

# ネットワークの可視化を簡単にするスプレッドシート用インタフェース

## Spreadsheet Interface to Simplify Network Visualization

小池 諭†  
Satoshi KOIKE †

三末 和男‡  
Kazuo MISUE ‡

田中 二郎‡  
Jiro TANAKA ‡

† 筑波大学 第三学群情報学類

‡ 筑波大学 システム情報工学研究科 コンピュータサイエンス専攻

† College of Information Sciences, ‡ Department of Computer Science, University of Tsukuba.

E-mail: † koike@iplab.cs.tsukuba.ac.jp, ‡ {misue, jiro}@cs.tsukuba.ac.jp

### Abstract

関係を表すデータが為すネットワーク構造を把握するには視覚的な表現が不可欠である。しかしながら、現在データベースとネットワーク可視化ツールは十分に連携しているわけではなく、関係データのネットワークを可視化するにはデータの抽出や変換などの煩雑な作業が必要である。筆者らはこの問題に対し、データベースにある関係データを閲覧・編集するためのツールとして用いられることの多いスプレッドシートとネットワーク可視化ツールを連携するというアプローチを試みた。本研究では、関係データが持つネットワーク型の知識の抽出・可視化を支援することを目的として、スプレッドシート上のデータからネットワークを対話的に抽出・可視化するウィザードを開発した。このウィザードを利用することでスプレッドシートから容易にネットワークを可視化できるようになり、ネットワークの可視化によるデータの分析・利用がより身近になると考えられる。

### 1. はじめに

知人関係は「人と人との関係」を表すデータであり、商品の購買関係は「購入者と商品との関係」や「商品とそれが売れる時間帯との関係」などを表すデータである。これら「関係データ」はビジネスにおける貴重な知識源である。そのため、これらのデータはデータベースで保存・管理されて様々なシーンで利用されている。例えば、コンビニエンスストアでは購買履歴から購入者の年代別や時間別の商品の売れ行きなどの購買傾向を読み取り、営業戦略に役立てている。また、インターネットのオンラインストアでは顧客の商品購買履歴から顧客の購入商品の共通点を調べて嗜好の類似度を読み取り、顧客に商品を紹介・推薦するサービスを提供している。

関係データはネットワーク構造を成すことが多く、その構造を把握することで有益な情報を発見しやすくなる。しかしながら、データベースに文字や数値の羅列で保存されている関係データからネットワーク構造を把握することは容易でなく、その構造を把握するには視覚的な表現が不可欠である。

ネットワークを視覚的に把握する手法として

ネットワーク図による可視化が挙げられ、多くの研究で様々な可視化手法が提案されている。現在多くのネットワーク可視化ツールが開発されており、データの種類や用途に応じて可視化手法を選び、ツールで可視化することができる。

データベースに保存された関係データのネットワークをツールで可視化するには、関係データからネットワークを構成する情報を抽出し、ツールに対応したデータ形式に変換する作業が必要になる。しかしながら、これらの作業はデータベースやツールで十分にサポートされているわけではなく、作業としては煩雑であり、また作業者の知識や技術に頼る部分が多い。そのため、ネットワーク図による可視化はネットワーク構造の把握に有効な手段であるにもかかわらず、広く普及しているとは言い難い。

筆者らはこの問題に対し、データベースに保存された関係データを表示・編集するためのツールとして用いられることの多いスプレッドシートとネットワーク可視化ツールを連携するというアプローチをとった。スプレッドシート上の関係データから簡単にネットワークの抽出・可視化を行えるようになれば、ネットワークの可視化によ

る関係データの分析・利用がより身近になると考えられる。

本研究は関係データが持つネットワーク型の知識の抽出・可視化を支援することを目的とし、スプレッドシート上で簡単な操作を行うだけでネットワークを可視化できるインタフェースを開発することを目指す。そのためにまず、関係データから抽出できるネットワークやその抽出法について考察し、抽出に必要な情報を整理した。その上で、それらの情報を対話的に獲得するインタフェースを設計、実装した。

## 2. データベースとネットワークの抽出法

データベースにおいて、同一のレコードに記録された要素の関係に基づくネットワークが構成できる。関係データから抽出できるネットワークはデータの種類によって様々であるが、本論文では、それらの中で「関係性に基づくネットワーク」と「類似度に基づくネットワーク」の2つに着目した。

### 2.1. データベース

本論文で扱うデータベースを以下のように表す。

$$\begin{aligned} \text{データベース } D &= [r_1, r_2, \dots, r_m] \\ \text{レコード } r_k &= [f_{k,1}, f_{k,2}, \dots, f_{k,n}] \end{aligned}$$

### 2.2. 関係性に基づくネットワークとその抽出法

異なる2つのフィールドの要素をノードとし、レコードごとの要素間の関係をエッジとすると、2つのフィールドの関係に基づくネットワークが構成できる。

データベース  $D$  の2つのフィールド  $f_p$  と  $f_q$  の関係に基づくネットワークは、重み付きグラフとして以下のように表せる。

$$\begin{aligned} \text{グラフ } G &= (V, E, w_r) \\ \text{ノード } V &= F_p \cup F_q \\ \text{エッジ } E &= \begin{cases} E \subseteq F_p \times F_q \\ E = \{e_1, e_2, \dots, e_m\} \\ e_k = (f_{k,p}, f_{k,q}) \end{cases} \end{aligned}$$

$$\text{エッジの重み関数 } w_r : E \rightarrow R$$

なお、 $F_p, F_q$  はそれぞれ  $p$  番目と  $q$  番目のフィールドに現れる要素全体の集合を表す。また、 $w_r$  はエッジの重みを与える関数である。

関係性に基づくネットワークを表すグラフのノード集合の各要素とそれに対するエッジ及び

エッジの重みは以下のように表せる。

$$F_p = \{a_{1,p}, a_{2,p}, \dots, a_{s,p}\}, (s \leq m)$$

$$F_q = \{b_{1,q}, b_{2,q}, \dots, b_{t,q}\}, (t \leq m)$$

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,t} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,t} \\ \vdots & \vdots & \ddots & \vdots \\ w_{s,1} & w_{s,2} & \cdots & w_{s,t} \end{bmatrix}$$

$$w_{i,j} = w_r(e_k)$$

関係性に基づくネットワークは、エッジに関係の強さを重みとして付加したグラフとして表せる。エッジの重みを与える関数  $w_r$  には以下のものが考えられる。

#### 1. 関係の頻度を重みとする場合

同一レコードに記録された回数は関係の強さを表すパラメータと言える。同一レコードに記録された回数をエッジの重みとする場合、 $w_r$  を以下のように定義できる。

$$\begin{aligned} w_r(e_k) &= |K_{i,j}| \\ \text{ただし, } K_{i,j} &= \{k \mid f_{k,p} = a_{i,p} \wedge f_{k,q} = b_{j,q}\} \end{aligned}$$

#### 2. 数値フィールドを重みとする場合

同一レコードに記録された数値フィールドの値を関係の強さとして扱いたい場合もある。エッジに数値フィールドの値を重みとして付加するときの計算方法には「合計」「平均」などが考えられる。エッジに数値フィールド  $f_r$  の値を重みとして付加する場合、 $w_r$  を以下のように定義できる。

$$\begin{aligned} \text{合計 } w_r(e_k) &= \sum_{k \in K_{i,j}} f_{k,r} \\ \text{平均 } w_r(e_k) &= \frac{\sum_{k \in K_{i,j}} f_{k,r}}{|K_{i,j}|} \end{aligned}$$

関係性に基づくネットワークを抽出するには、ノードとなるフィールド  $f_p, f_q$  と関係の強さとして扱うフィールド  $f_r$  を決定しなければならない。また、エッジの重みを与える関数  $w_r$  を指定する必要がある。

### 2.3. 類似度に基づくネットワークとその抽出法

データベースの1つのフィールドの要素をノードとし、要素間の類似度をエッジとすると、フィールドの要素間の類似度に基づくネットワークが構成できる。

フィールド  $f_p, f_q$  の関係性に基づくネットワークを考えたとき、 $a_{i,p} \in F_p$  の  $b_{1,q}, b_{2,q}, \dots, b_{t,q} \in F_q$  とのエッジの重み  $w_{i,1}, w_{i,2}, \dots, w_{i,t}$  は「 $a_{i,p}$  のフィールド  $f_p$  との関係の強さを表すベクトル要素」として表すことができる。

$$\begin{aligned} F_p &= \{a_{1,p}, a_{2,p}, \dots, a_{s,p}\}, (s \leq m) \\ F_q &= \{b_{1,q}, b_{2,q}, \dots, b_{t,q}\}, (t \leq m) \\ w_{i,j} &= w_r(e_k) \\ a_{1,p} &= [w_{1,1}, w_{1,2}, \dots, w_{1,t}] \\ a_{2,p} &= [w_{2,1}, w_{2,2}, \dots, w_{2,t}] \\ &\vdots \\ a_{s,p} &= [w_{s,1}, w_{s,2}, \dots, w_{s,t}] \end{aligned}$$

ベクトル  $a_{1,p}, \dots, a_{i,p}$  間の類似度を計算することで、「フィールド  $f_p$  のフィールド  $f_q$  についての類似度」がわかる。このときフィールド  $f_q$  は「類似度の計算対象のフィールド」と言える。

類似度の計算対象フィールドは複数選ぶこともできる。

$$\begin{aligned} F_v &= \{c_{1,v}, c_{2,v}, \dots, c_{u,v}\}, (u \leq m) \\ w_{i,j} &= w_r(e_k) \\ a_{1,p} &= [w_{1,1}, \dots, w_{1,t}, w_{1,(t+1)}, \dots, w_{1,(t+u)}] \\ a_{2,p} &= [w_{2,1}, \dots, w_{2,t}, w_{2,(t+1)}, \dots, w_{2,(t+u)}] \\ &\vdots \\ a_{s,p} &= [w_{s,1}, \dots, w_{s,t}, w_{s,(t+1)}, \dots, w_{s,(t+u)}] \end{aligned}$$

ベクトル  $a_{1,p}, \dots, a_{i,p}$  の間の類似度を計算することで、「フィールド  $f_p$  のフィールド  $f_q, f_v$  についての類似度」がわかる。

類似度はすべてのベクトル間で計算されるので、データベース  $D$  のフィールド  $f_p$  の要素間の類似度に基づくネットワークはすべてのノード間にエッジが存在する完全グラフとして以下のように表せる。

$$\begin{aligned} \text{グラフ} \quad G &= (V, E, w_s) \\ \text{ノード} \quad V &= F_p \\ \text{エッジ} \quad E &= F_p \times F_p \\ \text{エッジの重み関数} \quad w_s &: E \rightarrow R \end{aligned}$$

$w_s$  はエッジの重みを与える関数である。

類似度に基づくネットワークを表すグラフのノード集合の各要素とそれに対するエッジの重みは以下のように表せる。

$$F_p = \{a_{1,p}, a_{2,p}, \dots, a_{s,p}\}, (s \leq m)$$

$$W' = \begin{bmatrix} w'_{1,1} & w'_{1,2} & \dots & w'_{1,s} \\ 0 & w'_{2,2} & \dots & w'_{2,s} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w'_{s,s} \end{bmatrix}$$

$$w'_{i,j} = w_s(a_{i,p}, a_{j,p})$$

エッジの重みを求める関数  $w_s$  にはベクトル間の類似度計算法として「ベクトル間の距離の逆数」や「相関係数の絶対値」など様々なものが考えられる。

$$a_{i,p} = [w_{i,1}, \dots, w_{i,t}], a_{j,p} = [w_{j,1}, \dots, w_{j,t}]$$

### 1. Manhattan 距離の逆数

$$w_s(a_{i,p}, a_{j,p}) = \frac{1}{\sum_{k=1}^t |w_{i,k} - w_{j,k}|}$$

### 2. Euclid 距離の逆数

$$w_s(a_{i,p}, a_{j,p}) = \frac{1}{\sqrt{\sum_{k=1}^t (w_{i,k} - w_{j,k})^2}}$$

### 3. 相関係数の絶対値

$$w_s(a_{i,p}, a_{j,p}) = \left| \frac{\sum_{k=1}^t (w_{i,k} - \overline{a_{i,p}})(w_{j,k} - \overline{a_{j,p}})}{\sqrt{\sum_{k=1}^t (w_{i,k} - \overline{a_{i,p}})^2} \sqrt{\sum_{k=1}^t (w_{j,k} - \overline{a_{j,p}})^2}} \right|$$

$\overline{a_{i,p}}, \overline{a_{j,p}}$  はそれぞれのベクトル要素の平均値

類似度に基づくネットワークを抽出するには、ノードとなるフィールド  $f_p$ 、類似度の計算対象フィールド  $f_q, f_v, \dots$ 、関係の強さとして扱うフィールド  $f_r$  を決定しなければならない。また、ベクトル要素の計算方法  $w_r$  とエッジの重みの計算方法  $w_s$  を指定する必要がある。

### 3. ネットワークを可視化するための対話型インタフェース

#### 3.1. ネットワークの可視化における問題点

ネットワークの抽出・可視化のためには図1のような手順を踏む。

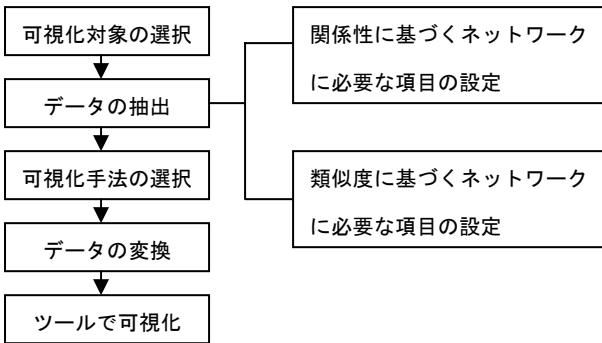


図 1: ネットワークの抽出・可視化の手順

前章で述べたように、データベースに保存された関係データからネットワークを抽出するには多くの項目を設定しなければならない。また、抽出したネットワークに対して効果的な可視化手法を選ぶ必要がある。さらに、抽出したネットワーク情報を可視化ツールに対応したデータに変換する必要があるが、このデータ変換は手作業や変換プログラムに通すなどして行われているのが現状である。これらの作業を利用者がすべて行うのは煩雑な作業である。また、これらの作業はデータ抽出に関する知識やデータ変換の技術が必要であり、これらの知識や技術のない者がこの作業を行うのは難しい。

#### 3.2. ウィザード

複雑な操作をサポートするインタフェースとして「ウィザード」が挙げられる。ウィザードは段階的な対話を通じてシステムの複雑な操作を行うインタフェースである。ウィザードでは、利用者はシステムからの質問に答える形で操作を行うため、操作のために何をしなければならないかを知らなくても質問に答えるだけで操作を行うことができる。またウィザードは段階的に質問をするので操作に必要な情報を絞り込むことができ、利用者は最低限の質問に答えるだけで操作を行える。ウィザードはアプリケーションのセットアップなど、複雑な操作を要求される場面で多く用いられている。

#### 3.3. ネットワーク可視化のためのウィザード

ウィザードはデータの抽出や可視化にも用いられている。スプレッドシートではデータから座

標系のグラフを描画するためのインタフェースとして「グラフウィザード」が実装されている。座標系グラフの描画には抽出フィールドやグラフの種類など多くの項目を設定する必要があるが、ウィザードを用いることで簡単にグラフを描画することを実現している。

関係データの閲覧や編集にはスプレッドシートがしばしば用いられている。また、関係データからネットワークを抽出・可視化する場合も、抽出フィールドや可視化手法の選択など座標系グラフの描画と類似した手順を踏む。このことから、ネットワークの可視化に関しても座標系グラフの描画と同様の手法が適用できると考えられる。

本研究では、データベースに内在するネットワークを抽出・可視化するためのインタフェースとしてウィザードを提案する。ウィザードにより、利用者はスプレッドシート上で起動したウィザードからの質問に答える形で情報を入力・選択することで、ネットワークを抽出して可視化ツールに対応したデータ形式に変換できるようになる。

### 4. ウィザードの設計と実装

#### 4.1. システム構成

図2は開発するウィザードのシステム構成図である。

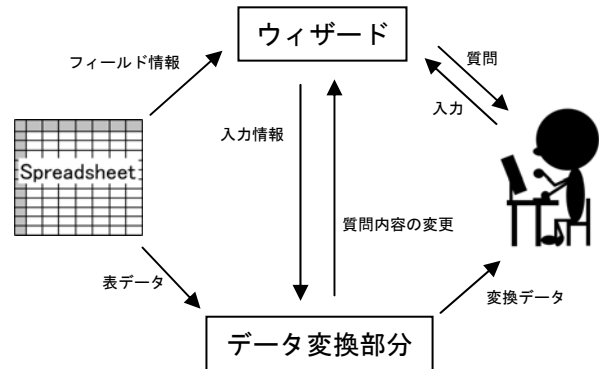


図 2: システム構成図

ウィザードは利用者に質問をすることでネットワークの抽出・可視化に必要な情報の入力を促す。また利用者に質問をする際、必要に応じてスプレッドシート上の関係データからフィールド情報などを取得する。

データ変換部分はウィザードで得た情報によってウィザードの質問内容を変更し、抽出・可視化に必要な情報を絞り込む。また、ウィザードで得た情報を基にスプレッドシート上の関係データからネットワーク情報を抽出し、データを変換して出力する。

## 4.2. ウィザードの機能

ネットワークの抽出・可視化に必要な情報は以下のような項目に分けられる。

1. 可視化対象の選択に関する項目
  - ・抽出するネットワークの選択
2. ネットワークの抽出に関する項目
  - 2-a. 関係性に基づくネットワーク
    - ・ノードとなるフィールド（2種類）
    - ・エッジの重みの対象フィールド
    - ・エッジの重みの計算方法
  - 2-b. 類似度に基づくネットワーク
    - ・ノードとなるフィールド
    - ・類似度の計算対象フィールド
    - ・関係性の対象フィールド
    - ・関係性の強さの計算方法
    - ・類似度の計算方法
3. 可視化手法の選択に関する項目
  - ・可視化手法の選択

ウィザードはこれらの情報を得るための質問を入力フォームで提示し、利用者に情報入力を促す。利用者はフォーム内のボックスやボタンに入力を行うことで抽出・可視化を行う。

また、それぞれの項目でフォームを分けて段階的に質問をすることで、抽出・可視化に必要な情報を絞り込み、最低限の情報入力でも抽出・可視化を行えるようにした。

## 4.3. データ変換

抽出したネットワークを可視化するには、可視化ツールに対応したデータに変換する必要がある。出力するデータの形式として、現在多くの可視化ツールの入力データ形式として使われている GraphML[6]と Web アプリケーションなどの入力データ形式として使われる CSV を採用した。

GraphMLによるグラフの記述例を図3に示す。node 要素はノードを定義する。要素の id 属性はノードの ID を記録し、後にエッジを定義する際に使用する。edge 要素はエッジを定義する。source 属性と target 属性にそれぞれのノードの ID を記録し、ノード間をつなぐ。ノード、エッジの子要素となる data key 要素を事前に定義しておき、子要素を追加することでノードやエッジに属性を付加する。図3の例では、ノードにラベルと色の属性、エッジに重みと色の属性を付加している。

```
<?xml version="1.0" encoding="UTF-8"?>
<graphml
xmlns="http://graphml.graphdrawing.org/xmlns">
<graph edgedefault="undirected">

  <key id="label" for="node"
attr.name="label" attr.type="string"/>
  <key id="n_color" for="node"
attr.name="n_color" attr.type="string"/>
  . . . . .
  <key id="weight" for="edge"
attr.name="weight" attr.type="double"/>
  <key id="e_color" for="edge"
attr.name="e_color" attr.type="string"/>
  . . . . .
  <node id="1">
    <data key="label">node A</data>
    <data key="n_color">red</data>
    . . . . .
  </node>
  <node id="2">
    <data key="label">node B</data>
    <data key="n_color">blue</data>
    . . . . .
  </node>
  . . . . .
  <edge id="1" source="1" target="2">
    <data key="weight">3</data>
    <data key="e_color">black</data>
    . . . . .
  </edge>
  <edge id="2" source="1" target="3">
    <data key="weight">2</data>
    <data key="e_color">black</data>
    . . . . .
  </edge>
  . . . . .
</graph>
```

図 3: GraphML によるグラフの記述例

CSV によるグラフの記述例を図4に示す。最も左の要素がノードやエッジであることを表す要素である。ノードである場合、ノードを表すキーワード「NODE」の後に、ID、ラベル、色などの属性をカンマで区切って記述する。エッジである場合、エッジを表すキーワード「EDGE」の後に、ID、ソースノードの ID、ターゲットノードの ID、重み、色、太さなどの属性を記述する。

```

//(Node),(Node ID),(Node Label),(Node Color),...
"NODE",1,NodeA,red,...
"NODE",2,NodeB,blue,...
"NODE",3,NodeC,green,...
.....
//(Edge),(Edge ID),(Source),(Target),
(Edge Weight),(Edge Color),(Edge Width),...
"EDGE",1,1,2,3,black,4,...
"EDGE",2,2,5,2,black,4,...
.....

```

図 4: CSV によるグラフの記述例

#### 4.4. 可視化手法

ウィザードで選択できる可視化手法として以下の3つを採用した。

1. アンカーマップレイアウト[3]  
2部グラフの片方のノードの位置に制約を課したレイアウト手法。一方のノードを円状に固定して配置（アンカーノード）し、もう一方のノードをばねモデルによって配置（フリーノード）することで、関係性の強いノード同士が近くに配置される。
2. アンカーマップの差分表現[4]  
2つの2部グラフの差分を算出し、1つのアンカーマップレイアウトの中でグラフ構造の変化を視覚的に表すレイアウト手法。フリーノードの移動をアイコンで表示する。
3. Animated Radial Layout[5]  
グラフの1つのノードを中心に配置し、中心ノードにリンクしたノードを円状に配するレイアウト手法。ノードをクリックすることで中心ノードを入れ替えることができる。

#### 4.5. ウィザードの実装

ウィザードの入力フォームは「可視化対象選択フォーム」「ネットワーク抽出フォーム」「可視化手法選択フォーム」の3つに分かれる。

可視化対象選択フォームのスクリーンショットを図5に示す。



図 5: 可視化対象選択フォーム

フォーム下には簡単な説明が表示される。フォーム上のオプションボタンで「関係性に基づくネッ

トワーク」と「類似度に基づくネットワーク」のいずれかを選択する。ここでの選択によって次に表示するネットワーク抽出フォームを変える。

関係性に基づくネットワークの抽出フォームのスクリーンショットを図6に示す。

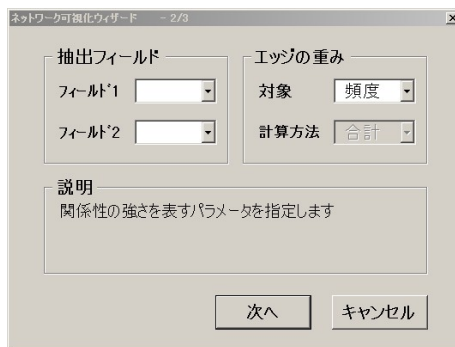


図 6: ネットワーク抽出フォーム(関係性)

フォーム左のコンボボックスの要素はデータのフィールドと対応しており、この部分でノードにするフィールドを選択する。またフォーム右の「対象」コンボボックスは重みとして付加できる数値フィールドと対応しており、この部分でエッジの重みの対象フィールドと重みの計算方法を選択する。

類似度に基づくネットワークの抽出フォームのスクリーンショットを図7に示す。



図 7: ネットワーク抽出フォーム(類似度)

フォーム左のコンボボックス及びリストボックスはデータのフィールドと対応しており、この部分でノードにするフィールドと類似度の計算対象フィールドを選択する。フォーム右上のコンボボックスは重みとして付加できる数値フィールドと対応しており、この部分で関係の強さの対象フィールド及びその計算方法を選択する。フォーム右下のコンボボックスで類似度の計算方法を選択する。

図のどちらのフォームでも、フォームを進める

と可視化手法を選択するフォームが表示される。  
可視化手法選択フォームのスクリーンショットを図8に示す。



図 8: 可視化手法選択フォーム

画面左のオプションボタンで出力するファイルの形式を選択する。画面右のコンボボックスもしくはイメージ図で可視化手法を選択する。このフォームで可視化手法を選んでフォームを進めると、ファイルを出力した旨を伝えるダイアログボックスが表示され、データが出力される。

#### 4.6. 利用例

商品の購買履歴のデータ(図9)からネットワークを可視化した例を示す。ここでは、「購入者と商品カテゴリとの関係」に基づくネットワークを抽出し、RadialLayout手法で可視化した。

	A	B	C	D	E
1	name	date	time	category	price
2	Y.S	2006/6/2	22:32:24	Instant	100
3	SH.U.S	2006/6/5	8:21:30	Juice	100
4	H.M	2006/6/5	11:21:57	Instant	100
5	SH.L.S	2006/6/5	12:36:17	Instant	100
6	N.N	2006/6/5	12:52:02	Instant	100
7	K.M	2006/6/5	13:39:50	Instant	100
8	SH.L.S	2006/6/5	14:35:13	Tea	100
9	J.Y	2006/6/5	14:42:52	Snack	100
10	SH.U.S	2006/6/5	15:09:38	Coffee	100
11	H.M	2006/6/5	16:49:36	Tea	100
12	H.M	2006/6/5	16:50:14	Juice	100
13	SH.U.S	2006/6/5	17:59:05	Juice	100
14	Y.T	2006/6/5	19:39:50	Tea	100
15	S.K	2006/6/5	20:31:20	Tea	100
16	N.N	2006/6/5	20:37:35	Juice	100
17	M.I	2006/6/5	20:42:47	Juice	100

図 9: 購買履歴のデータ

ウィザードの各フォームでの選択は以下の通り。

1. 可視化対象選択フォーム  
可視化対象：関係性に基づくネットワーク
2. ネットワーク抽出フォーム  
フィールド1：購入者  
フィールド2：カテゴリ  
エッジの重みの対象：(購入) 頻度

#### 3. 可視化手法選択フォーム

出力ファイル：GraphML

レイアウト手法：RadialLayout

出力されたデータをツールで可視化した(図10)。

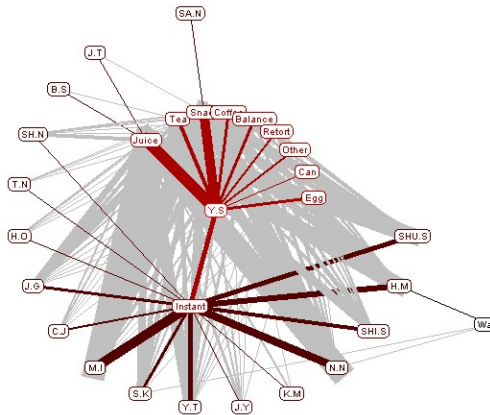


図 10: ツール(prefuse[5])による可視化結果

### 5. 評価と考察

#### 5.1. 作業ステップの比較

購買履歴のデータを使って、関係性に基づくネットワークと類似度に基づくネットワークについて、従来行われていた方法とウィザードを使った方法の2つの方法でネットワークを抽出、ツールで可視化し、作業ステップの比較を行った(表1)。

表 1: 作業ステップの比較

従来法	ウィザード
0. 中間ファイルをGraphMLファイルに変換するプログラムを事前に準備	
1. 手作業でネットワークを抽出	1. ウィザードから提示される項目を設定してネットワークを抽出し、GraphMLファイルを出力
2. 抽出したネットワーク情報を中間ファイルとして出力	
3. 中間ファイルを変換プログラムを通してGraphMLファイルに変換して出力	
4. GraphMLファイルをツールで可視化	2. GraphMLファイルをツールで可視化

#### 5.2. 考察

従来法では、可視化ツールに対応したGraphMLファイルを出力するために中間ファイルを出力しなければならない。また、中間ファイ

ルを GraphML ファイルに変換するプログラムを用意する必要があり、それがなければ手作業でデータ変換を行わなければならない。その一方で、ウィザードを使う方法ではそうした煩雑な作業を一切行うことなくツールに対応した GraphML ファイルを出力できる。作業ステップの減少と操作の簡略化という点で、このウィザードはネットワークの抽出・可視化を支援するという目的を果たすものであると言えるだろう。

## 6. 関連研究

ネットワークの可視化を支援するという目的に対して表データとネットワーク図を連携するというアプローチをとった研究は以前からいくつか行われてきた。

N.Henry らは、表形式とネットワーク図を同時に表示することでネットワーク構造の把握を支援するシステム MatrixExplorer[1]を開発した。このシステムは、表形式で表されたソーシャルネットワークにおける人物間のリンク関係をネットワーク図でも同時に表し、表と図の対応などを視覚的に表しながらネットワーク構造の把握を支援するというものである。

J.Mackinlay らは、関係データに対して有効な図表現をデザインするツール APT(A Presentation Tool)[2]を開発した。このツールは、関係情報の特徴に合わせて棒グラフや散布図、ネットワーク図と言った図表現を自動的にデザインするものである。

これらの先行研究に対して本研究では、表データとネットワーク図の連携にウィザードを用いるという新しいアプローチを試みた。また、表データを扱うツールとして広く普及しているスプレッドシートを用いることでネットワーク図による可視化を身近なものにした。

## 7. まとめと今後の課題

本研究では、ネットワークの抽出・可視化に必要な情報を対話的に獲得するウィザードを開発した。このウィザードを使うことでスプレッドシート上の関係データから簡単にネットワークを可視化できるようになった。

今後は、本論文で着目した「関係性に基づくネットワーク」「類似度に基づくネットワーク」以外のネットワークについても調査し、ウィザードの対象範囲を広げたいと考えている。

## 参考文献

- [1] N.Henry and J.D.Felete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE Transactions on Visualization and Computer Graphics, Vol.12, No.5, September/October 2006. pp. 677-684
- [2] J.Mackinlay. Automating the Design of Graphical Presentations of Relational Information. ACM Transactions on Graphics, Vol.5, No.2, April 1986, pp. 110-141
- [3] Kazuo Misue. Drawing Bipartite Graphs as Anchored Maps. Proceedings of Asia-Pacific Symposium on Information Visualization (APVIS2006), pp. 169-177 (February 1-3, 2006)
- [4] 石原正樹. 動的ネットワークの成長過程と差分の可視化手法. 筑波大学大学院博士課程 2006年度システム情報工学研究科修士論文
- [5] J.Heer, S.K.Card and J.A.Landay. Prefuse: a toolkit for Interactive Information Visualization. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 2005, pp. 421-430, 2005
- [6] U.Brandes and Pich. GraphML Transformation. Proceedings of Symposium on Graph Drawing 2004. pp. 89-99