

クロスエントロピーに基づく ネットワークデータの埋め込み

山田武士○、斉藤和巳、上田修功

NTT コミュニケーション科学基礎研究所
京都

研究の目的

- 複雑なリレーショナルデータの表現手段として、ネットワーク表現は有効

ネットワーク= N 個のノードとノード間のリンクからなる集合

- WEBページのハイパーリンクネットワーク
- 生物学の遺伝子制御ネットワーク
- 社会学のソーシャルネットワーク

- 低次元ユークリッド空間への効果的なネットワークの埋め込みが必要

- ネットワークを直接「ブラウジング」することによる新たな知見の獲得
 - ⇒ ネットワークに内在する知識の獲得と、科学的発見に結びつく構造の抽出の効率向上
 - ⇒ 発見科学、機械学習
- 次世代の3Dウェブブラウジングに向けたコア技術の提供
 - ⇒ Webサイエンス、Webテクノロジー
 - ⇒ Webに限らず各種情報の閲覧(写真、商品パンフレット、旅行案内、)



ブラウジングに最適な、ネットワークの低次元ユークリッド空間への埋め込みアルゴリズムを探求

埋め込みに関する二つの基本原則

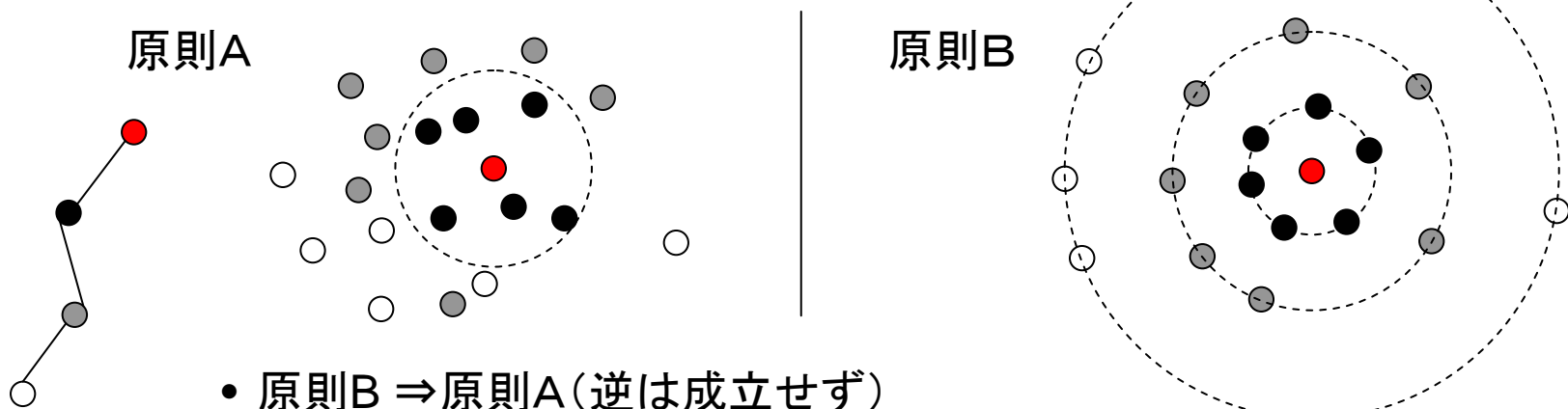
原則A (connectivity preserving principle)

各ノードは、それと隣接するノードを、隣接しないノードより相対的により近くに配置する。

原則B (distance preserving principle)

各ノードのペアにとって、相互のグラフ距離(最短経路)がユークリッド空間の距離としてもっとも忠実に復元されるように配置する。(主流)

⇒ 古典MDS(Torgerson)、ばねモデル(Kamada & Kawai)、スペクトラルクラスタリング(Ng)



- 原則B ⇒ 原則A (逆は成立せず)
- 原則Bを満たすのは困難 ⇒ 偏った配置になりがち
- グラフ距離がノード間の近接関係の忠実な反映とは限らない



できるだけ原則Aを満足する埋め込みを求めたい

クロスエントロピーに基づく埋め込み法

$\rho(d_{i,j}) \in [0,1]$, $\rho(0)=1$, $\rho(1)=0$, : (連続値) 類似度関数とし,
[原則A] の条件を類似度間のクロスエントロピーで近似に表現する

- N : ネットワークのノード数、 K : 埋め込み次元数
- $A = (a_{i,j})$, : 隣接行列, $a_{i,j} \in \{0,1\}$, $a_{i,j} = 1$, $a_{i,j} = a_{j,i}$ (無向を仮定)
- x_1, \dots, x_N : N 個のノードの座標 $d_{i,j} = \|x_i - x_j\|^2 = \sum_{k=1}^K (x_{i,k} - x_{j,k})^2$

- **[原則A]** 各ノードは, 隣接ノードを, 非隣接ノードより近くに配置

↓ 「近く」 = 「類似度 $\rho(d_{i,j})$ 大」

- **[類似度による言い換え]** 各ノードは隣接ノードとの類似度 $\rho(d_{i,j})$ が, 非隣接ノードとの類似度よりも大きくなるように配置

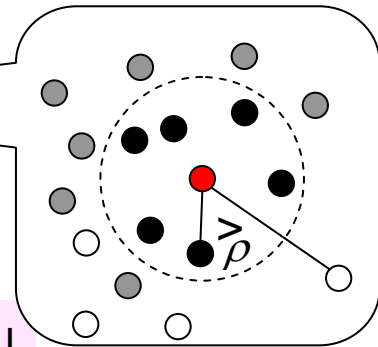
↓ 「類似度 $\rho(d_{i,j})$ 」 \doteq 「 $a_{i,j}$ (離散類似度とみなす)」

- **[類似度の近似]** $\rho(d_{i,j})$ を i,j の関数と見たとき、各ノードの連続類似度 $\rho(d_{i,j})$ が離散類似度 $a_{i,j}$ のもっとも良い近似となるように配置

↓ クロスエントロピー

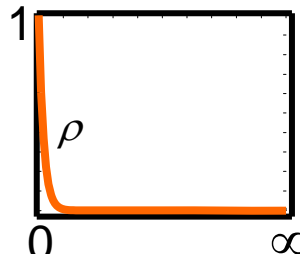
$\rho(d_{i,j})$ と $a_{i,j}$ との負のクロスエントロピーを最小化

$$E_{i,j} = -a_{i,j} \ln(\rho(d_{i,j})) - (1 - a_{i,j}) \ln(1 - \rho(d_{i,j}))$$



クロスエントロピーに基づく埋め込み法 (続き)

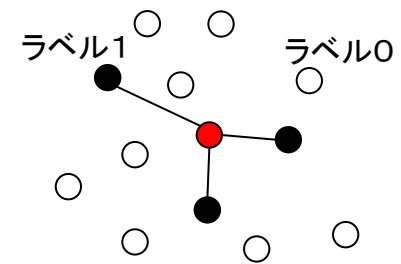
$$E_{i,j} = -a_{i,j} \ln(\rho(d_{i,j})) - (1 - a_{i,j}) \ln(1 - \rho(d_{i,j}))$$

$$\rho = \exp\left(-\frac{d_{i,j}}{2}\right) \quad \Downarrow$$


$$E_{i,j} = \frac{a_{i,j}}{2} \|x_i - x_j\|^2 - (1 - a_{i,j}) \ln\left(1 - \exp\left(\frac{1}{2} \|x_i - x_j\|^2\right)\right)$$

目的関数 $J = \sum_{i=1}^{N-1} \sum_{j=i+1}^N E_{i,j} + \frac{\mu}{2} \sum_{i=1}^n \|x_i\|^2$

正規化のためのWeight-Decay項



- i を固定したとき, $a_{i,j}$ はノード j の $\{0,1\}$ のクラスラベルを定義
- x_1, \dots, x_N をパラメータとする N 個の2値分類問題を同時に解くことと解釈可能

⇒ ニューラルネットワークのオンライン学習

学習アルゴリズム

各ノードにおいて勾配ベクトルがゼロとなるエネルギー極小配置を探索

- ニューラルネットワークのオンライン学習の手法を適用
(ばねモデルの学習法をほぼ踏襲)
 - 一度に1ノード移動し、その間他ノードは固定する逐次的アルゴリズム
 - 差分のみの更新で計算を効率化

入力: $N \times N$ 隣接行列 $A = (a_{i,j})$ 、埋め込み次元 K 、収束精度定数 ε

(1) $t = 1$ とし, x_1, \dots, x_N をランダムに初期化

(2) 勾配ベクトル $J_{x_1}^{(1)}, \dots, J_{x_N}^{(1)}$ を計算

(3) $i = \arg \max_j \{|J_{x_j}^{(t)}|^2\}$ となる x_i を選択

(4) $|J_{x_i}^{(t)}|^2 < \varepsilon$ ならば x_1, \dots, x_N を出力し終了

(5) 変分ベクトル Δx_i を計算
$$\Delta x_i = \begin{cases} -H^{-1} J_{x_i}^{(t)} & \text{if } J^{(t+1)} < J^{(t)} \\ \lambda J_{x_i}^{(t)} & \text{otherwise} \end{cases}$$

(\Rightarrow 収束を保証)

(6) $J_{x_1}^{(t)}, \dots, J_{x_N}^{(t)}$ を用いて $J_{x_1}^{(t+1)}, \dots, J_{x_N}^{(t+1)}$ を更新

(7) $x_i = x_i + \Delta x_i$ によって x_i を更新

(8) $t = t + 1$ とし, ステップ(3)へ

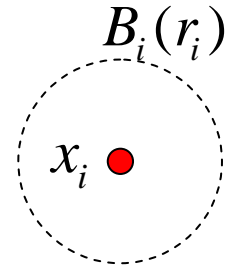
勾配ベクトルの変位を一次近似

$$J_{x_i}^{(t+1)} = 0 \approx J_{x_i}^{(t)} + \frac{\partial^2 J^{(t)}}{\partial x_i \partial x_i^t} \Delta x_i = 0$$

$$H = \frac{\partial^2 J^{(t)}}{\partial x_i \partial x_i^t} \quad (\text{ヘッシアン})$$

F -値に基づく評価

[原則A]を完全に実現する理想的な埋め込みにどれくらい近いかを、 F -値に基づいて評価する



$B_i(r_i)$: 各 x_i を中心とし、半径 r_i の K 次元球体

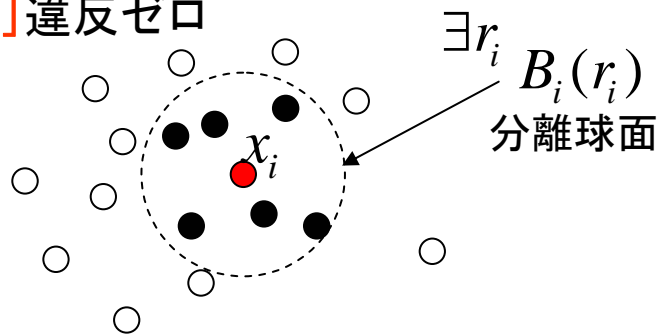
理想的な埋め込み \Rightarrow 各ノード i において完全分離球面が存在

- 各ノード i において適切な半径 r_i を選べば、 i に隣接ノードをすべて含み、非隣接ノードは全く含まない $B_i(r_i)$ を構成できる (違反ゼロ)

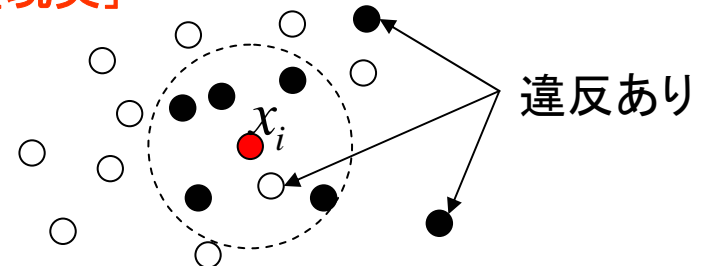
現実の埋め込み \Rightarrow 違反ゼロにはできない。

- 各ノード i において、 F -値の意味で違反度最小となる $B_i(r_i)$ を構成
- 全ノードの F -値の平均で埋め込みの良さを評価

[理想] 違反ゼロ



[現実]



F-値に基づく評価(続き)

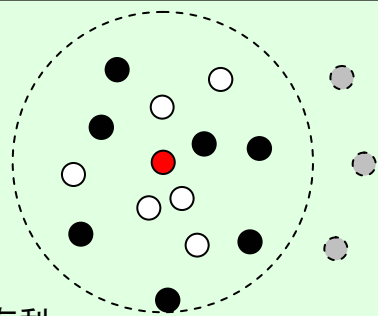
全ノードのF-値の平均で埋め込みの良さを評価

F-値: Precision と Recall の調和平均

$$F_i(r_i) = 1 / \left\{ \alpha \frac{1}{P_i(r_i)} + (1 - \alpha) \frac{1}{R_i(r_i)} \right\} \quad \longrightarrow \quad F = \sum_{i=1}^N \frac{F_i(\hat{r}_i)}{N} \quad \text{通常は } \alpha = 1/2$$

Precision

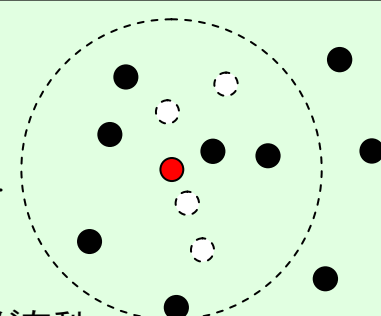
$$P_i(r_i) = \frac{7}{13}$$



半径が小さいほうが有利

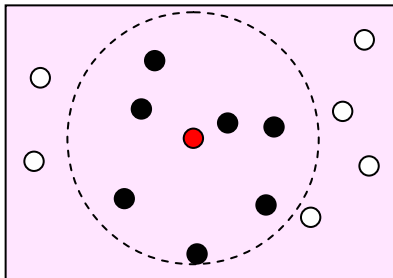
Recall

$$R_i(r_i) = \frac{7}{9}$$

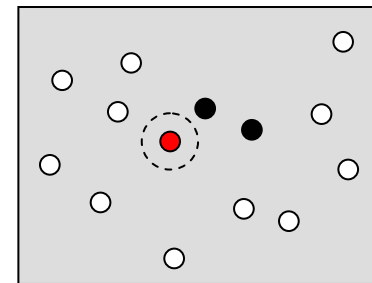


半径が大きいほうが有利

正解



Accuracy は高いが、
F-値は低い例 \Rightarrow
(スパースなNWでは非隣
接ノードがはるかに多い)

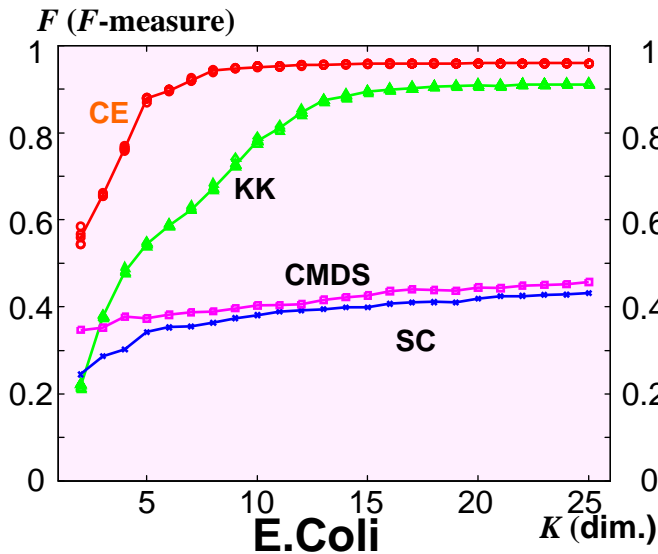


[原則B]を完全に実現する埋め込みでは $F = 1$ ($r_i = 1$ for all i)

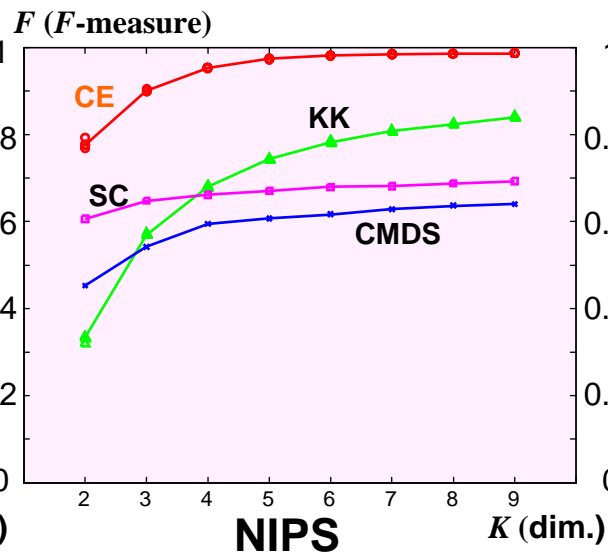
実データを用いた評価実験

E.Coli: 大腸菌 (*Escherichia coli*) の遺伝子制御ネットワーク
NIPS: NIPS掲載論文の共著者ネットワーク
NTT: NTTドメインに属するWWWページ

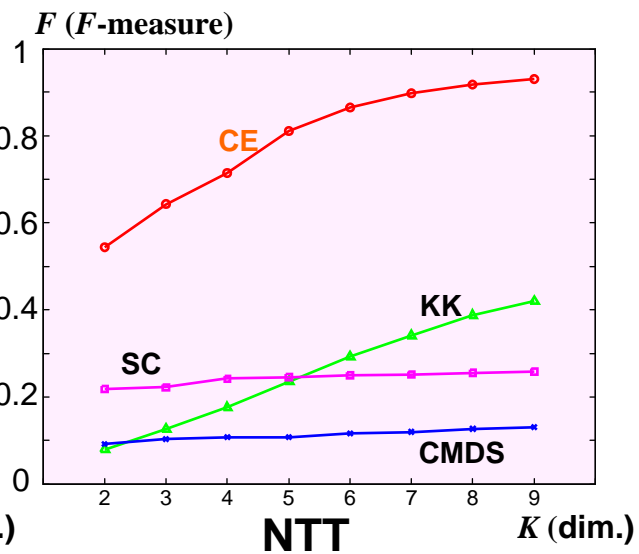
CE: 提案法、CMDS: Torgersonによる古典的MDS、KK: ばねモデル、SC: スペクトラルクラスタリング



ノード数 328, リンク数 456



ノード数1061, リンク数 2080



ノード数 2870, リンク数 9337

- 期待どおり, 提案法が他の手法に比べて, 特に配置が難しい低次元において, 優れている.
- 中でももっともサイズの大きいNTTを用いた実験において, 提案法と従来法の性能差が際立っている.

(参考) 比較に用いた従来法

CMDs: Torgersonによる古典的MDS

$$J_{\text{CMDs}} = \text{trace} \left\{ \left(-\frac{1}{2} \mathbf{Y} \mathbf{O} \mathbf{Y} - \mathbf{X} \mathbf{X}' \right)^2 \right\}$$

$\mathbf{G}=(g_{i,j})$: グラフ距離行列

$\mathbf{O}=(o_{i,j})$: $o_{i,j} = g_{i,j} \& g_{i,j}$

\mathbf{Y} : N -次元 Young-Householder 変換行列

SC: スペクトラルクラスタリング

$$J_{\text{SC}} = \text{trace} \left\{ \left(\mathbf{D}^{-2} \mathbf{B} \mathbf{D}^{-2} - \mathbf{X} \mathbf{X}' \right)^2 \right\}$$

$\mathbf{B}=(b_{i,j})$: $\exp(-g_{i,j}/2)$

$\mathbf{D}=(d_{i,i})$: 対角行列, $d_{i,i} = \mathbf{B}$ の第*i*行の全要素の和

$\hat{x}_{i,j} = \frac{x_{i,j}}{\sqrt{\sum_{j=1}^N x_{i,j}}}$ によって結果を球面上に表示

KK: Kamada & Kawaiによるばねモデル

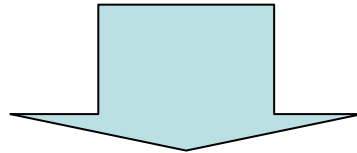
$$J_{\text{KK}} = \frac{1}{2} \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{(g_{i,j} - \|x_i - x_j\|)^2}{g_{i,j}^2}$$

2次元・3次元への埋め込み結果

配置例

各手法の特徴

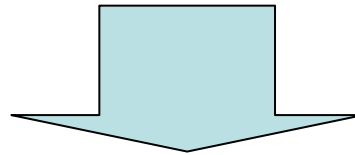
- **古典MDS**: 多くのノードが一点に縮退し、可視化、特にブラウジングの観点では問題あり⇒線型手法の限界
- **バネモデル**: 多くのノードが半円状にかたまって配置(「たんぽぽ効果」)
- **提案法**: より一様に放射状に広がり、空間を効率的に活用



- 埋め込み結果は F -値の性能差を良く反映している
- バネモデルが復元するグラフ距離が、離散値のため配置が偏在
- 従来法は特にノードが密集した領域において、ブラウジングが困難

結論

- クロスエントロピーに基づくネットワークの埋め込み法を提案
- F -値に基づく定量的評価によって、提案法は[原則A]をより忠実に反映していることを確認
- 視覚的にも従来法に比べ、より空間を効率的に活用した埋め込みであることを確認



今後

- 一般の(スパースな)重みつきネットワーク・類似度行列へ拡張
- 3Dウェブブラウザの配置エンジンとして組み込み
- 知識発見システムへ発展・組み込み
- もっと直接 F -値を最適化する手法を探求