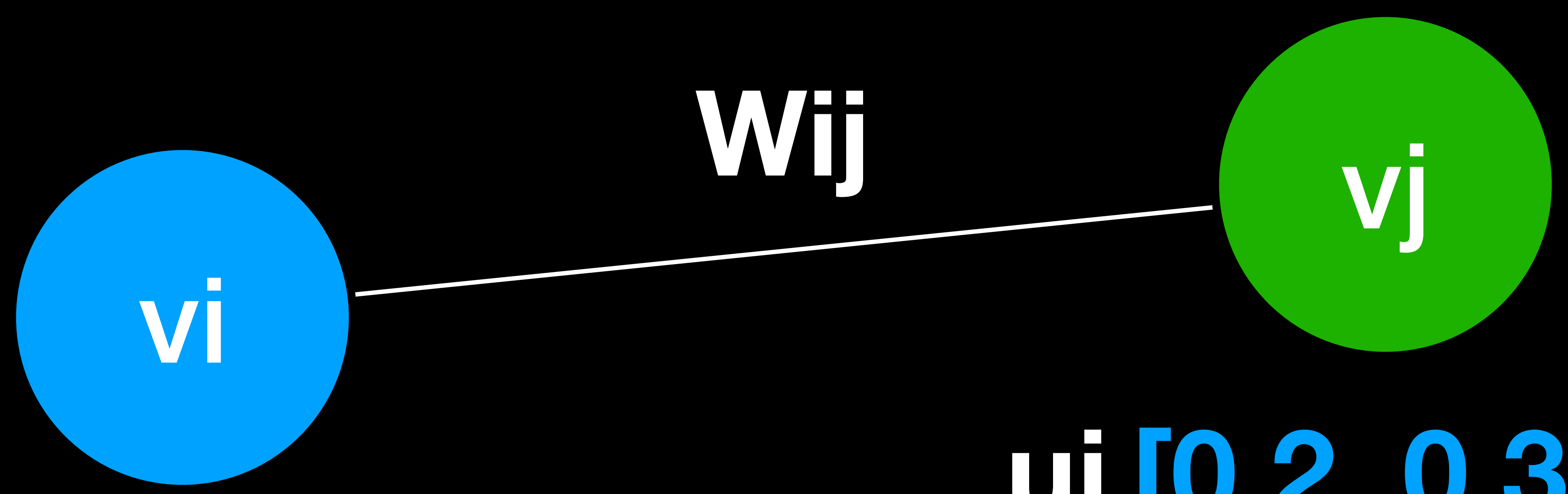
ui [0.3, 0.2, …]



$$p_1(v_i, v_j) = \frac{1}{1 + \exp(-\vec{u}_i^T \cdot \vec{u}_j)}, \quad (1)$$

被訓練機率 (sigmoid)

1st-order Proximity



uj [0.2, 0.3, …]

ui [0.4, 0.1, …]

$$p_1(v_i, v_j) = \frac{1}{1 + \exp(-\vec{u}_i^T \cdot \vec{u}_j)}, \qquad (1)$$

被訓練機率

調整u, 讓(1)逼近

$$\hat{p}_1(i, j) = \frac{w_{ij}}{W}, \text{ where } W = \sum_{(i,j)\in E} w_{ij}.$$

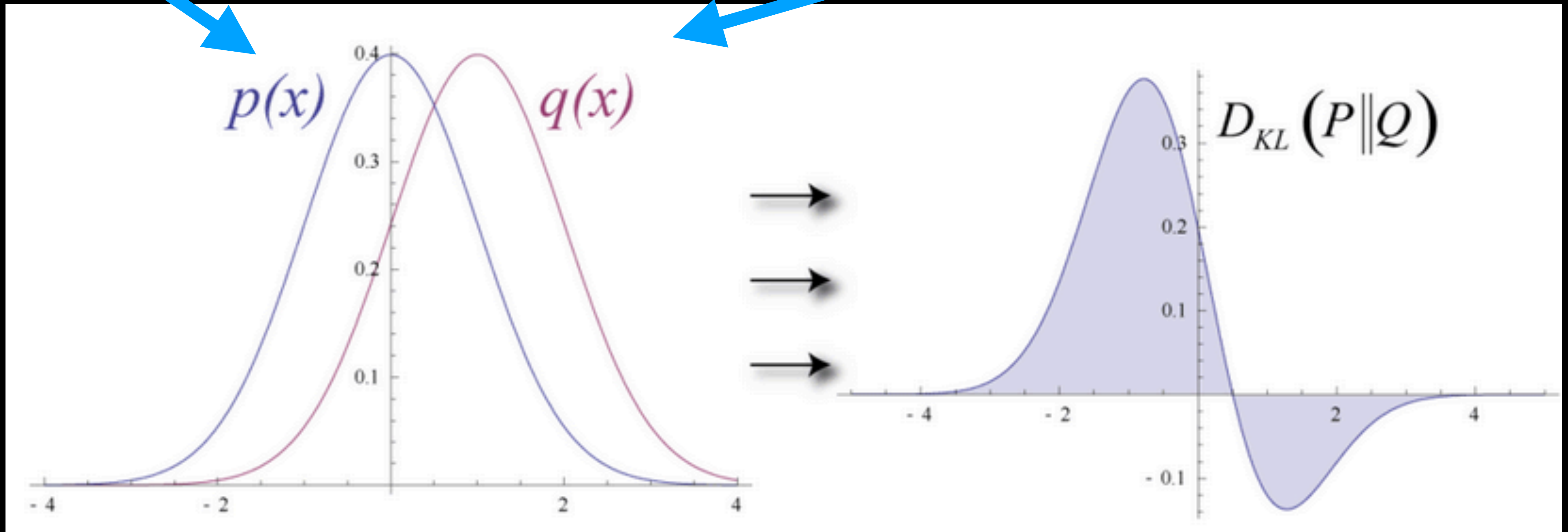
經驗機率

更新u

1st-order Proximity

$$\hat{p}_1(i,j) = \frac{w_{ij}}{W}, \text{ where } W = \sum_{(i,j)\in E} w_{ij}.$$

$$p_1(v_i, v_j) = \frac{1}{1+\exp(-\vec{u}_i^T \cdot \vec{u}_j)}, \quad (1)$$



DKL(P||Q) = $\sum p(x) log \frac{p(x)}{q(x)}$

最佳化思路: 調整u, 讓Q逼近P, 以最小化DKL(P||Q)

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \log p_1(v_i, v_j),$$

目標函示

1st-order (只用在無向圖)

$$P_1(v_i, v_j) = \sigma(\vec{u}_i \cdot \vec{u}_j)$$

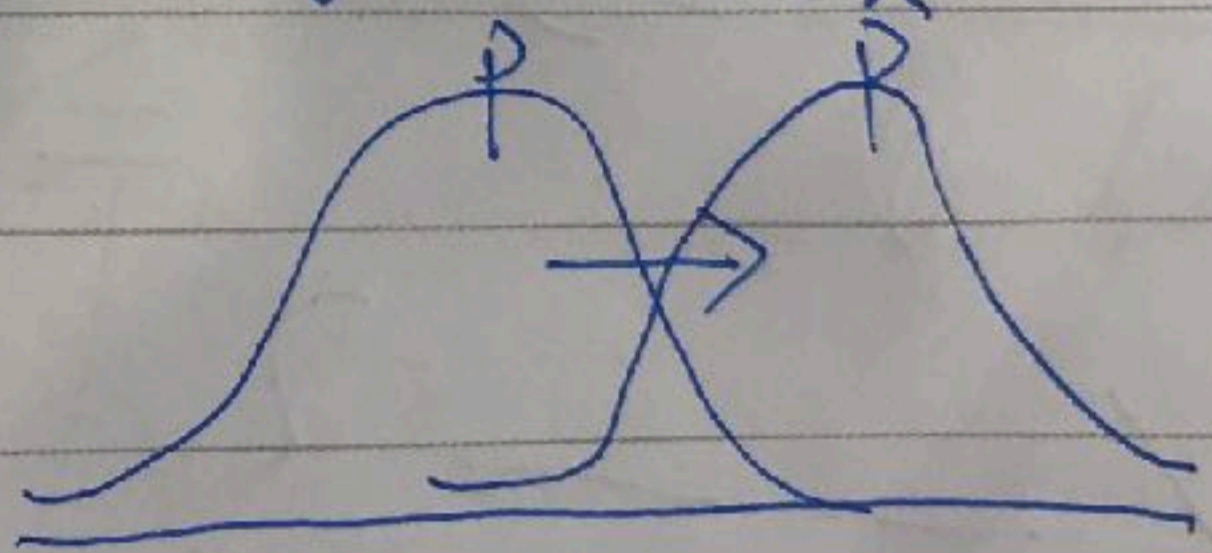
$$\hat{P}_1(v_i, v_j) = \frac{w_{ij}}{W}, \quad W = \sum_{(i,j)\in E} w_{ij}$$

$$\Rightarrow O_1 = \sum \hat{P}_1 \cdot \log \frac{\hat{P}_1}{P_1}$$

$$= \sum \frac{w_{ij}}{W} \cdot \log \frac{\frac{w_{ij}}{W}}{P_1(v_i, v_j)}$$

$$\approx -\sum_{(i,j)\in E} w_{ij} \cdot \log P_1(v_i, v_j)$$

$$\Rightarrow \min_{\vec{u}} O_1$$

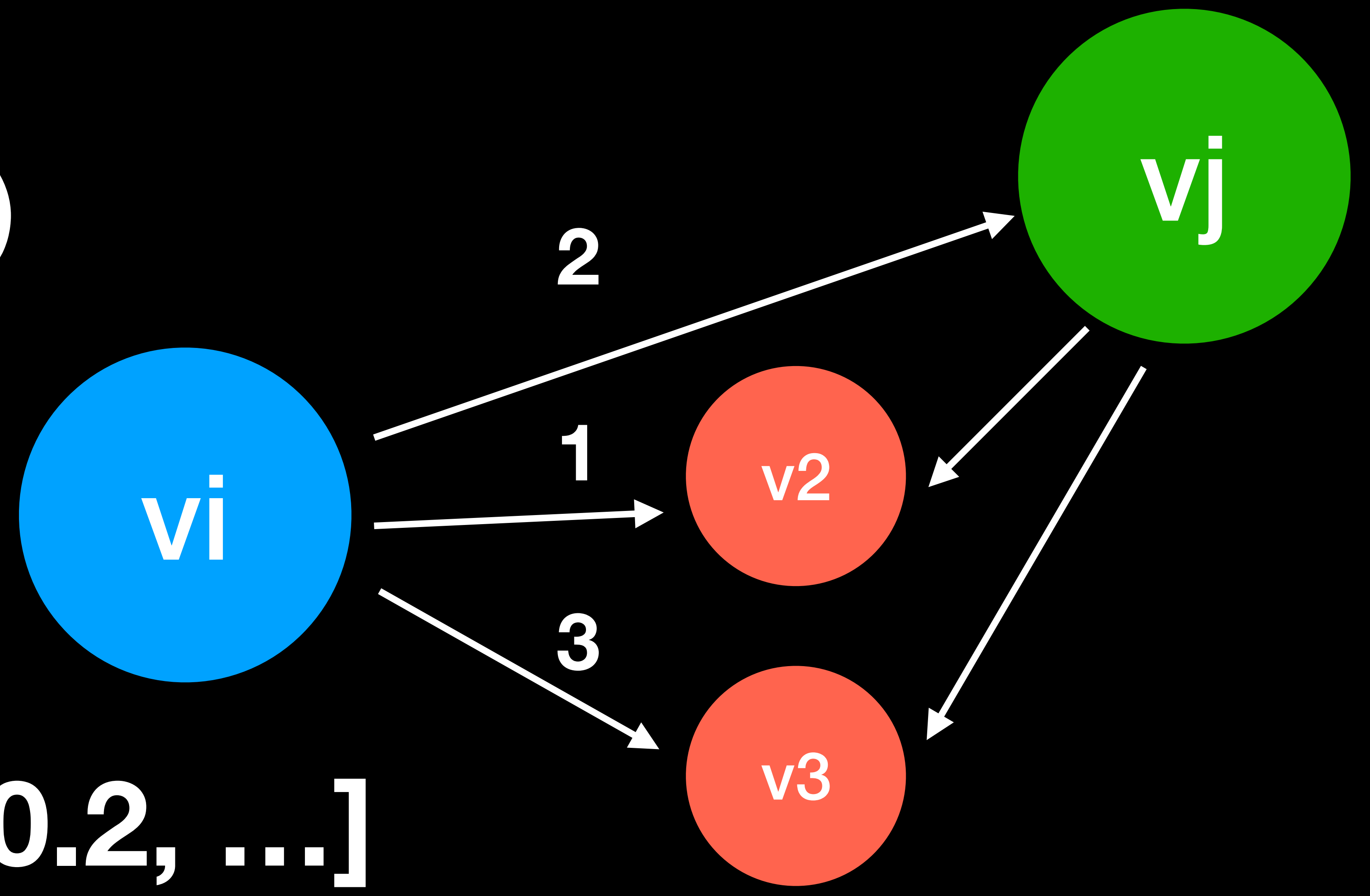


2nd-order Proximity

用在有向圖

無向圖: (u, v)=(v, u)

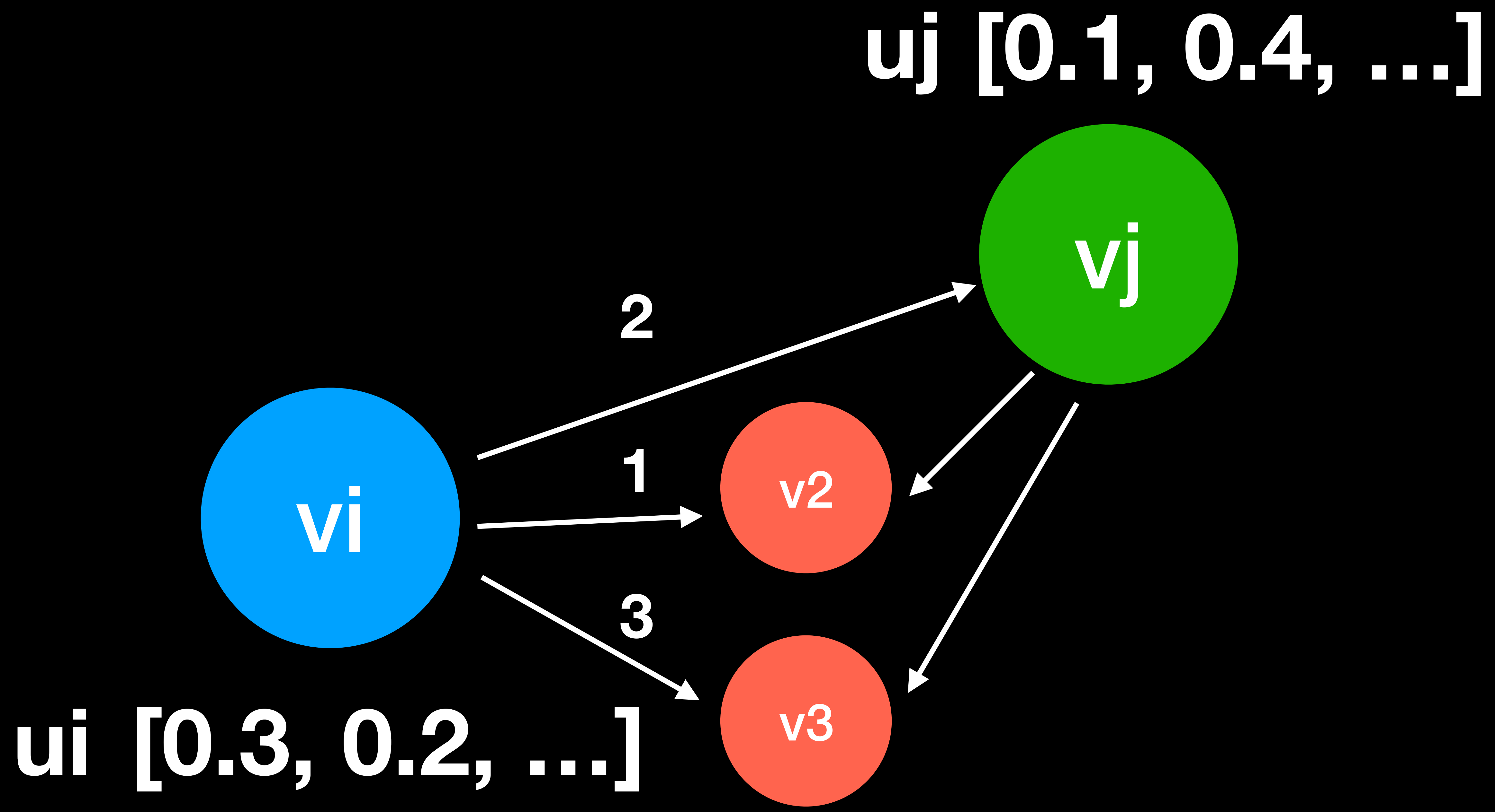
uj [0.1, 0.4, …]

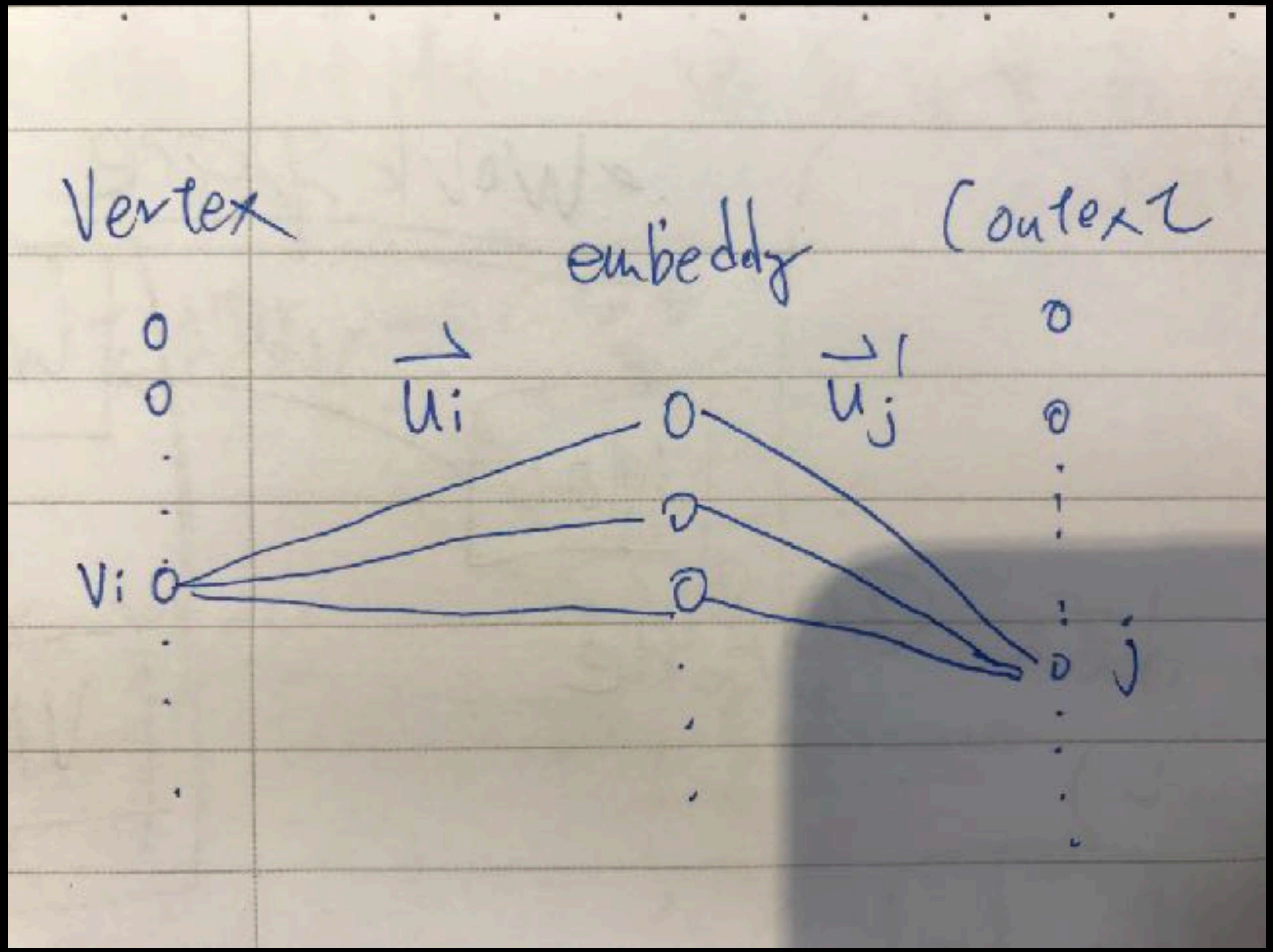


ui [0.3, 0.2, …]

概念: vi附近有vj的機率正比於wij/(i向外的權重和)

2nd-order Proximity



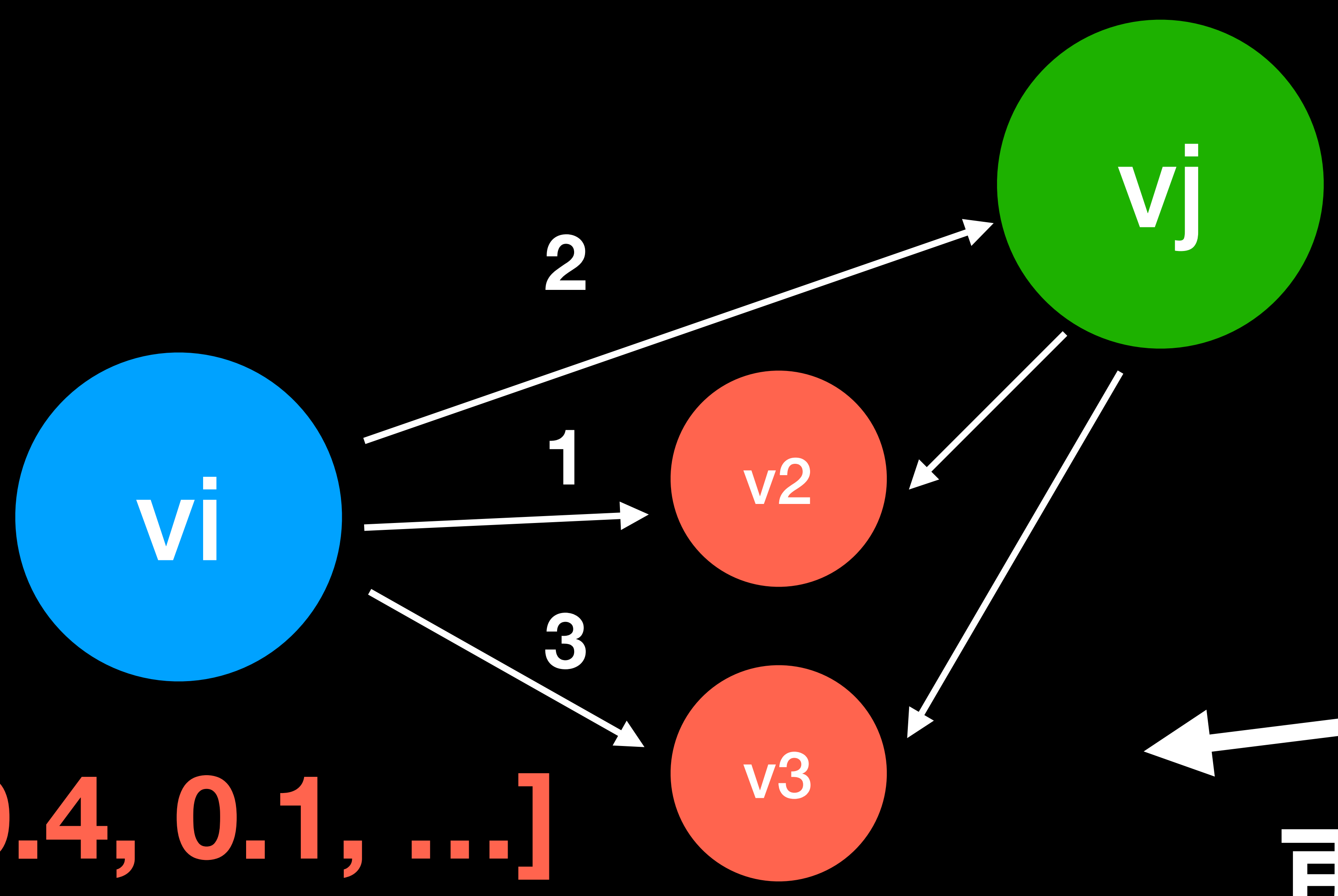


$$p_2(v_j|v_i) = \frac{\exp(\vec{u}'^T_j \cdot \vec{u}_i)}{\sum_{k=1}^{|V|} \exp(\vec{u}'^T_k \cdot \vec{u}_i)}, \tag{4}$$

被訓練機率 (softmax)

2nd-order Proximity

uj [0.2, 0.3, …]



ui [0.4, 0.1, …]

$$p_2(v_j|v_i) = \frac{\exp(\vec{u}'^T_j \cdot \vec{u}_i)}{\sum_{k=1}^{|V|} \exp(\vec{u}'^T_k \cdot \vec{u}_i)}, \quad (4)$$

被訓練機率

調整u, 讓(4)逼近

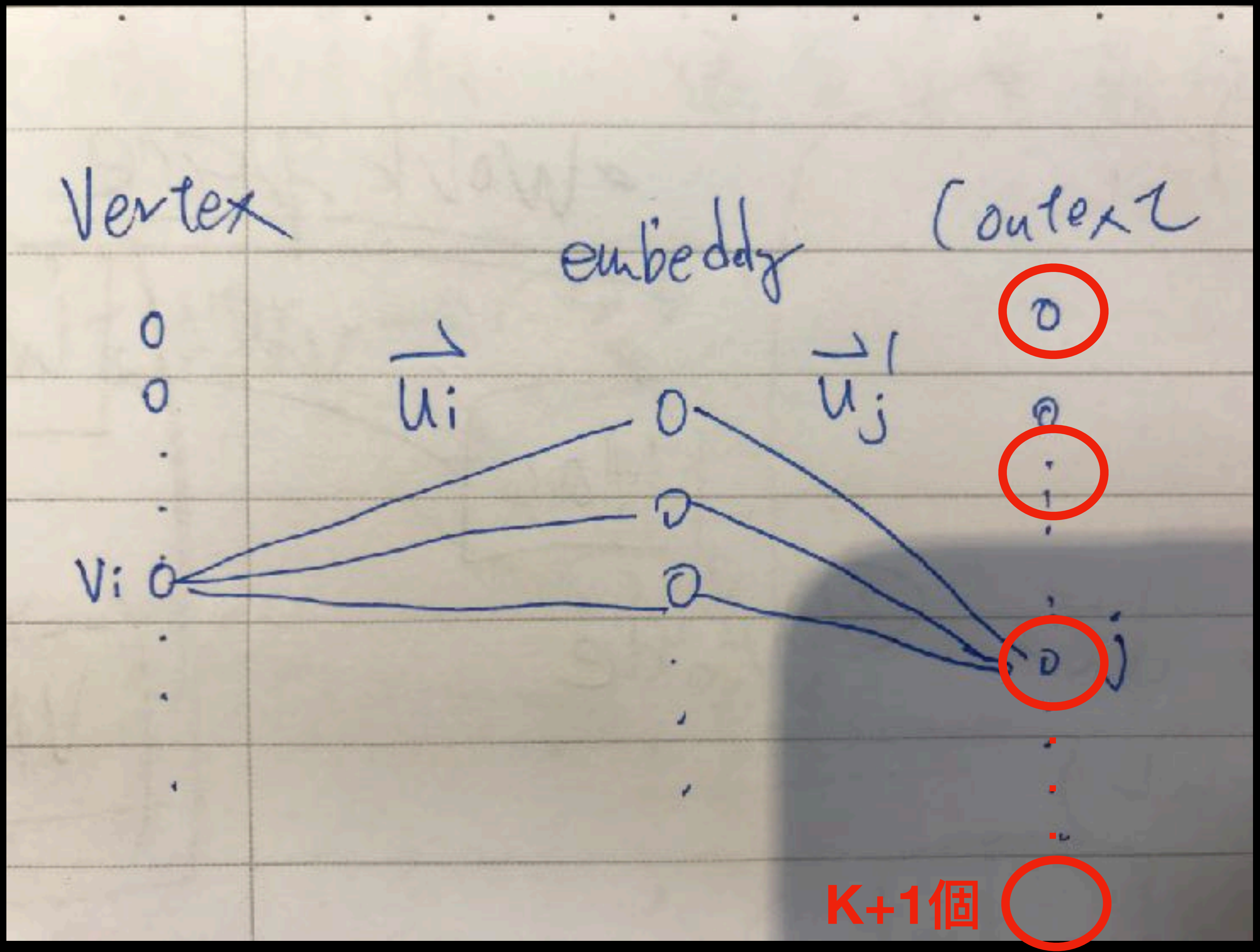
$$\hat{p}_2(v_j|v_i) = \frac{w_{ij}}{d_i}$$

經驗機率

2/(2+1+3) = 1/3

更新u

2nd-order Proximity



被訓練機率

$$p_2(v_j|v_i) = \frac{\exp(\vec{u}_j'^T \cdot \vec{u}_i)}{\sum_{k=1}^{|V|} \exp(\vec{u}_k'^T \cdot \vec{u}_i)}, \tag{4}$$

Optimize
Neg Sampling

$$\log \sigma({\vec{u}_j'}^T \cdot \vec{u}_i) + \sum_{i=1}^{K} E_{v_n \sim P_n(v)} [\log \sigma(-{\vec{u}_n'}^T \cdot \vec{u}_i)],$$

K = num of neg samples

2nd-order Proximity

$$O_2 = - \sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i).$$

目標函示

$$\frac{\partial O_2}{\partial \vec{u}_i} = w_{ij} \cdot \frac{\partial \log p_2(v_j|v_i)}{\partial \vec{u}_i}$$

目標函示對ui偏微

$$\boxed{O_2} = \sum_{i \in V} \lambda_i \, D_{KL}\left(\hat{p}_2(\cdot|v_i) \,|\, p_2(\cdot|v_i) \right)$$

λ_i: vi 的重要性 $\approx d_i$

$$= \sum_{i,j\in E} \hat{p}_2(v_j|v_i) \log \frac{\hat{p}_2(v_j|v_i)}{p_2(v_j|v_i)}$$

$$= \sum_{i,j\in E} d_i \cdot \frac{w_{ij}}{d_i} \cdot \log \frac{\cancel{\frac{w_{ij}}{d_i}}}{p_2(v_j|v_i)}$$

$$\approx -\sum \cancel{d_i} \cdot \frac{w_{ij}}{\cancel{d_i}} \cdot \log p_2(v_j|v_i) = \boxed{-\sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i)}$$

Optimization with Edge Sampling

$$\frac{\partial O_2}{\partial \vec{u}_i} = w_{ij} \cdot \frac{\partial \log p_2(v_j|v_i)}{\partial \vec{u}_i}$$

$$w_{ij} \log p_2(v_j|v_i). \longrightarrow \log p_2(v_j|v_i).$$

sample prob = wij / W

問題: w差異大, 以致梯度下降程度嚴重不均衡

實驗設計與結果

Data Sets

Table 1: Statistics of the real-world information networks.

	Word Co-occur	Friends		Co-Author	Paper -cite-> Paper
	Language Network	**Social Network**		**Citation Network**	
Name	WIKIPEDIA	FLICKR	YOUTUBE	DBLP(AUTHORCITATION)	DBLP(PAPERCITATION)
Type	undirected,weighted	undirected,binary	undirected,binary	dircted,weighted	directed,binary
$\|V\|$	1,985,098	1,715,256	1,138,499	524,061	781,109
$\|E\|$	1,000,924,086	22,613,981	2,990,443	20,580,238	4,191,677
Avg. degree	504.22	26.37	5.25	78.54	10.73
#Labels	7	5	47	7	7
#train	70,000	75,958	31,703	20,684	10,398

Multi-Label Document **Classification**

&

Network Layouts for **Visualization**

Compared Algorithm

Graph Factorization (GF) : **Matrix Factorization on Network Affinity Matrix**

DeepWalk : **Random Walk + SkipGram, No Weight & Directions**

LINE-SGD : $w_{ij} \log p_2(v_j|v_i).$

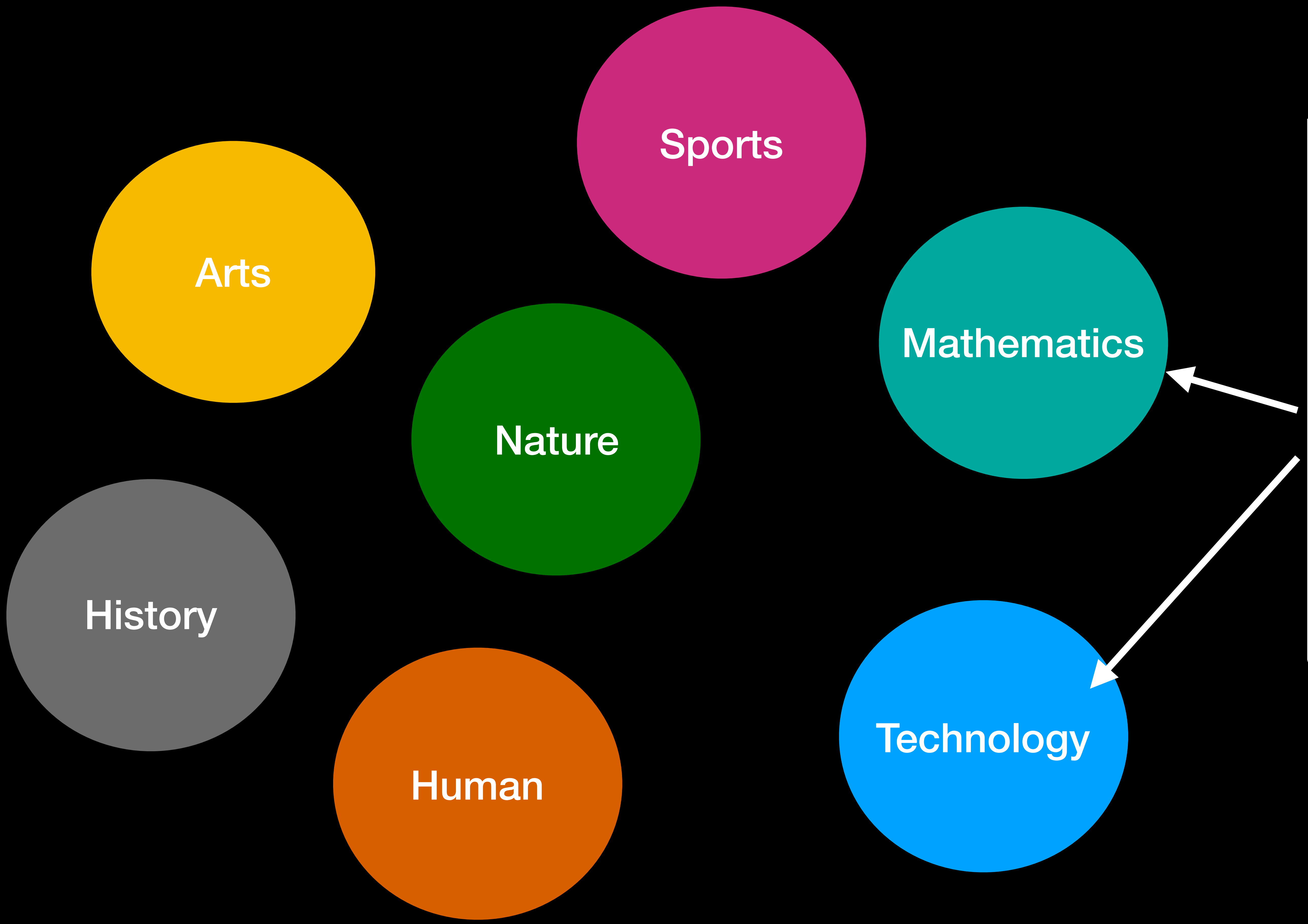
LINE : $\log p_2(v_j|v_i).$ **sample prob = wij / W**

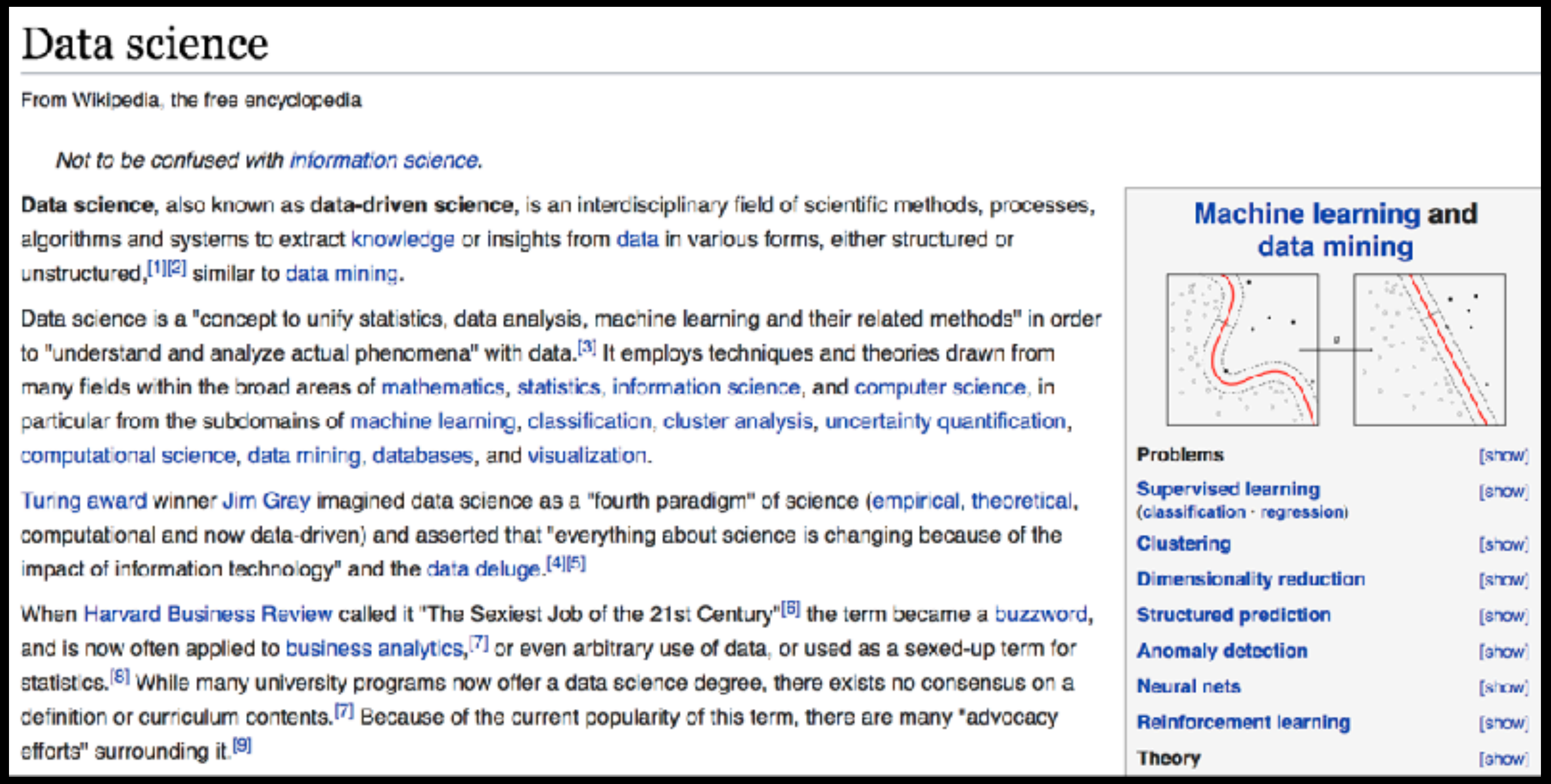
LINE (1st + 2nd) : **1 concat 2, Only applicable in supervised task**

Language Network

Wikipedia

Multi-Label Document Classification





Data science

From Wikipedia, the free encyclopedia

Not to be confused with information science.

Data science, also known as **data-driven science**, is an interdisciplinary field of scientific methods, processes, algorithms and systems to extract knowledge or insights from data in various forms, either structured or unstructured,[1][2] similar to data mining.

Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" in order to "understand and analyze actual phenomena" with data.[3] It employs techniques and theories drawn from many fields within the broad areas of mathematics, statistics, information science, and computer science, in particular from the subdomains of machine learning, classification, cluster analysis, uncertainty quantification, computational science, data mining, databases, and visualization.

Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.[4][5]

When Harvard Business Review called it "The Sexiest Job of the 21st Century"[6] the term became a buzzword, and is now often applied to business analytics,[7] or even arbitrary use of data, or used as a sexed-up term for statistics.[8] While many university programs now offer a data science degree, there exists no consensus on a definition or curriculum contents.[7] Because of the current popularity of this term, there are many "advocacy efforts" surrounding it.[9]

Machine learning and data mining

Problems	[show]
Supervised learning (classification · regression)	[show]
Clustering	[show]
Dimensionality reduction	[show]
Structured prediction	[show]
Anomaly detection	[show]
Neural nets	[show]
Reinforcement learning	[show]
Theory	[show]

Multi-Label Document Classification

Wikipedia Docs

Sample X% of Docs as training data

Representation learning

Word Vectors

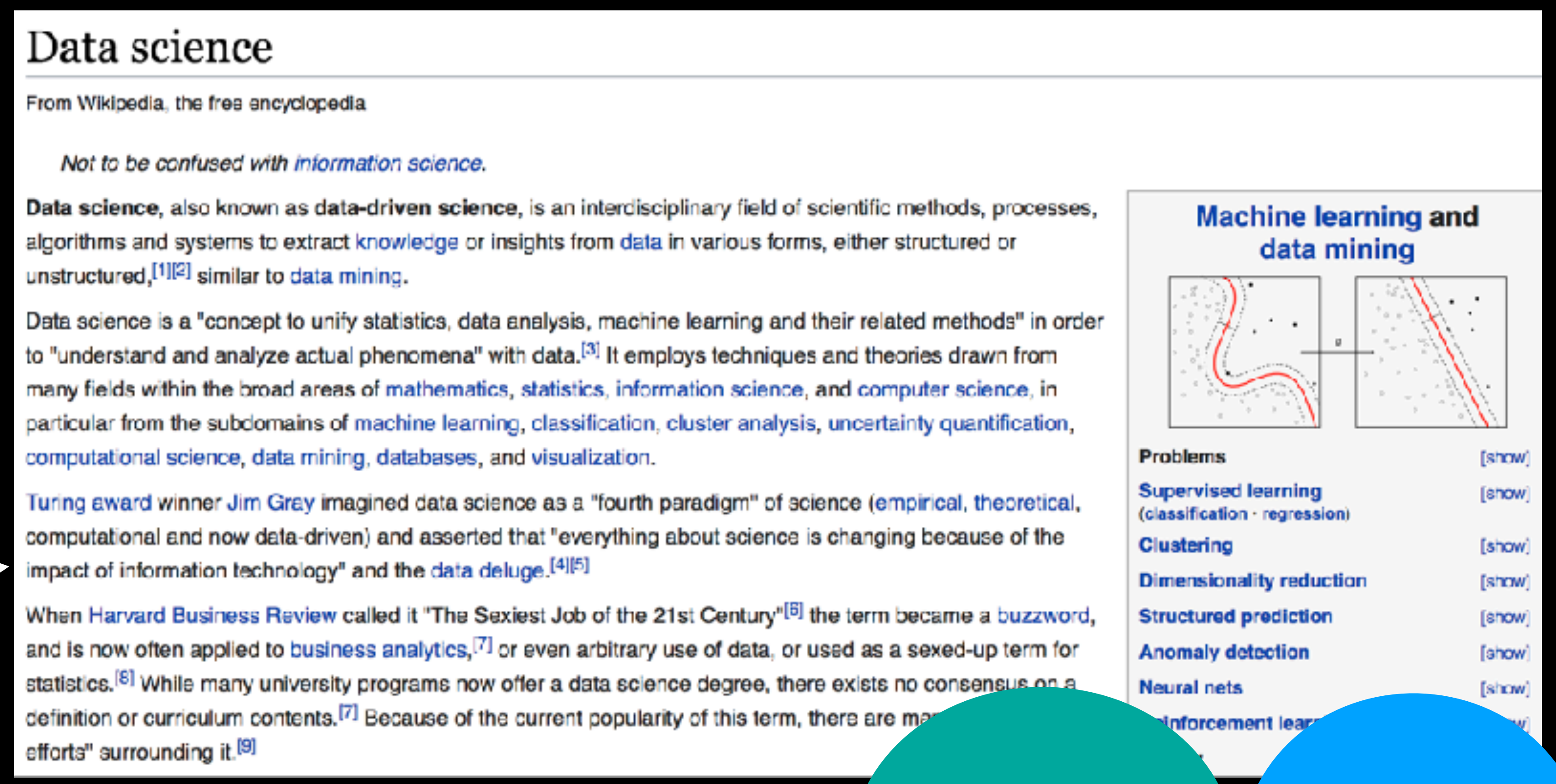
Average WV for every Docs

Document Vectors

Train One-vs-Rest logistic Regression

Multi-Label Classifier

Classify every label for testing docs



Data science

From Wikipedia, the free encyclopedia

Not to be confused with information science.

Data science, also known as **data-driven science**, is an interdisciplinary field of scientific methods, processes, algorithms and systems to extract knowledge or insights from data in various forms, either structured or unstructured,[1][2] similar to data mining.

Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" in order to "understand and analyze actual phenomena" with data.[3] It employs techniques and theories drawn from many fields within the broad areas of mathematics, statistics, information science, and computer science, in particular from the subdomains of machine learning, classification, cluster analysis, uncertainty quantification, computational science, data mining, databases, and visualization.

Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.[4][5]

When Harvard Business Review called it "The Sexiest Job of the 21st Century"[6] the term became a buzzword, and is now often applied to business analytics,[7] or even arbitrary use of data, or used as a sexed-up term for statistics.[8] While many university programs now offer a data science degree, there exists no consensus on a definition or curriculum contents.[7] Because of the current popularity of this term, there are ma efforts" surrounding it.[9]

Machine learning and data mining

Problems [show]

Supervised learning (classification · regression) [show]

Clustering [show]

Dimensionality reduction [show]

Structured prediction [show]

Anomaly detection [show]

Neural nets [show]

Mathematics

Technology

Word Vectors Similarity

Word	Similarity	Top similar words
good	1st	luck bad faith assume nice
	2nd	decent bad excellent lousy reasonable
information	1st	provide provides detailed facts verifiable
	2nd	infomation informaiton informations nonspammy animecons
graph	1st	graphs algebraic finite symmetric topology
	2nd	graphs subgraph matroid hypergraph undirected
learn	1st	teach learned inform educate how
	2nd	learned teach relearn learnt understand

Classification Result

Table 3: Results of Wikipedia page classification on WIKIPEDIA data set.

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	GF	79.63	80.51	80.94	81.18	81.38	81.54	81.63	81.71	81.78
	DeepWalk	78.89	79.92	80.41	80.69	80.92	81.08	81.21	81.35	81.42
	SkipGram	79.84	80.82	81.28	81.57	81.71	81.87	81.98	82.05	82.09
	LINE-SGD(1st)	76.03	77.05	77.57	77.85	78.08	78.25	78.39	78.44	78.49
	LINE-SGD(2nd)	74.68	76.53	77.54	78.18	78.63	78.96	79.19	79.40	79.57
	LINE(1st)	79.67	80.55	80.94	81.24	81.40	81.52	81.61	81.69	81.67
	LINE(2nd)	79.93	80.90	81.31	81.63	81.80	81.91	82.00	82.11	82.17
	LINE(1st+2nd)	**81.04****	**82.08****	**82.58****	**82.93****	**83.16****	**83.37****	**83.52****	**83.63****	**83.74****
Macro-F1	GF	79.49	80.39	80.82	81.08	81.26	81.40	81.52	81.61	81.68
	DeepWalk	78.78	79.78	80.30	80.56	80.82	80.97	81.11	81.24	81.32
	SkipGram	79.74	80.71	81.15	81.46	81.63	81.78	81.88	81.98	82.01
	LINE-SGD(1st)	75.85	76.90	77.40	77.71	77.94	78.12	78.24	78.29	78.36
	LINE-SGD(2nd)	74.70	76.45	77.43	78.09	78.53	78.83	79.08	79.29	79.46
	LINE(1st)	79.54	80.44	80.82	81.13	81.29	81.43	81.51	81.60	81.59
	LINE(2nd)	79.82	80.81	81.22	81.52	81.71	81.82	81.92	82.00	82.07
	LINE(1st+2nd)	**80.94****	**81.99****	**82.49****	**82.83****	**83.07****	**83.29****	**83.42****	**83.55****	**83.66****

Significantly outperforms GF at the: ** 0.01 and * 0.05 level, paired t-test.

LINE (1+2) is the Best!

LINE SGD is bad

LINE 2nd > 1st

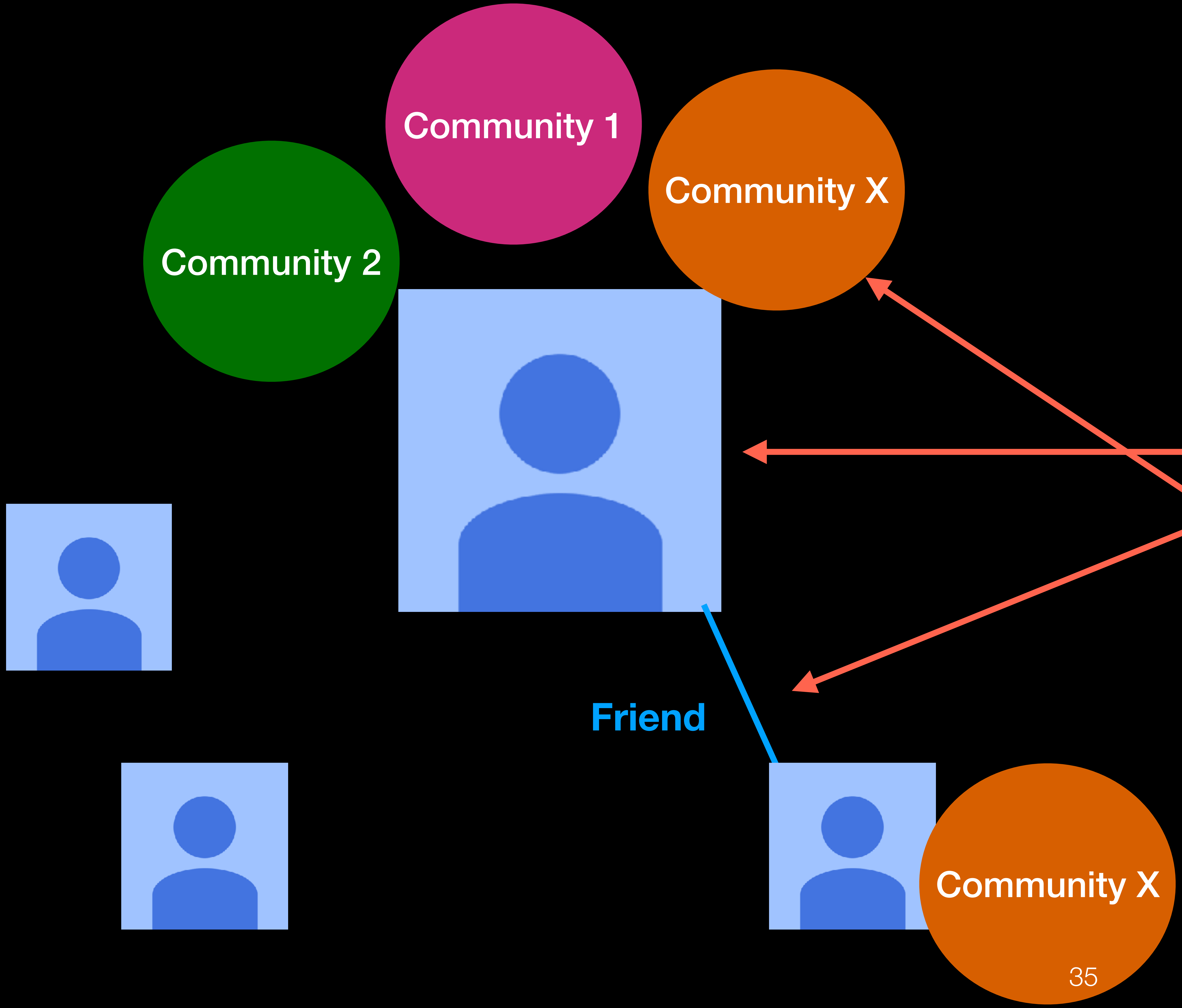
GF > Deep Walk

F1: http://sofasofa.io/forum_main_post.php?postid=1001112

Social Network

Flickr & Youtube

User would be in multiple communities



	Social Network	
Name	FLICKR	YOUTUBE
Type	undirected,binary	undirected,binary
$\|V\|$	1,715,256	1,138,499
$\|E\|$	22,613,981	2,990,443
Avg. degree	26.37	5.25
#Labels	5	47
#train	75,958	31,703

Sparse!

Table 5: Results of multi-label classification on the FLICKR network.

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	GF	53.23	53.68	53.98	54.14	54.32	54.38	54.43	54.50	54.48
	DeepWalk	60.38	60.77	60.90	61.05	61.13	61.18	61.19	61.29	61.22
	DeepWalk(256dim)	60.41	61.09	61.35	61.52	61.69	61.76	61.80	61.91	61.83
	LINE(1st)	63.27	63.69	63.82	63.92	63.96	64.03	64.06	64.17	64.10
	LINE(2nd)	62.83	63.24	63.34	63.44	63.55	63.55	63.59	63.66	63.69
	LINE(1st+2nd)	**63.20****	**63.97****	**64.25****	**64.39****	**64.53****	**64.55****	**64.61****	**64.75****	**64.74****
Macro-F1	GF	48.66	48.73	48.84	48.91	49.03	49.03	49.07	49.08	49.02
	DeepWalk	58.60	58.93	59.04	59.18	59.26	59.29	59.28	59.39	59.30
	DeepWalk(256dim)	59.00	59.59	59.80	59.94	60.09	60.17	60.18	60.27	60.18
	LINE(1st)	62.14	62.53	62.64	62.74	62.78	62.82	62.86	62.96	62.89
	LINE(2nd)	61.46	61.82	61.92	62.02	62.13	62.12	62.17	62.23	62.25
	LINE(1st+2nd)	**62.23****	**62.95****	**63.20****	**63.35****	**63.48****	**63.48****	**63.55****	**63.69****	**63.68****

Significantly outperforms DeepWalk at the: ** 0.01 and * 0.05 level, paired t-test.

LINE > DeepWalk

LINE (1+2) is the Best!

LINE 1st > 2nd

YouTube

DeepWalk > LINE

Table 6: Results of multi-label classification on the YOUTUBE network. The results in the brackets are on the reconstruct network, which adds second-order neighbors (i.e., neighbors of neighbors) as neighbors for vertices with a low degree.

Metric	Algorithm	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1	GF	25.43 (24.97)	26.16 (26.48)	26.60 (27.25)	26.91 (27.87)	27.32 (28.31)	27.61 (28.68)	27.88 (29.01)	28.13 (29.21)	28.30 (29.36)	28.51 (29.63)
	DeepWalk	39.68	41.78	42.78	43.55	43.96	44.31	44.61	44.89	45.06	45.23
	DeepWalk(256dim)	39.94	42.17	43.19	44.05	44.47	44.84	45.17	45.43	45.65	45.81
	LINE(1st)	35.43 (36.47)	38.08 (38.87)	39.33 (40.01)	40.21 (40.85)	40.77 (41.33)	41.24 (41.73)	41.53 (42.05)	41.89 (42.34)	42.07 (42.57)	42.21 (42.73)
	LINE(2nd)	32.98 (36.78)	36.70 (40.37)	38.93 (42.10)	40.26 (43.25)	41.08 (43.90)	41.79 (44.44)	42.28 (44.83)	42.70 (45.18)	43.04 (45.50)	43.34 (45.67)
	LINE(1st+2nd)	39.01* **(40.20)**	41.89 **(42.70)**	43.14 **(43.94**)**	44.04 **(44.71**)**	44.62 **(45.19**)**	45.06 **(45.55**)**	45.34 **(45.87**)**	45.69** **(46.15**)**	45.91** **(46.33**)**	46.08** **(46.43**)**
Macro-F1	GF	7.38 (11.01)	8.44 (13.55)	9.35 (14.93)	9.80 (15.90)	10.38 (16.45)	10.79 (16.93)	11.21 (17.38)	11.55 (17.64)	11.81 (17.80)	12.08 (18.09)
	DeepWalk	28.39	30.96	32.28	33.43	33.92	34.32	34.83	35.27	35.54	35.86
	DeepWalk (256dim)	28.95	31.79	33.16	34.42	34.93	35.44	35.99	36.41	36.78	37.11
	LINE(1st)	28.74 (29.40)	31.24 (31.75)	32.26 (32.74)	33.05 (33.41)	33.30 (33.70)	33.60 (33.99)	33.86 (34.26)	34.18 (34.52)	34.33 (34.77)	34.44 (34.92)
	LINE(2nd)	17.06 (22.18)	21.73 (27.25)	25.28 (29.87)	27.36 (31.88)	28.50 (32.86)	29.59 (33.73)	30.43 (34.50)	31.14 (35.15)	31.81 (35.76)	32.32 (36.19)
	LINE(1st+2nd)	**29.85** (29.24)	31.93 **(33.16**)**	33.96 **(35.08**)**	35.46** **(36.45**)**	36.25** **(37.14**)**	36.90** **(37.69**)**	37.48** **(38.30**)**	38.10** **(38.80**)**	38.46** **(39.15**)**	38.82** **(39.40**)**

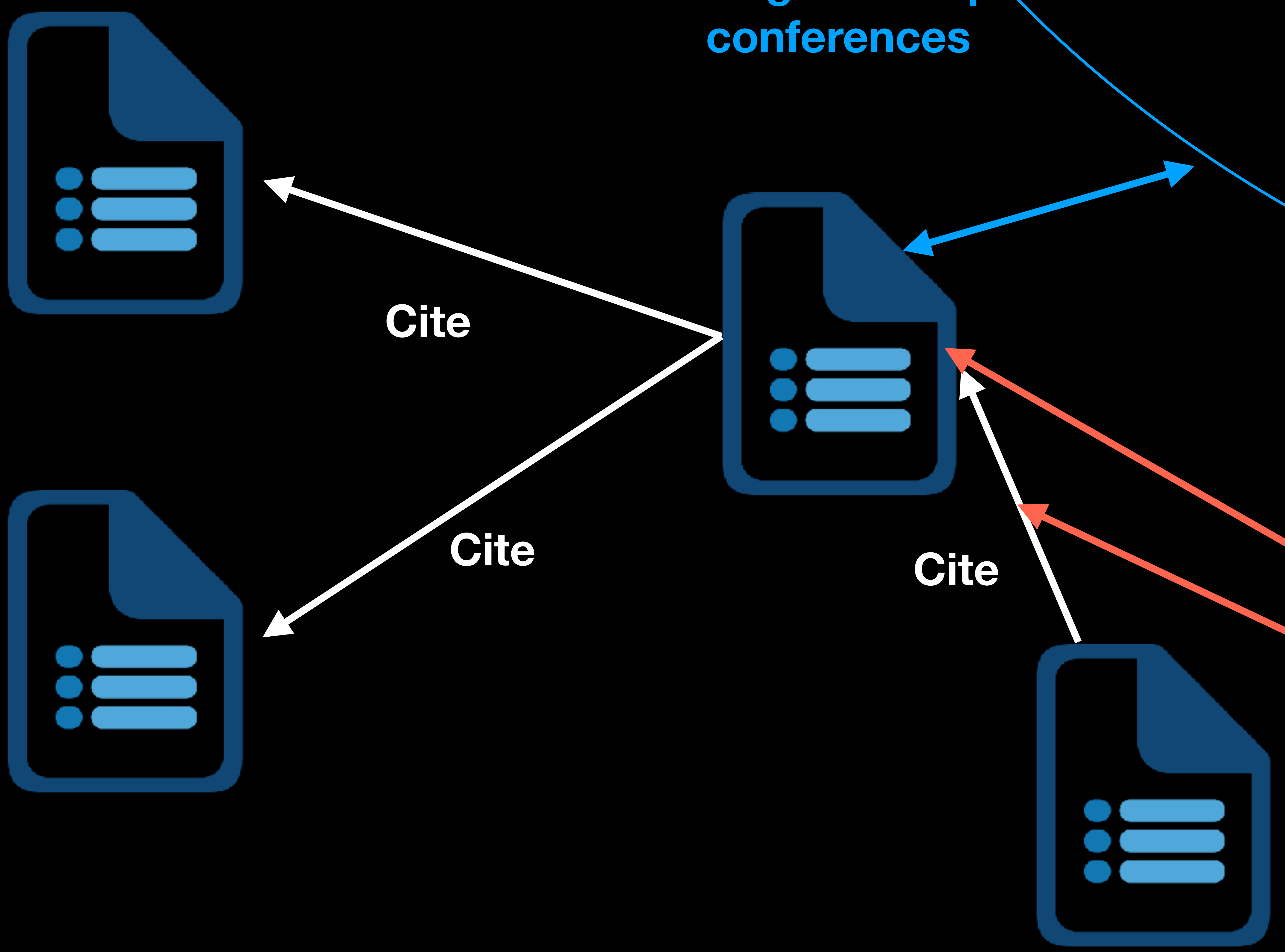
Significantly outperforms DeepWalk at the: ** 0.01 and * 0.05 level, paired t-test.

LINE在太Sparse狀況下表現不好, 要特別處理

Citation Network

DBLP Paper Citation & Co-Author

Paper Citation Network



Name	DBLP(PaperCitation)
Type	directed,binary
$\|V\|$	781,109
$\|E\|$	4,191,677
Avg. degree	10.73
#Labels	7
#train	10,398

Table 8: Results of multi-label classification on DBLP(PaperCitation) network.

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DeepWalk	52.83	53.80	54.24	54.75	55.07	55.13	55.48	55.42	55.90
	LINE(2nd)	58.42	59.58	60.29	60.78	60.94	61.20	61.39	61.39	61.79
		(**60.10****)	(**61.06****)	(**61.46****)	(**61.73****)	(**61.85****)	(**62.10****)	(**62.21****)	(**62.25****)	(**62.80****)
Macro-F1	DeepWalk	43.74	44.85	45.34	45.85	46.20	46.25	46.51	46.36	46.73
	LINE(2nd)	48.74	50.10	50.84	51.31	51.61	51.77	51.94	51.89	52.16
		(**50.22****)	(**51.41****)	(**51.92****)	(**52.20****)	(**52.40****)	(**52.59****)	(**52.78****)	(**52.70****)	(**53.02****)

Significantly outperforms DeepWalk at the: ** 0.01 and * 0.05 level, paired t-test.

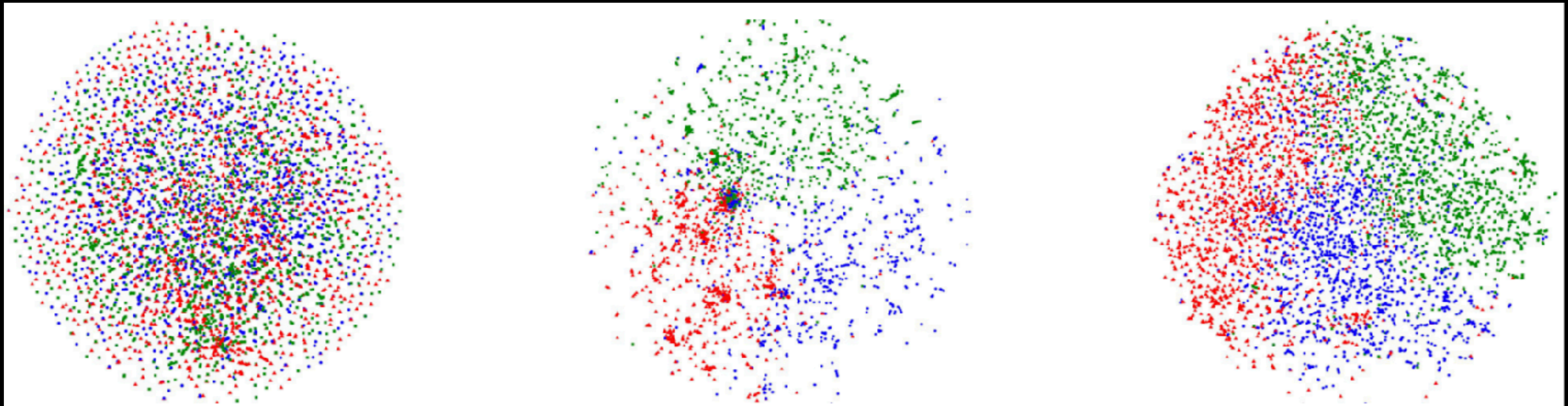
LINE 2nd > DeepWalk

Visualize Co-Author Network

Co-Author Network → Authors Embedding → TSNE Plot

Color of node = community of the author

Data Mining Machine Learning Computer Vision



(a) GF (b) DeepWalk (c) LINE(2nd)

結論與反思

LINE優點

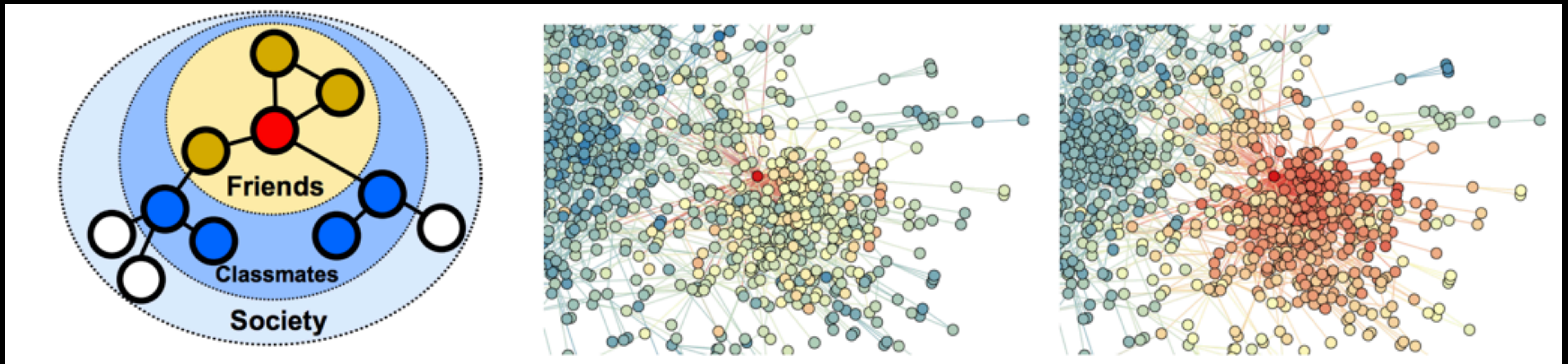
- 同時考慮1st Order & 2nd Order
- 不做Random Walk, 做Edge Sampling, 提升效率
- 適用於有權重、有方向邊

LINE缺點

- Vertex的degree過低時或太Sparse, 不夠準確
- 沒有Random Walk, 隱藏關係不容易學到

其他方法

WALKLETS

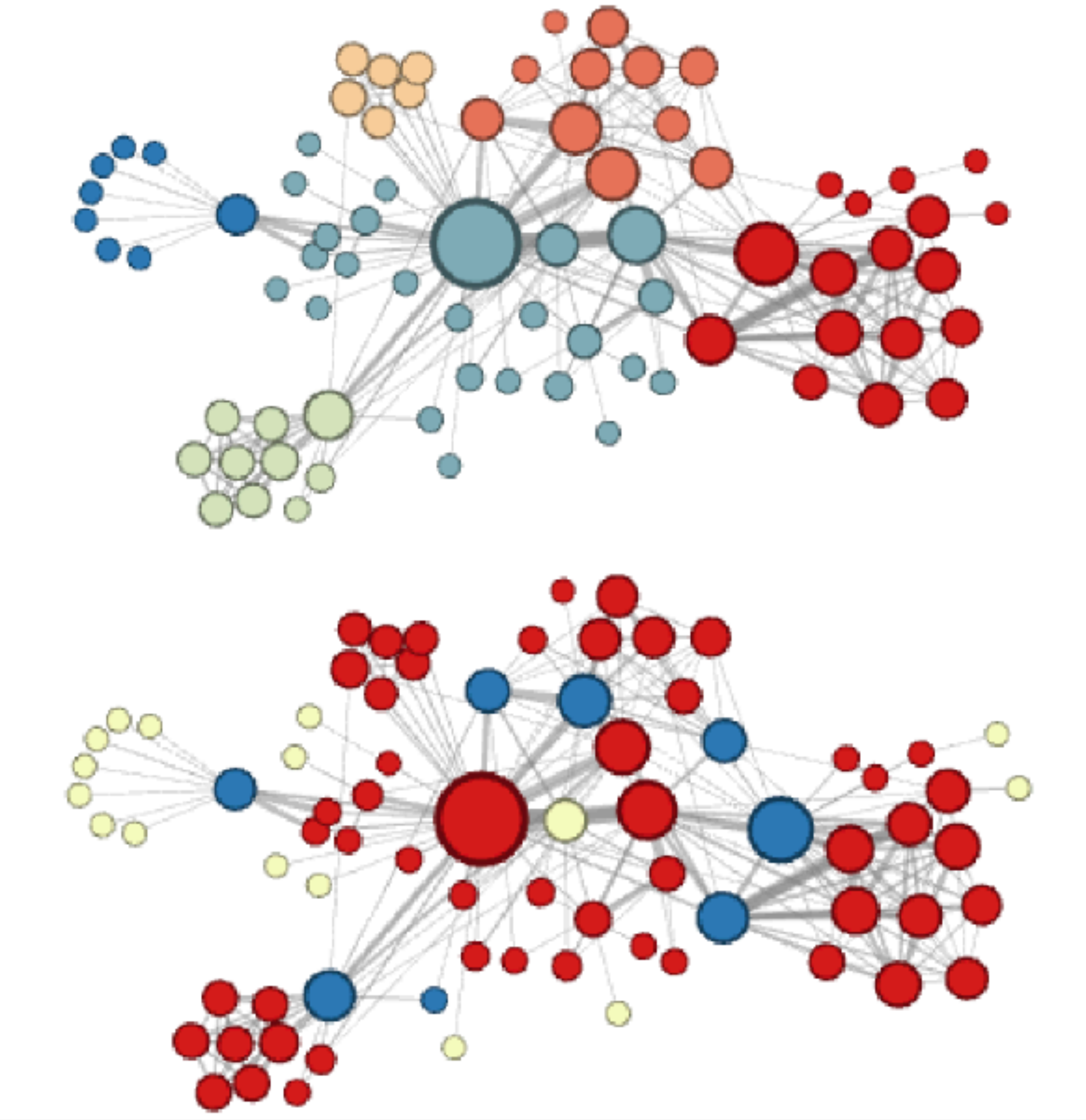


(a) A student (in red) is a member of several increasing larger social communities.

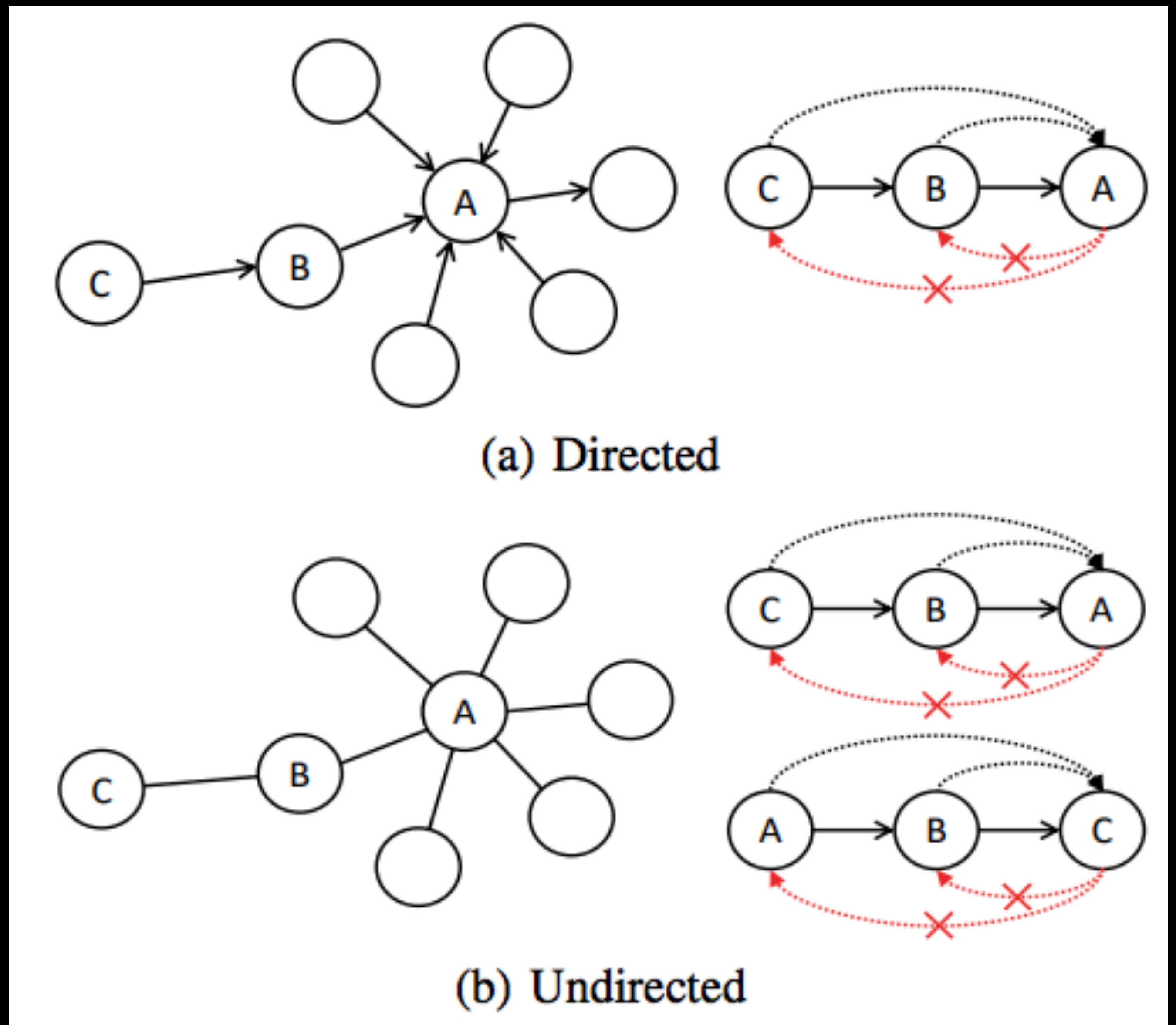
(b) WALKLETS Fine Representation

(c) WALKLETS Coarse Representation

node2vec



APP



(a) Directed

(b) Undirected

proNet-Core

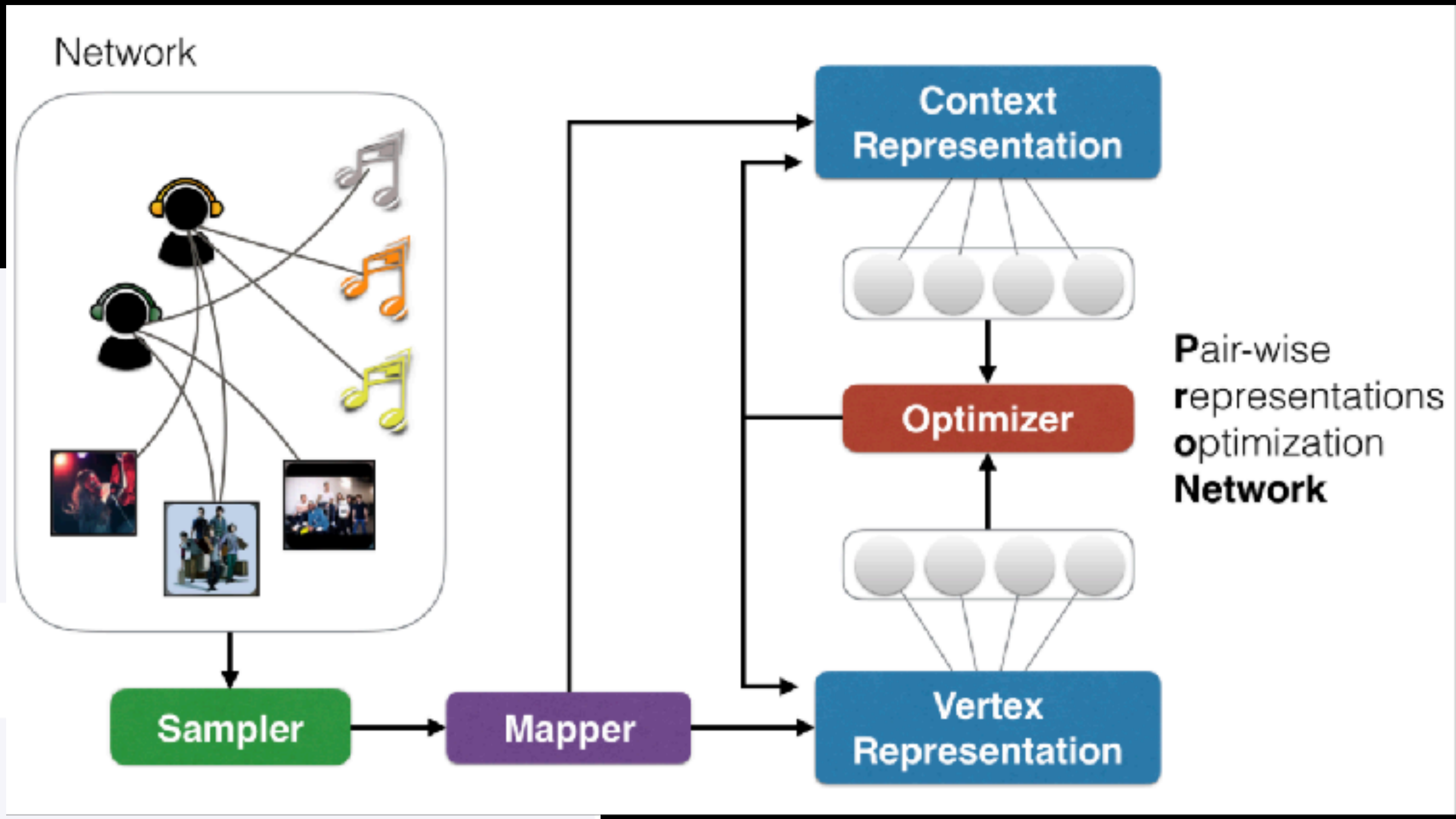
Given a network input:

```
userA itemA 3
userA itemC 5
userB itemA 1
userB itemB 5
userC itemA 4
```

```
./cli/deepwalk
./cli/walklets
./cli/line
./cli/hpe
./cli/app
./cli/mf
./cli/bpr
```

hen you will see the options description like:

```
Options Description:
        -train <string>
                Train the Network data
        -save <string>
                Save the representation data
        -dimensions <int>
                Dimension of vertex representation; default is 64
        -undirected <int>
                Whether the edge is undirected; default is 1
        -negative_samples <int>
                Number of negative examples; default is 5
        -window_size <int>
                Size of skip-gram window; default is 5
        -walk_times <int>
                Times of being staring vertex; default is 10
        -walk_steps <int>
                Step of random walk; default is 40
        -threads <int>
                Number of training threads; default is 1
        -alpha <float>
                Init learning rate; default is 0.025
Usage:
./deepwalk -train net.txt -save rep.txt -undirected 1 -dimensions
```



The model learns the representations of each vertex:

```
6 5
userA 0.0815412 0.0205459 0.288714 0.296497 0.394043
itemA -0.207083 -0.258583 0.233185 0.0959801 0.258183
itemC 0.0185886 0.138003 0.213609 0.276383 0.45732
userB -0.0137994 -0.227462 0.103224 -0.456051 0.389858
itemB -0.317921 -0.163652 0.103891 -0.449869 0.318225
userC -0.156576 -0.3505 0.213454 0.10476 0.259673
```

https://github.com/cnclabs/proNet-core

Vertex-Context Sampling for Weighted Network Embedding

Chih-Ming Chen*
Department of Computer Science,
National Chengchi University
Taipei, Taiwan
104761501@nccu.edu.tw

Ming-Feng Tsai
Department of Computer Science,
National Chengchi University
Taipei, Taiwan
mftsai@nccu.edu.tw

Yian Chen
Research Center, KKBOX Inc.
Taipei, Taiwan
annchen@kkbox.com

Yi-Hsuan Yang
Research Center for Information Technology Innovation
Taipei, Taiwan
yang@citi.sinica.edu.tw

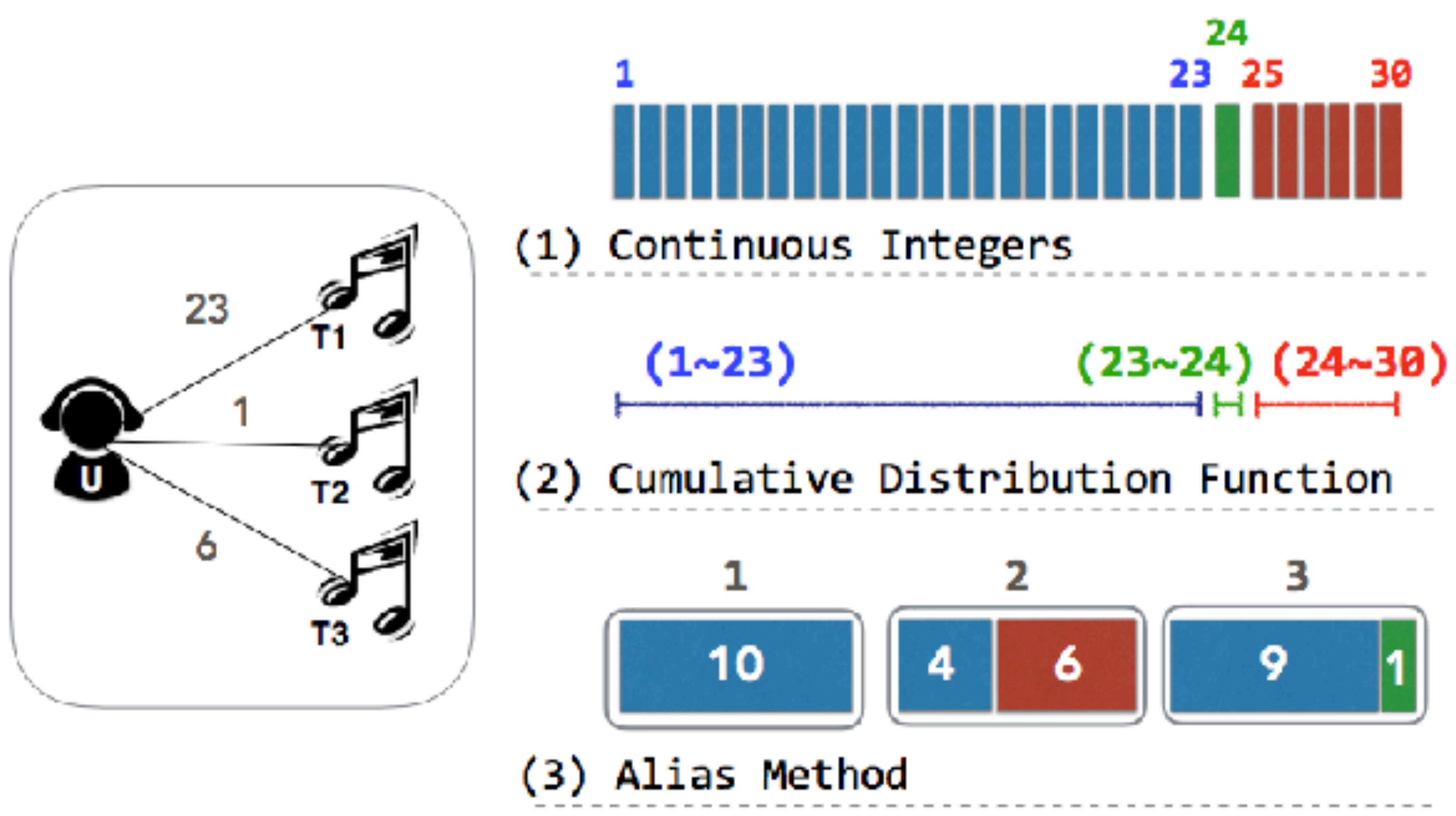
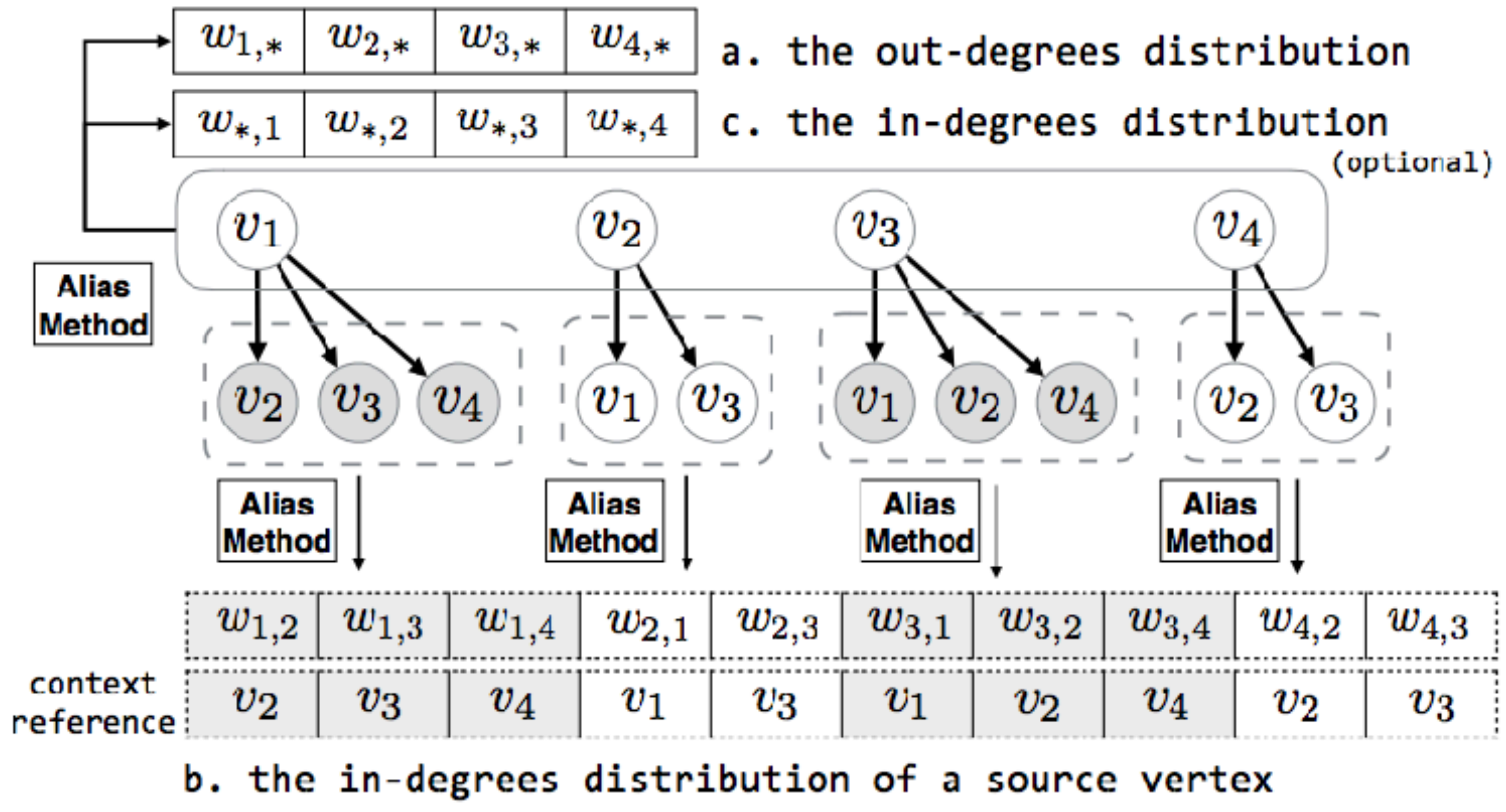


Figure 1: Three different approaches to sampling a vertex from a non-uniform distribution.



謝謝

Appendix

- LINE: https://arxiv.org/abs/1503.03578
- proNet-Core: https://github.com/cnclabs/proNet-core
- KL-Divergence: https://www.zhihu.com/question/41252833
- F1-Score: http://sofasofa.io/forum_main_post.php?postid=1001112
- awesome-network-embedding: https://github.com/chihming/awesome-network-embedding