

統計的テキスト解析 (17)

～ テキストにおけるアソシエーション分析と補遺 ～

同志社大学文化情報学部教授

金 明哲 (Jin Mingzhe)

■中国生まれ。総合研究大学院大学数物研究科統計科学専攻博士後期課程修了。博士(学術)。1995年札幌学院大学社会情報学部、助教授、教授を経て、2005年4月より現職。E-mail: mjin@mail.doshisha.ac.jp



テキスト解析およびテキストマイニングは、統計学、データマイニング、人工知能、自然言語処理などの学際領域である。多岐にわたる学際領域の話題を限られた誌面で漏れなく紹介することは困難である。本連載では、最も基本と思われ、かつ、今後のテキストマイニングや計量言語学などの研究と応用の基礎となる統計方法を中心として実例を用いながら説明してきた。説明した方法は、現時点のテキストマイニングツールに用いられている機能に関連する統計知識のほとんどをカバーしている。しかし、個別のツールに用いられてはいるが、触れていないものもある。例えば、データマイニングの中のアソシエーション分析 (Associations Analysis) の手法をテキスト分析に適用する方法や評判分析などがある。本稿では、これらについて説明する。

1. アソシエーション分析

アソシエーション分析は、百貨店や店舗の購買データ (バスケットの中の商品間の関係)

について分析を行う方法であり、相関ルール分析、連関ルール分析、関連ルール分析などとも呼ばれる。この分析の主な目的は、バスケットのデータから頻出するアイテムの組み合わせの規則を漏れなく抽出し、その中から興味深い結果を探し出すことである。

アソシエーション分析では、「商品Aを買うと商品Bも買う」のようなルールを簡潔に「 $\{A\} \Rightarrow \{B\}$ 」、あるいは $X \Rightarrow Y$ の形式で表す。ルールの「 \Rightarrow 」の左辺を条件部 (Antecedent: Left-hand-side or LHS)、右辺を結論部 (Consequent: Right-hand-side or RHS) と呼ぶ。最も広く知られているルールを検出するアルゴリズムはAprioriである。

データベースの中からアソシエーションルールを検出する際、何らかの評価指標が必要である。多く用いられている指標は、支持度 (support)、確信度 (confidence)、リフト (lift) である。アイテム集合 X を含むケース (データベースの分野では、トランザクションと呼ぶ) の数を X の支持度数と呼び、 $\sigma(X)$ を用

いて表すことにする。ルール $X \Rightarrow Y$ の支持度は、アイテム集合 X と Y を含むケースがその総数 M に占める比率であると定義されている。

$$\text{supp}(X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{M}$$

確信度とは、アイテム集合 X と Y を含むケースの数 $\sigma(X \cup Y)$ を、条件 X を含むケースの数 $\sigma(X)$ で割った値である。

$$\text{conf}(X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} = \frac{\text{supp}(X \Rightarrow Y)}{\text{supp}(X)}$$

リフトは、確信度を結論部のケースの数 $\sigma(Y)$ が全体の中に占める比率 $\text{supp}(Y)$ で割った値で定義する。

$$\text{lift}(X \Rightarrow Y) = \frac{\text{conf}(X \Rightarrow Y)}{\text{supp}(Y)}$$

支持度が低いほど、そのルールが現れる比率が低い。しかし、アイテム数が多い場合は、個別のアイテムの支持度が非常に大きい値になることは、まれである。ルールの評価に当たっては、支持度、確信度、リフトを総合的に考慮する必要がある。

短い文を語や文節に分割し、その語や文節をバスケットの中のアイテムと見なすことで、語や文節の共起関係のルールを抽出することができる。

例を用いて説明するため、ある大学で定期的に行う学生実態調査のアンケート調査の自由回答文10例を次に示す。

***** 自由回答文の例 *****

- ・ 学費を下げ、講義の充実をはかって欲しい。適当に授業をしていると思われる先生が、かなり居る。
- ・ 学費をもう少し安くして欲しい。
- ・ 休み期間が多い割に学費が高い。何に使われているかははっきりして欲しい。

- ・ 授業担当の教員を、生徒に選ばせて欲しい。
- ・ 学費の削減を！あとロッカーを！
- ・ 個人ロッカーを作ってください。自動車通学認めて下さい。
- ・ 学費軽減。
- ・ 学費をもっと安くして欲しい。
- ・ クーラーをつけて欲しい。
- ・ 学費安くして下さい。

Rの中のパッケージarulesを用いて説明する。上記の文について形態素解析を行い、若干整理した自立語を回答者ごとにRに読み込む。ここでは、キーボードで入力することにする。

```
data1<-list(
c("学費","下げ","講義","充実","はかっ","欲しい",
"適当","授業","いる","思わ","れる","先生","かなり",
"居る"),
c("学費","もう少し","安く","欲しい"),
c("休み","期間","多い","割","学費","高い","何",
"使わ","いる","はっきり","欲しい"),
c("授業","担当","教員","生徒","選ば","欲しい"),
c("学費","削減","ロッカー"),
c("個人","ロッカー","作っ","下さい","自動車",
"通学","認め"),
c("学費","軽減"),
c("学費","もっと","安く","欲しい"),
c("クーラー","つけ","欲しい"),
c("学費","安く","下さい")
)
```

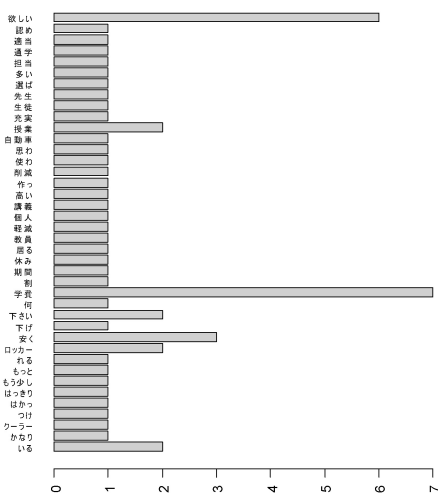
パッケージarulesで主に用いているデータ形式は、トランザクション形式である。パッケージarulesをインストールし、次のように関数asと引数transactionsを用いると、リスト形式をトランザクション形式に変換することができる。

```
>install.packages("arules")
>library(arules)
>data1.tran<-as(data1,"transactions")
```

以上の操作で作成したデータを用いると、アイテムの集計やルール抽出などの操作を行うことが可能である。関数`itemFrequencyPlot`を次のように用いると、図1のような語の頻度の棒グラフが返される。

```
>itemFrequencyPlot(data1.tran,
type="absolute",horiz=TRUE,
col="lightblue",cex=0.7)
```

図1 語の頻度の棒グラフ



関数`apriori`を用いてルールの集計と支持度などを計算するコマンドを次に示す。コマンドを実行すると、結果の要約が返される。その中に、集計されたルールの総数が含まれている。このデータにおいては、ルールが26,921個集計されている。

```
>data.ap<-apriori(data1.tran)
```

```
parameter specification:
confidence minval smax arem aval originalSupport support minlen
0.8 0.1 1 none FALSE TRUE 0.1 1
```

```
algorithmic control:
filter tree heap memopt load sort verbose
0.1 TRUE TRUE FALSE TRUE 2 TRUE
```

```
apriori - find association rules with the apriori algorithm
version 4.21 (2004.05.03) (c) 1996-2004 Christian Borgelt
set item appearances ... [0 item(s)] done [0.02s].
set transactions ... [43 item(s), 10 transaction(s)] done [0.00s].
sorting and recoding items ... [43 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [26921 rule(s)] done [0.02s].
creating S4 object ... done [0.01s].
```

集計された結果は、関数`SORT`と`inspect`を組み合わせて操作することができる。関数`SORT`はルールを何らかの条件で並べ替える関数であり、`inspect`はルールを返す関数である。その使用例のコマンドと検索された結果の上位10個を次に示す。

```
>data.ap1<- SORT(data.ap,by="support")
>inspect(head(data.ap1,n=10))
```

	lhs	rhs	support	confidence	lift
1	{安く}	=> {学費}	0.3	1	1.428571
2	{授業}	=> {欲しい}	0.2	1	1.666667
3	{いる}	=> {学費}	0.2	1	1.428571
4	{いる}	=> {欲しい}	0.2	1	1.666667
5	{安く, 欲しい}	=> {学費}	0.2	1	1.428571
6	{いる, 学費}	=> {欲しい}	0.2	1	1.666667
7	{いる, 欲しい}	=> {学費}	0.2	1	1.428571
8	{軽減}	=> {学費}	0.1	1	1.428571
9	{削減}	=> {ロッカー}	0.1	1	5.000000
10	{削減}	=> {学費}	0.1	1	1.428571

また、関数`subset`を用いて、条件にマッチしているルールのみを抽出することも可能である。例として、ルールの右辺が「安く」という語になっているルールのみを抽出するコマンドと結果を次に示す。

```
>data.ap1 <- subset(data.ap,
subset = rhs %in% "安く")
>inspect(SORT(data.ap1))
```

	lhs	rhs	support	confidence	lift
1	{もっと}	=> {安く}	0.1	1	3.333333
2	{もう少し}	=> {安く}	0.1	1	3.333333
3	{もっと, 学費}	=> {安く}	0.1	1	3.333333
4	{もっと, 欲しい}	=> {安く}	0.1	1	3.333333
5	{もう少し, 学費}	=> {安く}	0.1	1	3.333333
6	{もう少し, 欲しい}	=> {安く}	0.1	1	3.333333
7	{下さい, 学費}	=> {安く}	0.1	1	3.333333
8	{もっと, 学費, 欲しい}	=> {安く}	0.1	1	3.333333
9	{もう少し, 学費, 欲しい}	=> {安く}	0.1	1	3.333333

このように、アソシエーション分析は、ケースごとに用いられている語の組み合わせを返す。これらは基本的には語（あるいは文節）の共起関係であるが、語の前後の関係に関する情報はない。したがって、共起関係について分析する視点では、語の前後の関係が示されるネットワーク分析法と比べると、メリットが見られない。しかし、アンケート調査結果の分析において、自由回答文とアンケート調査の他の項目を選択した情報とをリンクして分析する場合には、アソシエーション分析法を用いた方が便利であろう。

2. 潜在的意味解析

近年のテキストデータ分析に関連する書物では、潜在的意味解析 (LSA: Latent Semantic Analysis)、あるいは潜在的意味インデキシング (LSI: Latent Semantic Indexing) という技法が強調されている。前者はテキストマイニングの分野で用いられている用語であり、後者は情報検索の分野で用いられている用語である。両者とも「意味」という語が含まれているが、言語の意味情報を用いたテキスト解析のことではなく、行列の特異値分解 (SVD: Singular Value Decomposition) という手法を用いて、高次元のデータを低次元へ射影することにすぎない。

主成分分析では、分散共分散行列、あるいは相関係数行列の固有値と固有ベクトルを求めている。分散共分散行列、相関係数行列や距離行列は正方形の対称行列であるため、低次元に射影する方法としては固有値と固有ベクトルが有効である。

しかし、データ行列の行数と列数が同じではない一般の $n \times p$ 行列は、直接固有値を求

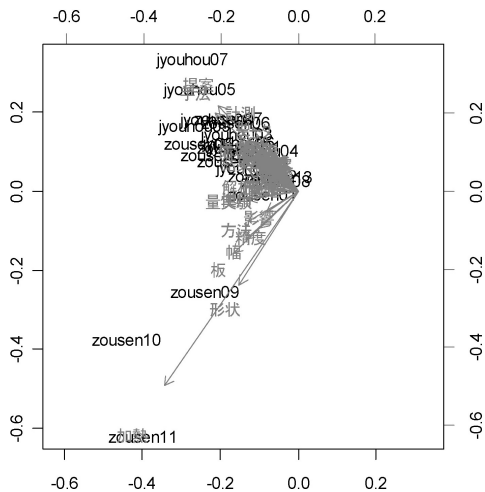
めることが不可能であるため、特異値分解法が用いられている。特異値分解は、 $n \times p$ の行列を次の式のように3つの行列に分解する。

$$A_{n \times p} = U_{n \times k} \Lambda_{k \times k} V_{p \times k}$$

式の中の U を左特異行列、 V を右特異行列と呼ぶ。 Λ (ラムダと呼ぶ) を特異値行列と呼ぶ。特異値行列は対角線行列であり、その要素を特異値と呼び、大きい順に並べてある ($\lambda_1 > \lambda_2 > \dots > \lambda_k$)。これらは、それぞれ主成分分析における主成分得点、主成分、固有値に対応する。特異値分解アルゴリズムは、固有値の問題の場合と同じく、特異値が大きい順に並べられており、それに対応する左特異行列と右特異行列は、左から右の方向に並べられている。したがって、1～3次元で分析を行う場合は、特異行列 U 、 V の第1～3列の値 (ベクトル) を用いればよい。

R には、特異値分解を行う関数 `svd` がある。本連載第67回 (2009年2月号) で紹介した主成分分析と対応分析に用いたデータの特異値分解を行った左特異行列と右特異行列の第1、2ベクトルのバイプロットを図2に示す。この結果は、主成分分析および対応分析のバイプロットよりテキストの特徴が分析しがたい。かつ、データの散布図が「く」の字の逆の形をしていることに注意して欲しい。これはこのデータに限る問題ではなく、このような結果がしばしば得られる。このような現象をデータ解析の分野では、馬蹄問題、あるいはアーチ問題とも呼ぶ。データ解析においては、決して望ましい現象ではない。このようなことから、主成分分析や対応分析で解決できる問題は、特異値分解の方法を用いる必要はない。

図2 特異値分解のバイプロット



主成分分析および対応分析と類似の方法として、因子分析と独立成分分析という方法がある。因子分析は、データの項目間の相関関係を用いて、関連性が強い項目を共通因子としてまとめる方法である。通常、得られた結果は主成分分析と大きな差はない。

独立成分分析方法は、元々、信号処理における信号とノイズを分離する方法として開発されたが、次第に汎用のデータ解析方法として用いられるようになった。独立成分分析をテキスト分析に用いた例も散見される。

近年、カーネル法によるデータ解析が注目されている。通常のカーネル主成分は基本的には古典的多次元尺度法と同じである。テキストの特徴の分析に用いるためには、カーネルトリックについて工夫が必要であろう。

また、人間の神経回路をモデル化したニューラルネットワークのアルゴリズムを用いた自己組織化マップという方法も、テキストマイニングに用いられている。

3. 評判分析

企業におけるテキストマイニングの事例としては、評判分析が多く見られる。評判分析とは、テキストの中から商品やイベントなどに関する顧客やユーザの生の評価を抽出して分析することである。評判分析は、評判に関わる語や語句を用いて、クラスター分析や評価の推移などの分析を行う。クラスター分析では良い評価、悪い評価、どちらもともいえない、などにグルーピングし、評価の推移では、時系列の分析手法で評価がどのように変化しているかを分析する。したがって、分析の方法としては、すでに紹介した方法を組み合わせて使用することになる。

このような分析を行う際に重要なことは、テキストに用いられた語や語句をどのようにグルーピング（あるいはカテゴリ化）するかである。機械的に処理するためには、どのような語や語句が良い評判（ポジティブ）であるか、悪い評判（ネガティブ）であるか、どのような要望と期待があるかを機械的に識別可能な状態にしなければならない。

第1節のアンケート調査の自由回答文を例にすると、「学費を下げ」「学費安くして下さい」「学費が高い」「学費軽減」「学費削減」などは表現が異なるが、全て同じ意味を表している。アソシエーション分析の説明のように機械的に形態素解析や構文解析をした結果を分析するだけでは、的確な情報を抽出することが難しいが、これらを1つの項目にグルーピングすることで、結果分析がしやすくなることはいうまでもない。

このような語や語句のグルーピングには、語や語句の意味情報を用いるので、テキストマイニングシステムでは分類語彙表（シソー

ラス辞書)のような辞書が必要である。商業用のテキストマイニングツールには、一般的にはシソーラス辞書、あるいは辞書作成や登録の機能が備わっている。ツールKH Coderでは、コーディング・ルールファイルに単語を論理和(or)の関係で記述する方法でグルーピングした項目を集計することができる。ツールMLTPでは、語や語句の論理演算(or, and)のリストを作成し、複数のテキストにおけるデータ行列を集計する機能をタブWord Listに備えている。

4. 意味処理と辞書

前節の内容とも関連しているが、われわれ人間は文を読み、その意味情報を分析する。しかし、現段階のテキスト解析やテキストマイニングは、基本的には形態素解析、構文解析の結果を用いてテキストに現れている要素やそれらの共起関係などを集計して分析する。このような方法では、意味情報を分析するには限界がある。機械による意味情報の処理は、自然言語処理における大きな研究課題である。意味情報を処理する主な方法としては、シソーラス辞書、概念辞書、文脈情報などを用いることが考えられる。

シソーラス辞書としては市販されているものもあり、研究用であれば、(独)国立国語研究所で作成した分類語彙表がある。

概念辞書とは、単語間の羅列だけではなく、その用法や他の単語との関連などを記録した辞書である。概念辞書としては、語の上下関係を記述したEDR (Electronic Dictionary Research) 電子化辞書がある。

EDR電子化辞書は、基盤技術研究促進センターと8つのコンピューターメーカーの共同

出資のもとに進められた知的情報処理のソフトウェア開発を目的とするプロジェクトにより開発された辞書群である。EDR電子化辞書は、単語辞書、対訳辞書、概念辞書、共起辞書、専門用語辞書、EDRコーパスの6つから構成されている。現在、その研究開発は(独)情報通信研究機構に移管されている。EDRの概念辞書は約41万の概念に対して、それらの間の上位下位関係を記述したものである。上位下位関係とは、概念間の包含関係である。例えば、人間、犬、猫、鳥などは動物である。この場合、動物と犬の関係は、動物が上位であり、犬は下位である。

より知的なテキストマイニングシステムを構築するためには、このような様々な辞書を取り入れた研究開発が必要である。近年、市販されている商業用のテキストマイニングツールにおける、いわゆる意味解析は、シソーラス辞書などを導入した初期的なものであり、本格的な意味処理が可能なシステムの開発にはかなり時間がかかるであろう。

テキストマイニングは、本連載で取り上げた内容以外に、検索システムにおける情報の抽出、テキスト要約などの内容も包含しているが、本連載では統計的テキスト解析を中心としているため、これらに関しては触れていない。これらの参考文献としては、情報抽出に関しては徳永(1999)、北・津田・獅々堀(2002)があり、テキストの要約に関しては奥村・難波(2005)がある。

*参考文献

- [1] 徳永健伸(1999): 情報検索と言語処理: 東京大学出版会.
- [2] 北 研二・津田和彦・獅々堀正幹(2002): 情報検索アルゴリズム: 共立出版.
- [3] 奥村 学・難波英嗣(2005): テキスト自動要約: オーム社.