

# ノイズを含むデータに対する FP木を用いた頻出飽和アイテム集合マイニング

Frequent Closed Itemset Mining from Distorted Data Using FP-trees

成田 和世<sup>\*</sup> 北川 博之<sup>\*</sup>

Kazuyo NARITA Hiroyuki KITAGAWA

情報技術の発展につれて、巨大なデータから特徴や法則を発見するデータマイニングはますます重要視され、様々な研究がなされている。相関ルールマイニングにおいては、近年、真超集合を持たない頻出アイテム集合(頻出飽和アイテム集合, frequent closed itemsets)を見つけ出すことが重要であるとして注目されつつある[1, 2, 3]。一方で、情報に対するプライバシーへの関心が高まっていることを背景に、プライバシーを考慮したマイニングも様々に研究されてきた[3, 4]。そこで本稿では、プライバシーを保護するためにノイズを入れたデータからいかにして頻出飽和アイテム集合を求めるかを検討し、マイニングの手法を提案する。プライバシーを考慮に入れた相関ルールマイニング研究で、頻出飽和アイテム集合を見つけ出す手法はこれまで提案されていない。本提案手法では、マイニングの処理の効率化を図るため、コンパクトなデータ構造で知られるJ. HanらのFP木[1, 5]を用いる。

As the recent advance of information technology, data mining, which discovers features or rules from large data, has become more important and researched variedly. In association rule mining, frequent closed itemset mining, which mines only frequent itemsets having no proper superset, is attracting a lot of research interest in recent years. The study fields have extended to privacy preserving mining against the background of increased respect for privacy. In this paper, we study how we should mine frequent closed itemsets from privacy preserving data, and propose a mining algorithm. There has so far been no proposal on mining frequent closed itemsets from privacy preserving data. Aiming more efficiency, we use J. Han's FP-tree [1, 5] which is known as a compact data structure.

## 1. はじめに

情報技術の発展につれて、巨大なデータから特徴や法則を発見するデータマイニングはますます重要視され、様々な研究がなされている。アイテム間の相関性を発見する相関ル

<sup>\*</sup> 学生会員 筑波大学大学院理工学研究科修士課程

[narita@kde.cs.tsukuba.ac.jp](mailto:narita@kde.cs.tsukuba.ac.jp)

<sup>\*</sup> 正会員 筑波大学大学院システム情報工学研究科, 計算科学研究センター

[kitagawa@cs.tsukuba.ac.jp](mailto:kitagawa@cs.tsukuba.ac.jp)

ルマイニングで近年注目されている頻出飽和アイテム集合(frequent closed itemsets)マイニングは、等しいサポート値を持つ真超集合を持たない頻出アイテム集合をすべて発見することであり、これまで様々なアルゴリズムが提案されている[1, 2, 3]。一方、情報に対するプライバシーへの関心が高まっていることを背景に、プライバシーを保護するために故意にノイズを入れたデータに対するマイニングも研究されている[3, 4]。

本稿では、プライバシー保護のためにノイズを入れたデータから、どのようにして頻出飽和アイテム集合を発見するかを検討する。プライバシーを考慮に入れた相関ルールマイニングにおいて、頻出飽和アイテム集合を発見する手法はこれまでのところ提案されていない。本研究では、プライバシー保護のためのノイズ入りデータとして、R. Agrawalらの提案[3]に基づくものを対象とする。また、マイニングの処理の効率化を図るため、コンパクトなデータ構造で知られるJ. HanらのFP木[1, 5]を用いる。

本稿の構成は次の通りである。まず2.で本研究に関連するノイズ入りデータ、FP木、頻出飽和アイテム集合について述べ、3.でFP木を用いた提案手法を説明し、4.で実験とその結果について述べる。最後に、5.でまとめと今後の課題について述べる。

## 2. 関連研究

### 2.1 ノイズ入りデータと元データへの復元法

本研究では、ノイズ入りデータとして、R. Agrawalの提案[3]に基づくものを対象とする。[3]はプライバシー保護のために、個人情報を含むデータに確率的にノイズを入れる手法 *cut-and-paste randomization* を考案し、そこからノイズを入れる前の情報をマイニングするフレームワークを提案している。*cut-and-paste randomization* とは以下のような手法である。

[Definition 1] ノイズのない元のトランザクションデータベース  $DB$  に含まれるサイズ  $m$  のトランザクション  $t_i$  に対して、2つのパラメタ  $\rho_m (0,1)$ 、整数  $K_m > 0$  が与えられるとする。 $DB$  における全てのアイテムの集合を  $I$  としたとき、トランザクション  $t_i$  に対して以下の処理を行うことでノイズを含むトランザクション  $t_i$  を得る操作を *cut-and-paste randomization* という。

1.  $0 < j < K_m$  となる整数  $j$  を一様ランダムに選択する。その際、 $j > m$  ならば  $j = m$  とする。
2. ノイズのないトランザクション  $t_i$  から一様ランダムに  $j$  個のアイテムを選択し、それらのアイテムを含むトランザクション  $t_i'$  を得る。
3. 選択した  $j$  個のアイテム以外の各アイテム  $a \in I$  を、各々確率  $\rho_m$  で  $t_i'$  に加える。

以上の操作でノイズを入れたデータから元のデータの頻出アイテム集合を発見するために、[3]ではサポート値を確率によって推定している。

サポート値の推定方法を説明するために、まずは必要な定義を記述する。アイテム集合  $A$  の濃度を  $card(A)=k$ 、サポートを  $supp(A)$ 、データベース中のトランザクションの総数を  $N$  とする。また、ノイズのないデータベース  $DB$  のトランザクション集合を  $T$ 、 $DB$  を *cut-and-paste randomization* によってノイズ入りデータベース  $DB'$  にしたとき、 $DB'$  のトランザクション集合を  $T'$  とする。

[Definition 2] アイテム集合  $A$  と整数  $h$  が与えられたとき,  $card(A \cap t) = h$  であるトランザクション  $t$  の全てのトランザクションに対する割合を,  $A$  の共通集合濃度  $h$  の部分サポート (partial support) と呼び, 次で表される:

$$supp_h^T(A) = \frac{card(\{t \in T \mid card(A \cap t) = h\})}{N}$$

また,  $card(A \cap t) = h$  かつ  $card(t) = m$  であるトランザクション  $t \in T$  が, cut-and-paste randomization によって  $card(A \cap t') = h'$  であるトランザクション  $t' \in T'$  となる確率は,

$$p_k^m[h \rightarrow h'] = P[card(t' \cap A) = h' \mid card(t \cap A) = h]$$

であり, パラメタ確率  $\rho_m$  と整数  $K_m$  を用いて求めることが可能である[3]. 以上から,  $DB$  における  $A$  の部分サポートを  $s_h = supp_h^T(A)$ ,  $DB'$  における  $A$  の部分サポートを  $s_{h'} = supp_{h'}^T(A)$  と表したとき, ベクトル  $\vec{s} = (s_0, s_1, \dots, s_k)'$  の期待値  $E\vec{s}'$  はベクトル  $\vec{s} = (s_0, s_1, \dots, s_k)'$  を用いて次式で表される.

$$E\vec{s}' = \mathbf{P} \cdot \vec{s} \quad (1)$$

$\mathbf{P}$  は各要素  $P_{hh'}$  が  $p_k^m[h \rightarrow h']$  の  $(k+1) \times (k+1)$  行列である.  $card(A) = k$  であることより  $A$  のサポートは  $supp_k^T(A)$  であるので, 式(1)を解くことにより  $supp_k^T(A)$  の推定値(推定サポート)が求められる.[3]によると, アイテム集合  $A$  の推定サポート  $s_{est}(A)$  は式(2)となる.

$$s_{est}(A) = \sum_{h=0}^k s_{h'} \cdot q[k \leftarrow h']; \quad \text{where } q[h \leftarrow h'] = Q_{hh'}, \quad \mathbf{Q} = \mathbf{P}^{-1} \quad (2)$$

以上から, ノイズ入りデータから任意のアイテム集合の部分サポートを全て求め, かつ確率  $P_{hh'}$  を計算することで, 元のデータにおけるサポートの推定値が得られることが分かる.

## 2.2 FP 木

表1 データベースとアイテムリスト

Table 1 Database and frequent ordered item list

TID	アイテム集合	リスト
101	A,C,D,F,H	A,D,F
102	D,E,F,J	D,E,F
103	A,B,E,G,I	A,B,E,G
104	A,B,D,G	A,B,D,G

本研究ではノイズを含むデータから頻出飽和アイテム集合を発見するのに, FP 木を用いた深さ優先探索型のアルゴリズムを提案する.

FP 木とは, データベース中の似たパターンをもつトランザクションをひとつのパスとしてマージすることでデータをコンパクトに圧縮するデータ木構造である[5].

表1の1, 2列目はトランザクションデータベース, 3列目は最小サポートを2と設定した場合に, 各トランザクション中の最小サポート条件を満たすアイテムを, サポートの降順に列挙したリストである. このリストから FP 木を構築すると図1のようなになる. 各リストが FP 木の各パスに対応している. 例えば2行目は FP 木の最右のパスに対応している. 複数のリストの共通接頭辞は, FP 木では一つのパスにマージして表す. ノードに付いている値はアイテムの出現回数である. また, FP 木は木構造と同時にヘッダテーブルと呼ばれる表を持つ. ヘッダテーブルの各エントリには, 最小サポート条件を満たすアイテムと, それをラベルに持つ最初のノードへのポインタが入っており, そのノードは同じラ

ベルを持つ次のノードを指すポインタを持つ. [5]は, 最小サポート以上の回数出現するアイテム集合(頻出アイテム集合)をマイニングするアルゴリズムとして FP 木を用いた手法を提案している. その概要は次の通りである.

ヘッダテーブルの対応エントリから, FP 木  $fpt$  上で頻出アイテム  $a$  をラベルに持つノード  $ndi$  を探す.  $ndi$  からパスを上へ調べ,  $a$  の接頭辞パス  $p_i$  を全ての  $i$  について得る.  $p_i$  に出現するアイテムの具合を一つのトランザクションと見なして, 全ての  $p_i$  に含まれるアイテムのサポートを調べる. 最小サポート未満のアイテムを刈り取った  $p_i$  から FP 木  $cond\_fpt$  を作り, この木に対して先ほどと同様の処理を行う. 以下, 全ての頻出アイテムについて, それを接尾辞に持つ頻出アイテム集合をマイニングするまで, 再帰的に処理が続く.

図2は図1でGに対して上述の処理を行ったときに構築される FP 木である(最小サポートは2). データベースから構築される FP 木をグローバル FP 木と呼ぶのに対して, FP 木から構築されるこのような木を条件付き FP 木という.

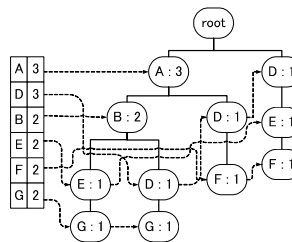


図1 FP 木  
Fig.1 FP-tree

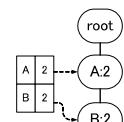


図2 条件付き FP 木  
Fig. 2 The conditional FP-tree

## 2.3 頻出飽和アイテム集合

頻出飽和アイテム集合とは以下の条件を満たすアイテム集合である.

[Definition 3] アイテム集合  $A$  が頻出アイテム集合であり, かつ  $supp(A) = supp(A')$  である真超集合  $A'$   $A$  が存在しないとき,  $A$  には飽和性があると言い, このようなアイテム集合を頻出飽和アイテム集合と呼ぶ.

上記定義より, 頻出飽和アイテム集合の全体集合は全ての頻出アイテム集合の情報を完全に保持しているながら, 頻出アイテム集合の全体集合よりもコンパクトであることが分かる.

頻出飽和アイテム集合をマイニングする主な深さ優先型のアルゴリズムの中で, 空間的なコストを削減するために導入されている技術を説明する.

- (1) **item margin**: 頻出アイテム集合  $X$  が出現する全てのトランザクションに  $card(X \cap Y) = 0$  であるアイテム集合  $Y$  が出現し, かつ,  $X$  が出現する全てのトランザクションに  $card(X \cap Y) = 0$  であるアイテム集合  $Y'$   $Y$  が存在しないなら,  $X \setminus Y$  は頻出飽和アイテム集合である.
- (2) **sub-itemset pruning**: 新しく発見した頻出アイテム集合  $X$  が, 発見済みの頻出飽和アイテム集合  $Y$  の部分集合で, かつ  $supp(X) = supp(Y)$  ならば,  $X$  と  $supp(X)$  と等しいサポートを持つ  $X$  の部分集合は, 頻出飽和アイテム集合になり得ない.

[1]では[5]の手法を, 頻出飽和アイテム集合を発見するために改良し, 同時に次の(3)を提案している.

- (3) **item skipping**: 条件付き FP 木を構築するとき, その木におけるアイテムのサポートが, 異なるレベルの FP 木における同一アイテムのサポートと等しいならば, 上位

のレベルにある木のヘッダテーブルから、そのアイテムの情報を刈り取ることができる。

以上3つは本提案手法にも導入されている。今、[1]のアルゴリズムで図1のFP木をマイニングすることを考える。図1から図2の条件付きFP木を構築するまでの過程は、先に述べた[5]のアルゴリズムと同じである。図2より、Gが出現するトランザクション中ではA、Bのサポートは共に2である。これはGのサポートに等しい。すなわちitem marginより頻出飽和アイテム集合{A, B, G} (サポートは2)が発見される。また、図1, 2ではBのサポートが等しいので、Bを接尾辞に持つ頻出飽和アイテム集合をマイニングする結果はどちらの木でも同じである。そこでitem skippingが適用され、図2より上位のレベルにある図1では、Bについてのマイニング処理は飛ばされる。仮にitem skippingなしにマイニングを続けていくと、サポートが2のアイテム集合{A, B}を接尾辞に持つ頻出アイテム集合を探すことになるが、この場合はsub-itemset pruningが適用され{A, B}についての処理はそこで終了する。

### 3. 提案手法

本節では提案手法の説明をする。まず3.1, 3.2節でノイズ入りデータから頻出飽和アイテム集合をマイニングする時に発生する問題点を明らかにし、それぞれ対応策を述べる。最後に3.3節でアルゴリズムをまとめる。

#### 3.1 候補集合の絞り込み

2.1節で、ノイズ入りのデータから元のデータ上のアイテム集合のサポートを推定する方法を説明した。しかし、このようにして得られた推定サポートは相関ルールマイニングにおいて重要なアприオリ特性を満たさない。アприオリ特性は、任意のアイテム集合のサポートはその部分集合のサポートを上回らないというアイテム集合の性質のことであり、現在までに提案されている主要な相関ルールマイニングはこの特性を利用して、頻出アイテム集合の候補集合を効率的に刈り取っている。データから得られるアイテム集合は膨大な数となり、それら全てについて推定サポートを計算するのは高コストであることから、推定サポートがアприオリ特性を満たさないこの場合も、アイテム集合の何らかの刈り取りは欠かせない。そこで、推定サポートが最小サポート未満のアイテム集合は、元のデータにおいても最小サポート条件を満たさないとみなす。さらに、最小サポートを満たさないアイテム集合があれば、それを含むどんな超集合も最小サポートを満たさないと仮定して、そのアイテム集合を刈り取ることとする。このとき次の2つのエラーが起こる可能性がある。最小サポート条件を満たさない集合が頻出であるとして検出される場合と、最小サポート条件を満たす集合が検出されない場合である。前者のエラーのように、検出されるべきではないのに検出されるアイテム集合をfalse positive、後者のように検出されるべきものが検出されないアイテム集合をfalse negativeと呼ぶ。[3]はfalse negativeが発生する後者のエラーがより問題であるとして、それに対応するために推定サポート  $s_{est}$  の不偏分散  $\sigma^2$  を式(3)のように導入している。

$$\sigma^2 = \frac{1}{N} \sum_{h=0}^k s'_h (q[k \leftarrow h]^2 - q[k \leftarrow h']). \quad (3)$$

このとき、[3]は推定サポートをこの標準偏差だけ引き上げた修正値  $s_{modify} = s_{est} + \sigma$  が最小サポート未満であるアイテム

集合を刈り取っている。本研究でもそれに従う。

#### 3.2 飽和性の判定

ある集合に飽和性があるどうかは、同値のサポートを持つ超集合が存在しないことを調べればよい。しかし、ノイズ入りデータから求めた推定サポートは、元のデータから求められる真のサポートから、ある程度の差が生じていると考えられる。そこで我々は、2つのアイテム集合のサポートに差があるかどうかを、推定サポートから次のように判定する。

既に発見済みの頻出飽和アイテム集合を  $X$ 、新しく発見した頻出アイテム集合を  $Y$  とし、これら2つの集合間には  $X \supset Y$  が成り立つと仮定する。また、 $Y$  の真のサポートを  $supp(Y)$ 、 $X$  の真のサポートを  $supp(X)$  と記述する。さらに、 $X$  と  $Y$  の2つの集合のサポートの推定値に関する確率変数を  $s_x, s_y$  としたとき、2つの確率変数の差の平均  $E(|s_x - s_y|)$  が著しく大きい場合は、2つの確率変数に差があると考えることとする。「著しく大きい」と判断する基準には、2確率変数の差の標準偏差  $\sqrt{V_{xy}}$  を用いる。すなわち、差の平均  $E(|s_x - s_y|)$  が差の標準偏差  $\sqrt{V_{xy}}$  を上回るとき、真のサポート  $supp(X)$ 、 $supp(Y)$  が  $supp(X) < supp(Y)$  であると判定する。ここで、 $E(|s_x - s_y|) = E(s_x) - E(s_y)$  であり、 $E(s_x), E(s_y)$  は式(1)から式(2)を導出する過程で、それぞれ集合  $X, Y$  の推定サポート  $s_{est}(X), s_{est}(Y)$  であることが分かっている。判定条件をより具体化するため、差の分散  $V_{xy}$  を以下で導く。

$$V_{xy} = E(((s_x - s_y) - (s_{est}(X) - s_{est}(Y))))^2) = \sigma_x^2 + \sigma_y^2$$

ここで  $\sigma_x^2, \sigma_y^2$  は各々  $s_{est}(X), s_{est}(Y)$  の不偏分散である。上式で得られる分散  $V_{xy}$  を用いた飽和性の判定条件をまとめる。

$$|\sqrt{V_{xy}}| < |s_{est}(X) - s_{est}(Y)| \Rightarrow \text{supp}^T(X) < \text{supp}^T(Y)$$

この式から2通りの判定条件が得られるが、本稿では、判定条件  $s_{est}(X) < s_{est}(Y) + \sqrt{V_{xy}}$  を用いて実験を行った。

#### 3.3 提案アルゴリズム

**Algorithm:** 頻出飽和アイテム集合マイニング

**Input:** ノイズ入りデータベース  $DB'$ , トランザクションの各サイズ  $m$  に対応するパラメタ  $\rho_m, Km$ , 最小サポート  $min\_sup$

**Output:** 頻出飽和アイテム集合の完全集合  $CF$

**Method:**

1.  $DB'$  をスキャンし、各アイテムのサポートをカウントする。 $min\_sup$  を下回るサポートを持つアイテムは刈り取り、残ったアイテムをサポートの降順に並べたリスト  $\ell\_list$  を作る。
2.  $DB'$  をスキャンし、 $\ell\_list$  を用いてグローバルなFP木  $fpt$  を構築する。
3. ヘッダテーブルの各エントリから  $fpt$  を辿り、全ての頻出アイテムに対してマイニングを行う。item margin, sub-itemset pruning を行いながら条件付きFP木を再帰的に構築し、ヘッダテーブルに対してitem skippingを行いながら頻出アイテム集合を見つけていく。サポートを推定するときに必要な部分サポートは、ヘッダテーブルを辿ることで求める。推定サポートと  $\sigma^2$  を求め、修正値  $s_{modify}$  を得る。 $s_{modify} = min\_sup + \sigma$  ならば、結果集合  $result$  をスキャンし、目的のアイテム集合が  $result$  内の全ての集合に対して飽和性を満たすなら  $result$  に加える。
4. ヘッダテーブルの全てのエントリについて  $fpt$  をマイニングしたら、 $result$  から  $CF$  を作って出力する。

上記の処理の通り,本手法ではノイズ入りデータを対象としているため,推定サポートを計算する.これは,ノイズを含まないデータを対象に頻出飽和アイテム集合をマイニングする[1]の手法では行われない.本手法では,計算に必要な部分サポートの数え上げを,アイテム集合ごとにグローバルFP木を一回スキャンすることで行っている.

#### 4. 実験

本提案手法の評価実験を行った.提案手法の実装はJavaで行い,1GメモリのLinuxデスクトップPCを使用した.

##### 4.1 データセット

本実験では,3-アイテム集合までの小規模アイテム集合をマイニングすることとした.トランザクションの濃度は全て3である.実験に使用したデータは全てIBM Almadenによるデータ生成器[6]で生成した.いくつか得られた結果の中から,例として,トランザクション数31K,総アイテム数60,  $K=7$ ,  $\rho=0.24$ とした場合について示す.実験ではこのデータにcut-and-paste randomizationを用いてノイズを入れ,最小サポートを0.6%として提案手法によりマイニングを行った.

##### 4.2 結果

表2 実験結果

Table 2 Experimental result

アイテム集合の濃度	True F.C.I	True positive	False negative	False positive
1	47	47	0	9
2	275	212	63	94
3	56	41	15	26

表2にマイニング結果を示す.1列目はアイテム集合の各濃度を示している.2列目には元データの頻出飽和アイテム集合(true F.C.I)の数が,濃度別に表されている.3列目は提案手法でマイニングしたもののうち,True F.C.Iと一致する集合(true positive)の数である.4列目,5列目はそれぞれ,false negative, false positiveの数である.これらの結果がどれだけ正しいかを測るために,再現率,適合率,F値をそれぞれ求めた.その結果を表3に示す.

表3 再現率, 適合率, F値

Table 3 再現率, 適合率 and F-measure

アイテム集合の濃度	再現率	適合率	F値
1	100.0	83.9	91.2
2	77.1	69.3	73.9
3	73.2	61.2	66.6

#### 結果の評価

1-アイテム集合の再現率は100%で,適合率は80%程度となった.2-アイテム集合については,再現率がおよそ80%,適合率がおよそ70%で,3-アイテム集合は,再現率が70%強,適合率が60%強となった.2-アイテム集合,3-アイテム集合の評価が下がった要因として,現時点ではトランザクションの数の差が考えられる.式(3)より推定サポートの分散はトランザクションの数の数に依存している.部分サポートの値にもよるが,一般に,トランザクションの数が小さいほど分散は大きくなる. $\sigma$ により大きく見積もられた推定サポートの修正値により,false positiveが増え,そのため飽和性の判定時にエラーが生じやすくなり,false negativeも増えたのでは

ないかと考える.今後,より大きなデータを用いた実験や,そのほかのマイニングの結果へ影響を与える要因の調査をする予定である.

#### 5. まとめと今後の課題

本稿では,プライバシーを保護する目的で,ノイズを入れたデータに対して,頻出飽和アイテム集合をどのように推定するかを検討し,FP木を用いたマイニング手法を提案した.

今後の課題として,より多様なデータを用いて実験を続けること,その際,刈り取り判定と飽和性の判定に使用する条件を変えて試みることで,マイニングの評価結果がデータのどの部分に影響されるのか明らかにすることなどが上げられる.

#### [謝辞]

本研究の一部は,日本学術振興会科学研究費補助金基盤研究(B)(#15300027),ならびにCREST「自立連合型基盤システムの構築」の助成による.

#### [文献]

- [1] J. Wang, J. Han, and J. Pei. : "CLOSET+: Searching for the best strategies for mining frequent closed itemsets", Proc. of the 9th ACM SIGKDD, pp. 236-245 (2003)
- [2] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. : "Discovering frequent closed itemsets for association rules", LNSC 1549, pp. 398-416 (1998)
- [3] A. V. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. : "Privacy preserving mining of association rules", Proc of the 8th ACM SIGKDD, pp. 217-228 (2002)
- [4] S. Rizvi and J. R. Haritsa. : "Maintaining data privacy in association rule mining", Proc of VLDB2002, pp. 682-693 (2002)
- [5] J. Han, J. Pei and Y. Yin. : "Mining frequent patterns without candidate generation", Proc of the 2000 ACM SIGMOD, pp. 1-12 (2000)
- [6] IBM Almaden. Quest synthetic data generation code. <http://www.almaden.ibm.com/cs/quest/syndata.html>

#### 成田 和世 Kazuyo NARITA

筑波大学大学院理工学研究科修士課程在学中. 相関ルールマイニングの研究に従事. ACM SIGMOD 日本支部, 日本データベース学会, 各学生会員.

#### 北川 博之 Hiroyuki KITAGAWA

筑波大学大学院システム情報工学研究科, 計算科学研究センター教授. 1980 東京大学大学院理学系研究科修了, 理学博士(東京大学). 異種情報源統合, 文書データベース, WWWの高度利用などの研究に従事. 著書「データベースシステム」(昭晃堂), 「Unnormalized Relational Data Model」(共著, Springer-Verlag)等. ACM SIGMOD 日本支部長, 日本データベース学会理事. ACM, IEEE-CS, 情報処理学会, 電子情報通信学会, 日本データベース学会, 日本ソフトウェア科学会, 各会員.