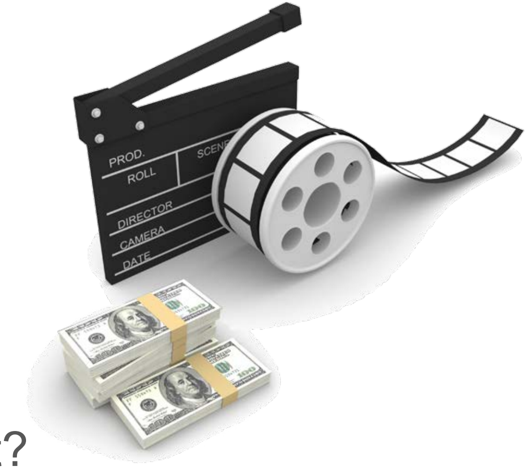
How to invest in movies

Deng Zhantao Huang Yu-Ting
Xia Shengzhao Zhang Yinan

Motivation

- Whether a movie will **succeed** decides the investment
- **Success = revenue - budget / budget**
- Successful factors:
  - Theme
  - Cast
  - Production company
- Successful factors of different genres would be **different**
- For a **specific genre**, **which factor** would influence it most?
- Hopefully, help investors to do a wise investment

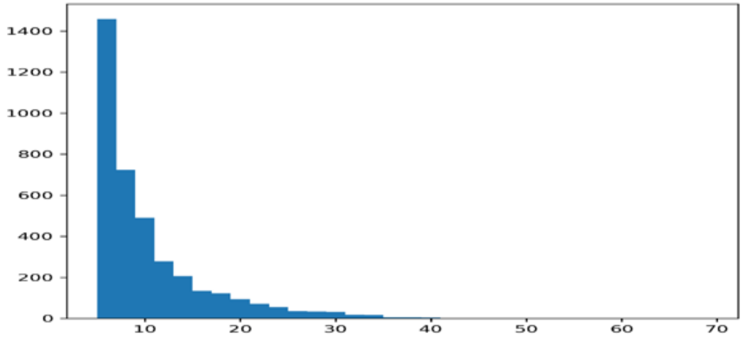


Data exploration

- **Kaggle** TMDB 5000 Movie Dataset
- Interested **features**:
  - Actors
  - Directors
  - Budget
  - Keywords
  - Production companies
- Let's do some data exploration:
  - Explore **actors feature** detailedly
  - Analyze and visualize via **networkx** package

Actors data exploration

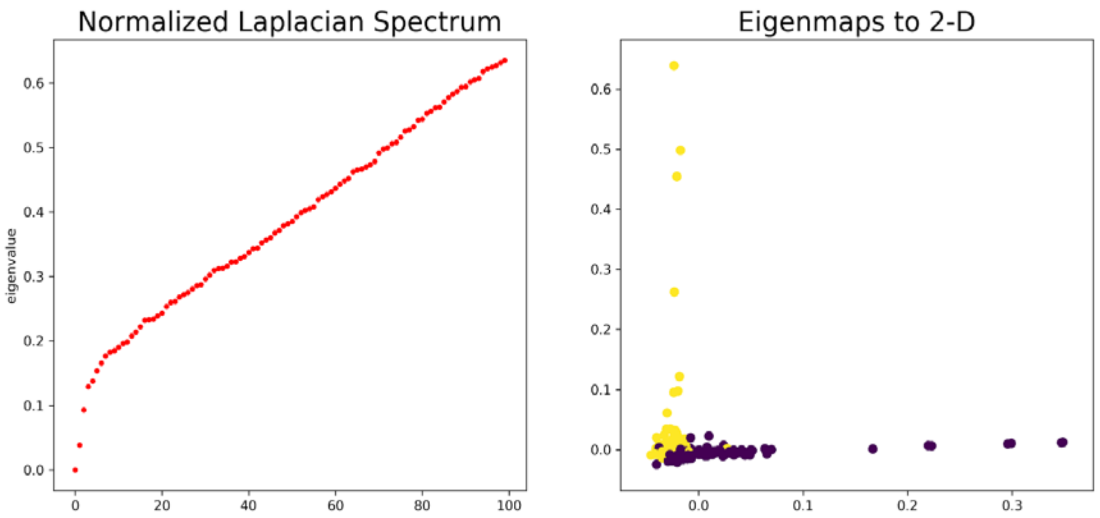
- **54201** distinct actors
- **Filter** actors appearing less than 5 movies, **3794** actors were left
- **Distribution**:



- Too **massive** for visualization and analysis
- **Subsump** 400 nodes **according to distribution**
- Preserve **generality** of data and easier to utilize

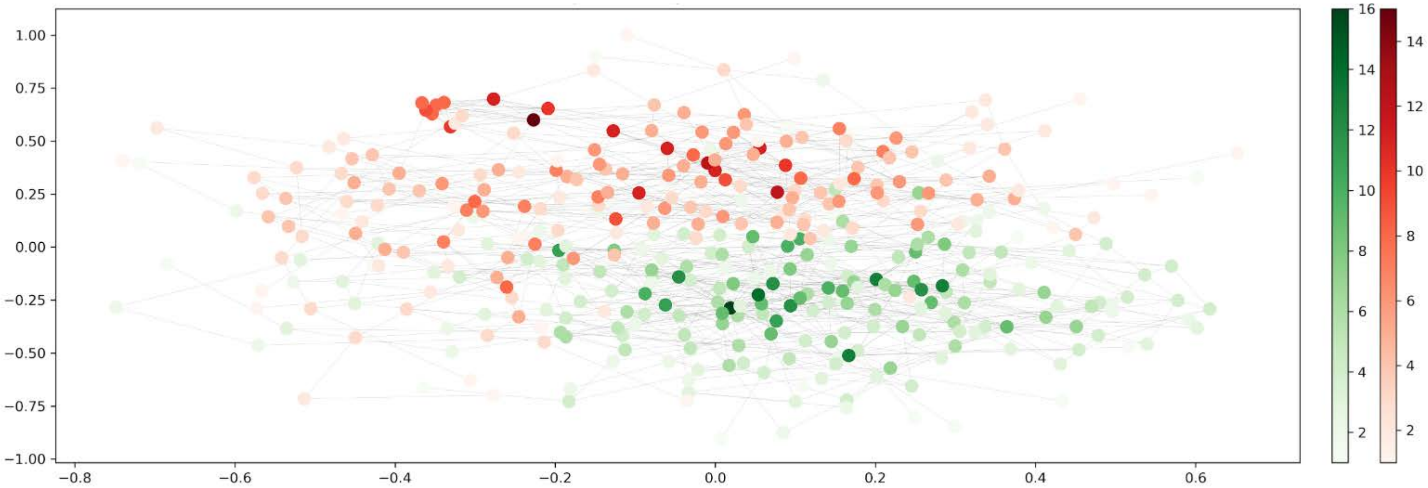
Clustering and actor social network

- Apply **spectral clustering** method



- **Largest gap** appeared after the second eigenvalue => **2 clusters**
- **Laplacian eigenmaps**: embed our graph in a 6-d Euclidean space
- **k-means**: a binary clustering
- Visualize the result on 2-d space

- Visualized the actor social network:



- 5 most **'sociable'** actors of each cluster:

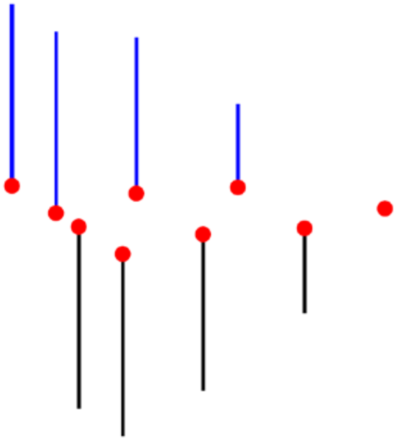
	Name	Degree
128	Julian Glover	15
111	Jason Ritter	12
44	Spencer Wilding	12
87	David Kelly	11
15	Elwin 'Chopper' David	11

	Name	Degree
108	Vera Farmiga	16
78	Kevin Corrigan	14
41	Adam LeFevre	13
144	Marcia Gay Harden	13
190	Carlos Alazraqui	13

Idea

- What we have:

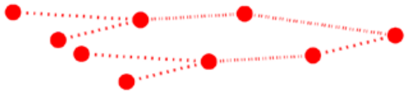
  Movie node, ROI signals



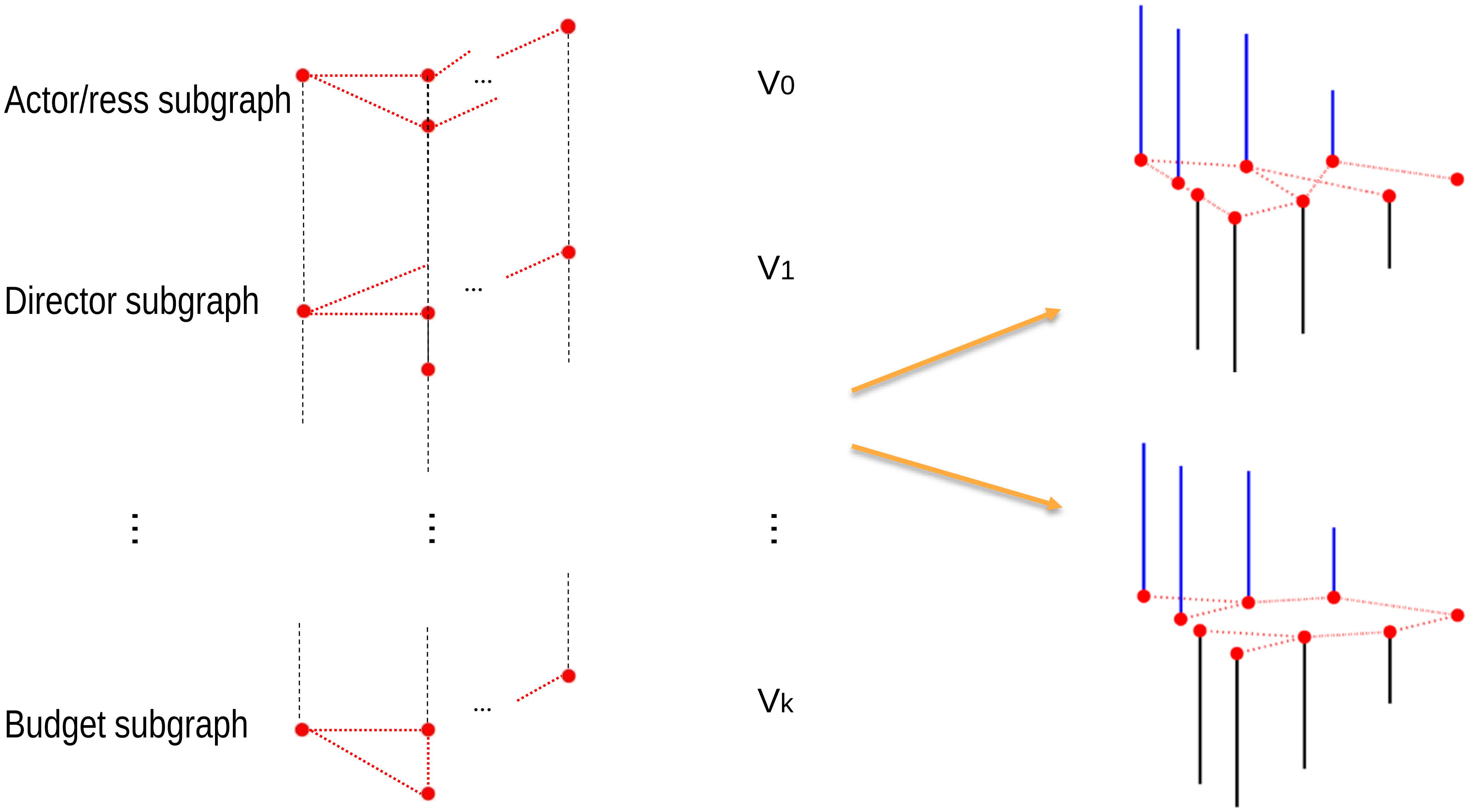
Features: budget, actor/ress, director, etc.

- What we want:

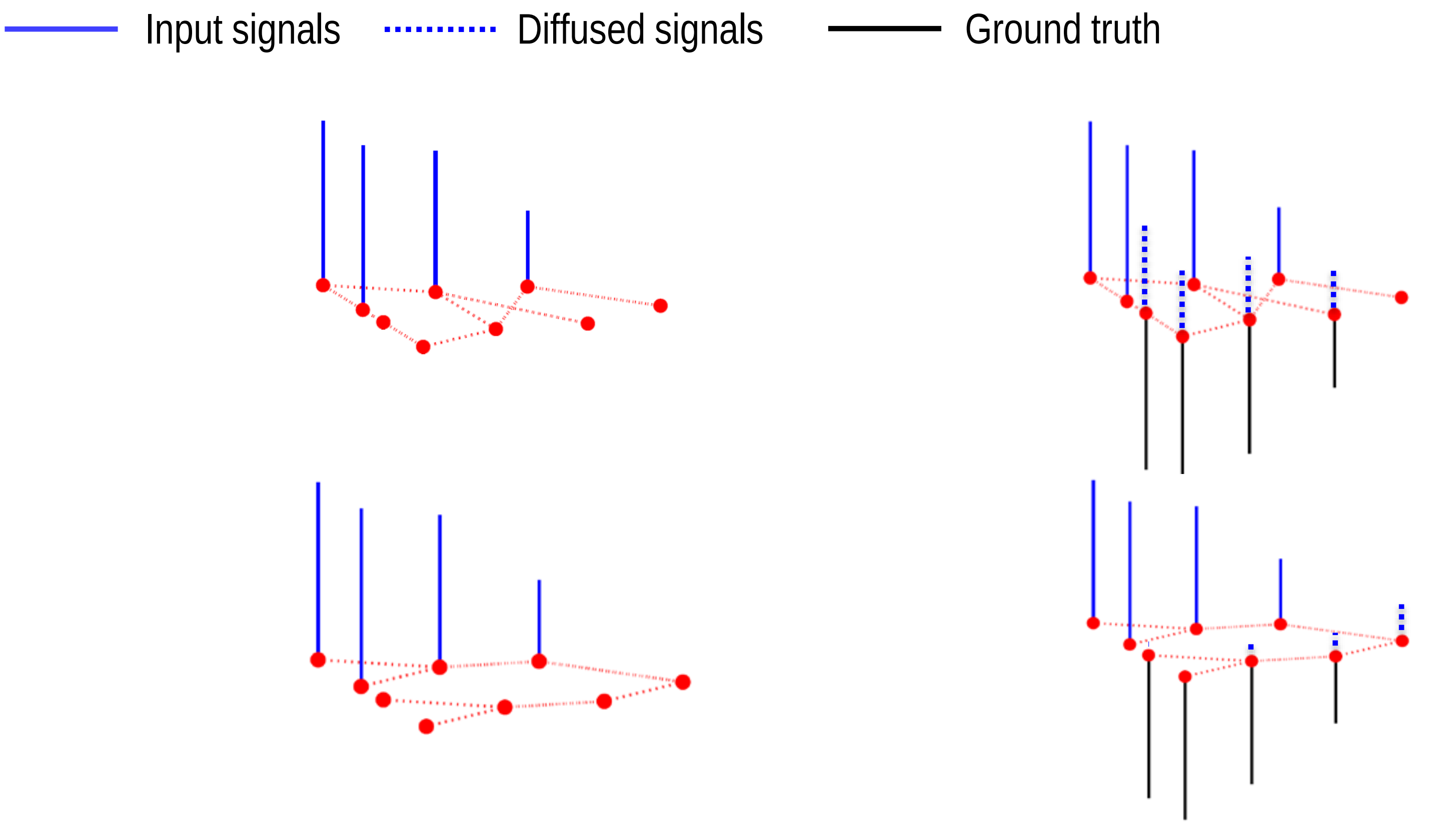
  A graph that can predict ROI



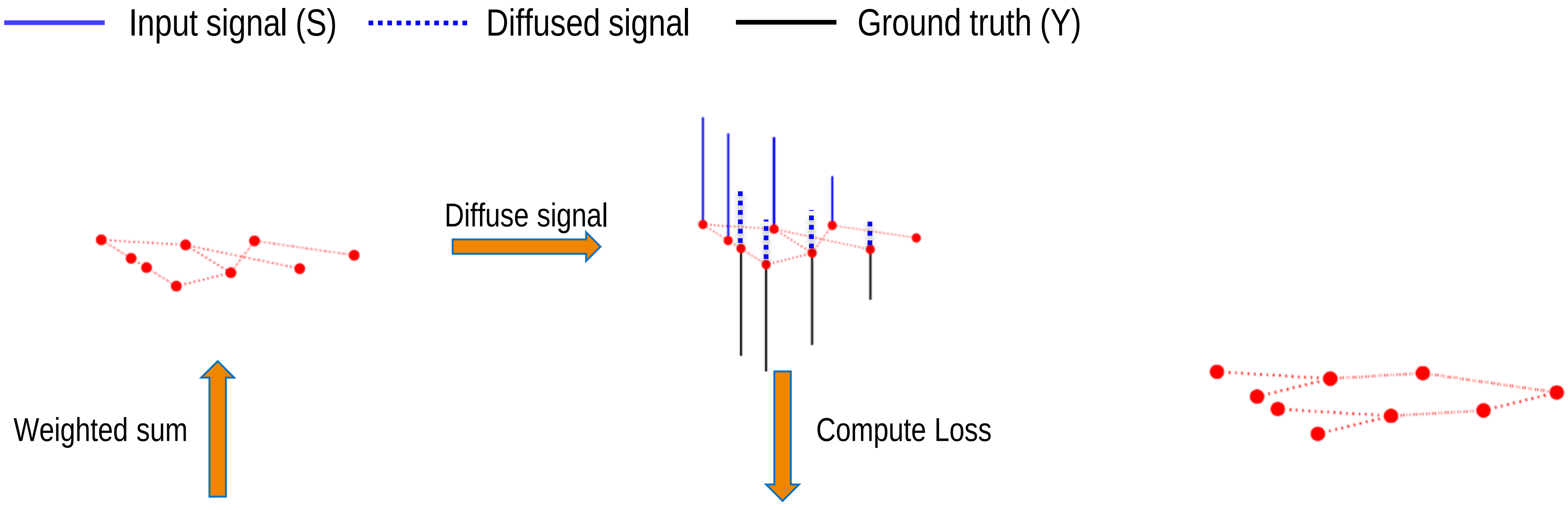
Idea


Actor/ress subgraph
Director subgraph
Budget subgraph
V0
V1
Vk

Heat kernel and Diffusion



Optimization



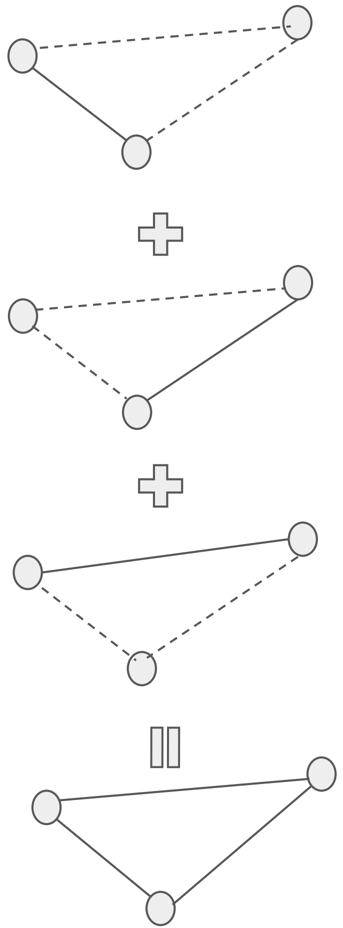
Gradient Decent

$V_{k,t}$

$$Loss = \frac{1}{2N}\|f \odot (H_t \cdot S - Y)\|_2^2 + \frac{\lambda}{2}\|v\|_2^2$$

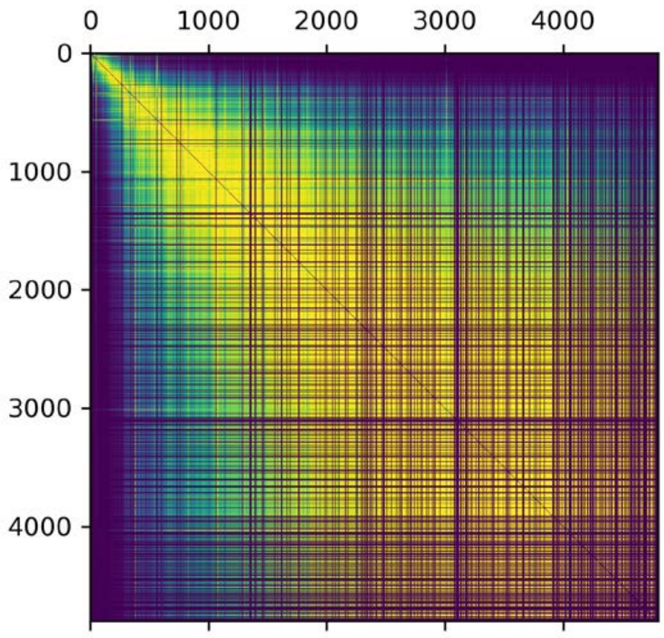
Building subgraphs

- Interpretring “success” as Return on Investment (ROI)
- A **linear weighted** sum of subgraphs
- Every node represents a movie
- Edge relates to actor / director/ keyword/ production company



Building subgraphs - Budget

- Range from 0 to 3E+8
- Treat nodes with missing and wrong values as **isolated** nodes
- Use the Gaussian function to turn Euclidean distance into edge weights



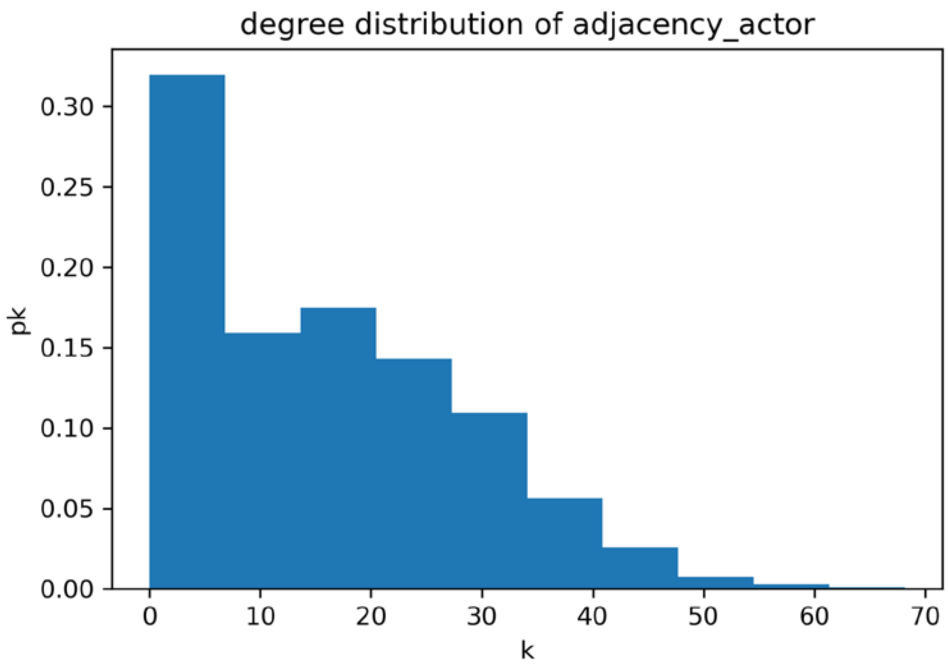
heat map of budget adjacency matrix

Building subgraphs - Actors, Directors

- Keep four actors or two directors at most
- Special characters replaced by '_'
- Use names to run the **bag-of-words** model
- **Double roles**

[0, 0, 0, 1, 0, 0, 0, 0, 1, 0, ⋯, 0, 0]

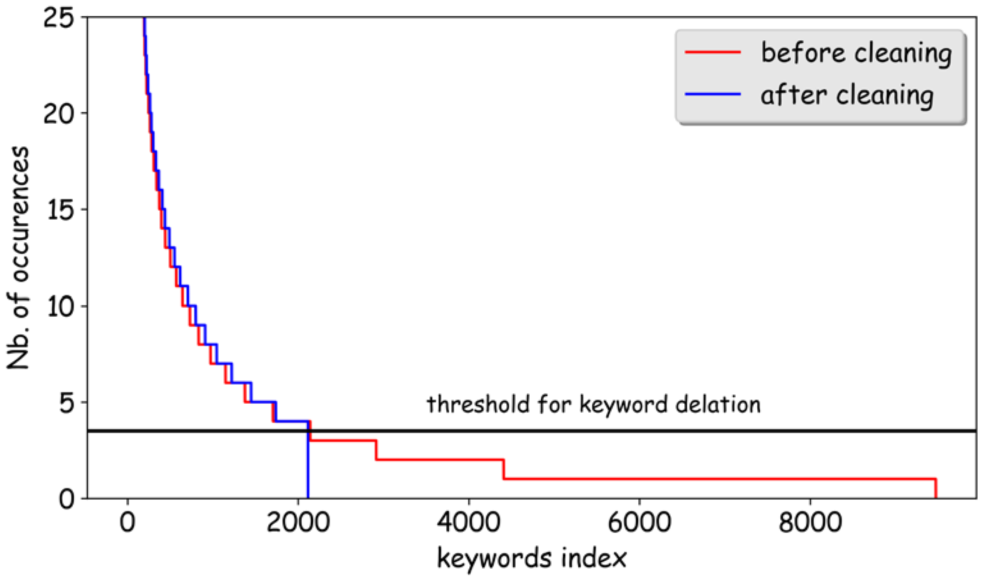
[0, 0, 0, 2, 0, 0, 0, 0, 1, 0, ⋯, 0, 0]



Degree distribution of actor adjacency matrix

Building subgraphs - Keywords

- More than 9000 keywords
- Group keywords that have the same **root**
- Replace keywords by **synonyms** of higher frequency
- Delete keywords that appear in fewer than four movies
- Bag-of-words model

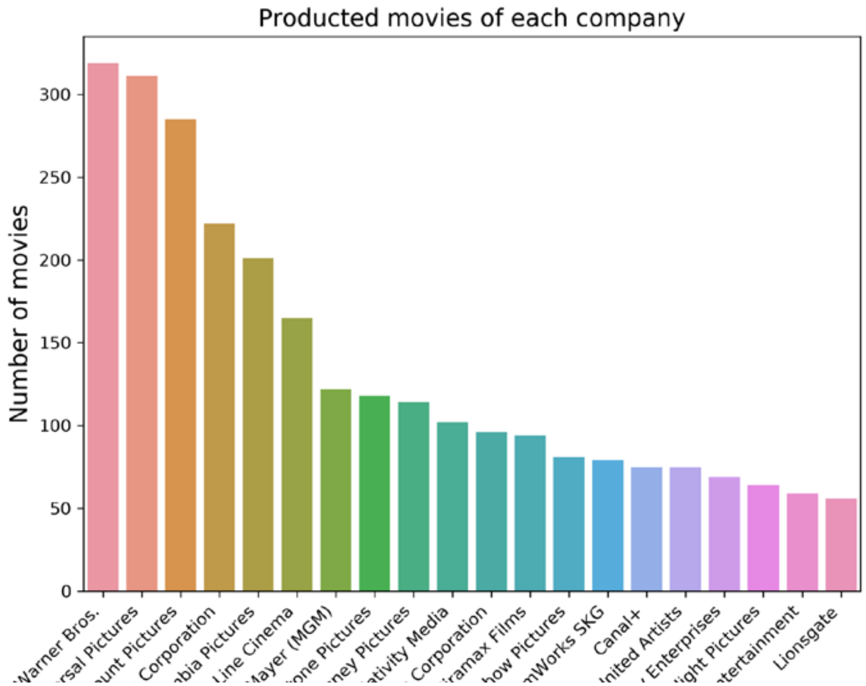


frequency of occurence of keywords before and after cleaning

[1] F. Daniel, “Film rrecommendation engine,” https://www.kaggle.com/fabiendaniel/film-recommendation-engine

Building subgraphs - Company

- Extracting production companies for each movie as subsets
- The **union** of all these subsets serve as the bag of companies
- Generatings feature vectors as **intersersections** which only contain 0 and 1



The frequency distribution map of movies producted by different companies

Loss function

$$Loss = \frac{1}{2N} \|f \odot (H_t \cdot S - Y)\|_2^2 + \frac{\lambda}{2} \|v\|_2^2$$
$$= \frac{1}{2N} \sum_i f_i \left[\sum_j H_{ij} S_j - Y_i \right]^2 + \frac{\lambda}{2} \sum_k v_k^2$$

$$H_t = e^{-tL} \approx I - tL + \frac{1}{2} t^2 L^2$$
$$L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$$

Ht: Heat kernel
S: Signal
Y: Groundtruth
f : Mask
N: number of valid indices

Gradient on t

$$\frac{\partial Loss}{\partial t} = \frac{1}{N}\sum_{i,j}\left[(H_t S - Y) \odot f \cdot S^T \odot (-L + tL^2)\right]_{ij}$$

Gradient on v

$$\frac{\partial Loss}{\partial v_k} = \sum_{i,j} \frac{\partial Loss}{\partial H_{ij}} \frac{\partial H_{ij}}{\partial v_k}$$
$$= \sum_{i,j} \frac{\partial Loss}{\partial H_{ij}} \sum_{k,l} \frac{\partial H_{ij}}{\partial L_{kl}} \frac{\partial L_{kl}}{\partial v_k}$$
$$= \sum_{i,j} \frac{\partial Loss}{\partial H_{ij}} \sum_{k,l} \frac{\partial H_{ij}}{\partial L_{kl}} \sum_{m,n} \frac{\partial L_{kl}}{\partial W_{mn}} \frac{\partial W_{mn}}{\partial v_k}$$

$$\frac{\partial W_{mn}}{\partial v_k} = W^k_{m,n}$$

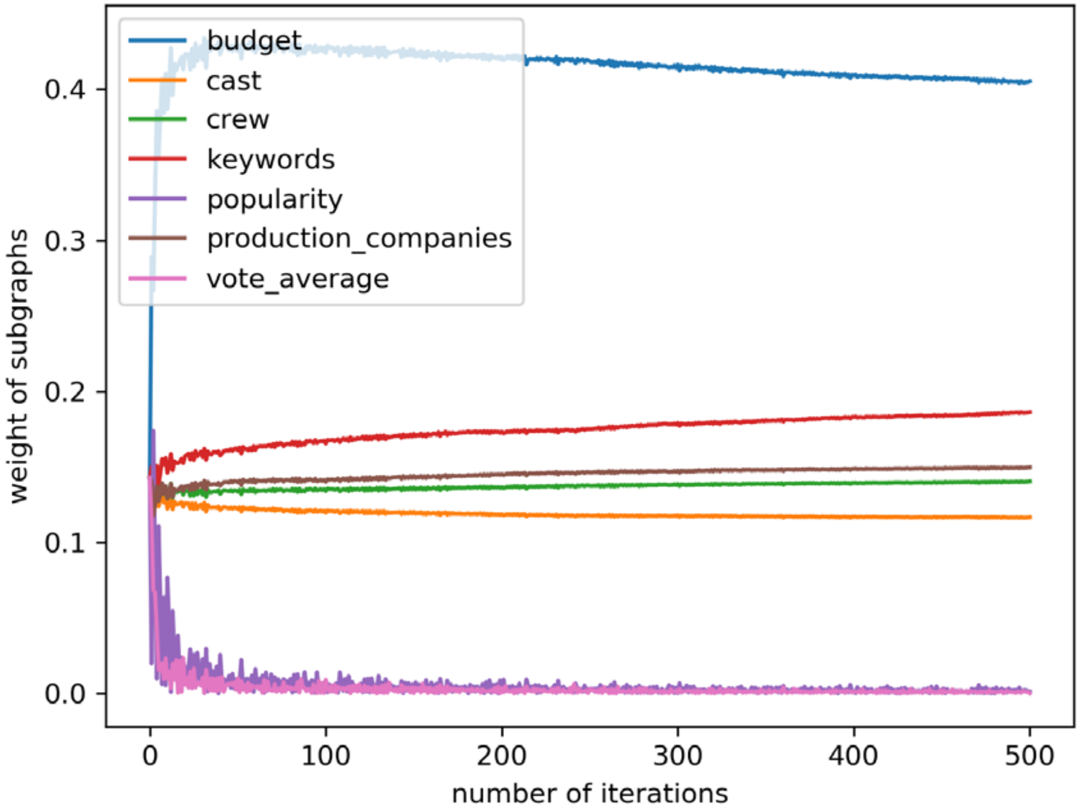
$$\frac{\partial L_{kl}}{\partial W_{mn}} = \left\{ \begin{array}{c} (m,n) = (k,l), \\ \frac{1}{2} D_{kk}^{-\frac{3}{2}} W_{kl} D_{ll}^{-\frac{1}{2}} - D_{kk}^{-\frac{1}{2}} D_{ll}^{-\frac{1}{2}} + \frac{1}{2} D_{kk}^{-\frac{1}{2}} W_{kl} D_{ll}^{-\frac{3}{2}} \\ m = k, n \neq l, \\ \frac{1}{2} D_{kk}^{-\frac{3}{2}} W_{kl} D_{ll}^{-\frac{1}{2}} \\ m \neq k, n = l, \\ \frac{1}{2} D_{kk}^{-\frac{1}{2}} W_{kl} D_{ll}^{-\frac{3}{2}} \end{array} \right\}$$

$$\frac{\partial H_{ij}}{\partial L_{kl}} = \left\{ \begin{array}{lc} (k,l) = (i,j), & -t + \frac{t^2}{2}(L_{ii} + L_{jj}) \\ k = i, l \neq j, & \frac{1}{2} t^2 L_{lj} \\ k \neq i, l = j, & \frac{1}{2} t^2 L_{ik} \end{array} \right\}$$

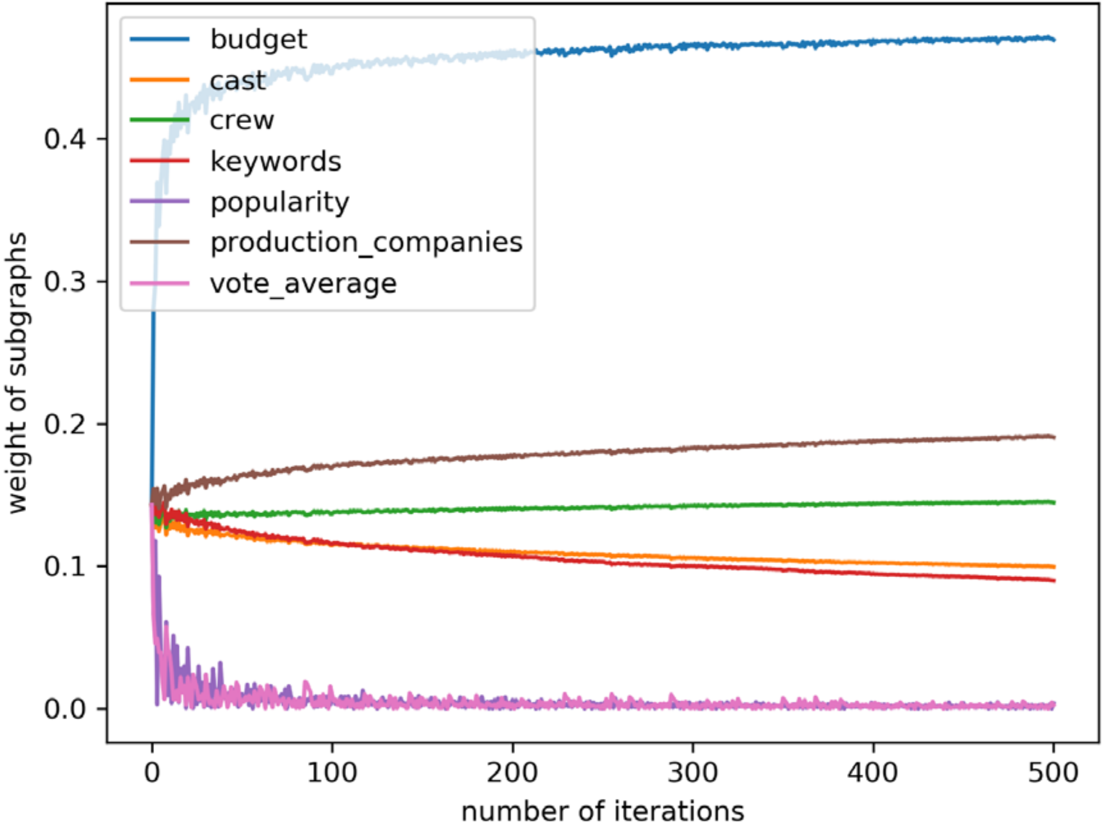
$$H_t = e^{-tL} \approx I - tL + \frac{1}{2} t^2 L^2$$
$$L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$$

$$\frac{\partial Loss}{\partial H_{ij}} = \frac{1}{N} [(H_i S - Y_i) f_i] S_j$$

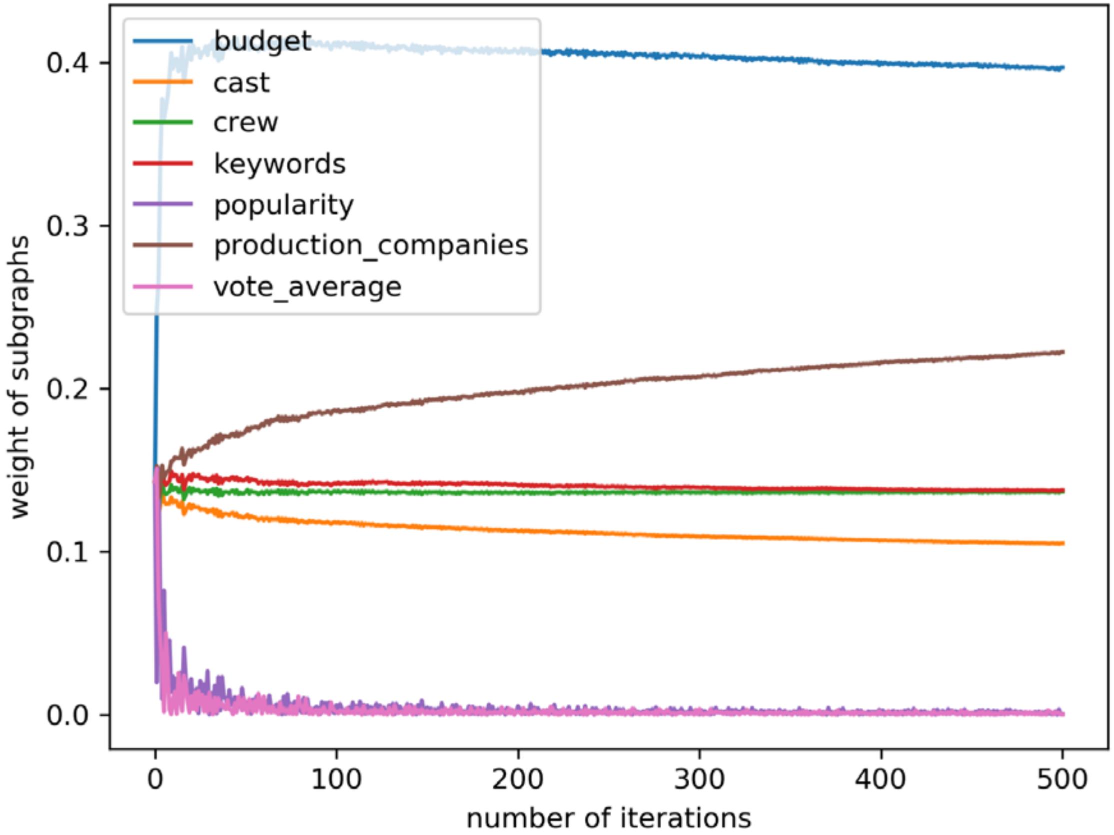
Result - Comedy



Result - Drama



Result - Romance



Prediction on ROI

Science fiction and drama

Interstella: (K = 3)

Predicted ROI: 3.10

Groundtruth : 3.09

Action

Pacific rim 2: (K = 5)

Predicted ROI: 0.87

Groundtruth : 0.93

Romance

La La Land: (K = 5)

Predicted ROI: 2.35

Groundtruth : 18.36

Thank you for your listening