情報知識ネットワーク特論
Data Mining 1: Data Mining and Frequent Item Set Mining

Hiroki Arimura, Takuya Kida

Graduate School of Info. Sci. and Tech, Hokkaido University
email: {arim,kida}@ist.hokudai.ac.jp

Slide PDF: http://www-ikn.ist.hokudai.ac.jp/ikn-tokuron/
Oct 2014

目次

1回: データマイニングと頻出集合発見

- データマイニング
- 頻出集合マイニング

ポイント

- データマイニングとは何か？
- 列挙アルゴリズム
- 高速なアルゴリズムの設計

Data Mining

データマイニングとは？

Backgrounds

Data Mining

- Study on efficient “semi-automatic” methods for extracting “**interesting and useful**” patterns and rules from massive data sets
- Emerged in the mid 1990s.
  - Apriori algorithm [Agrawal, Srikant, VLDB1994]
- Potentially, a collection of existing studies.
  - But, emphasis on efficient computation for massive data
- Boundary of Machine Learning, Statistics, and Databases

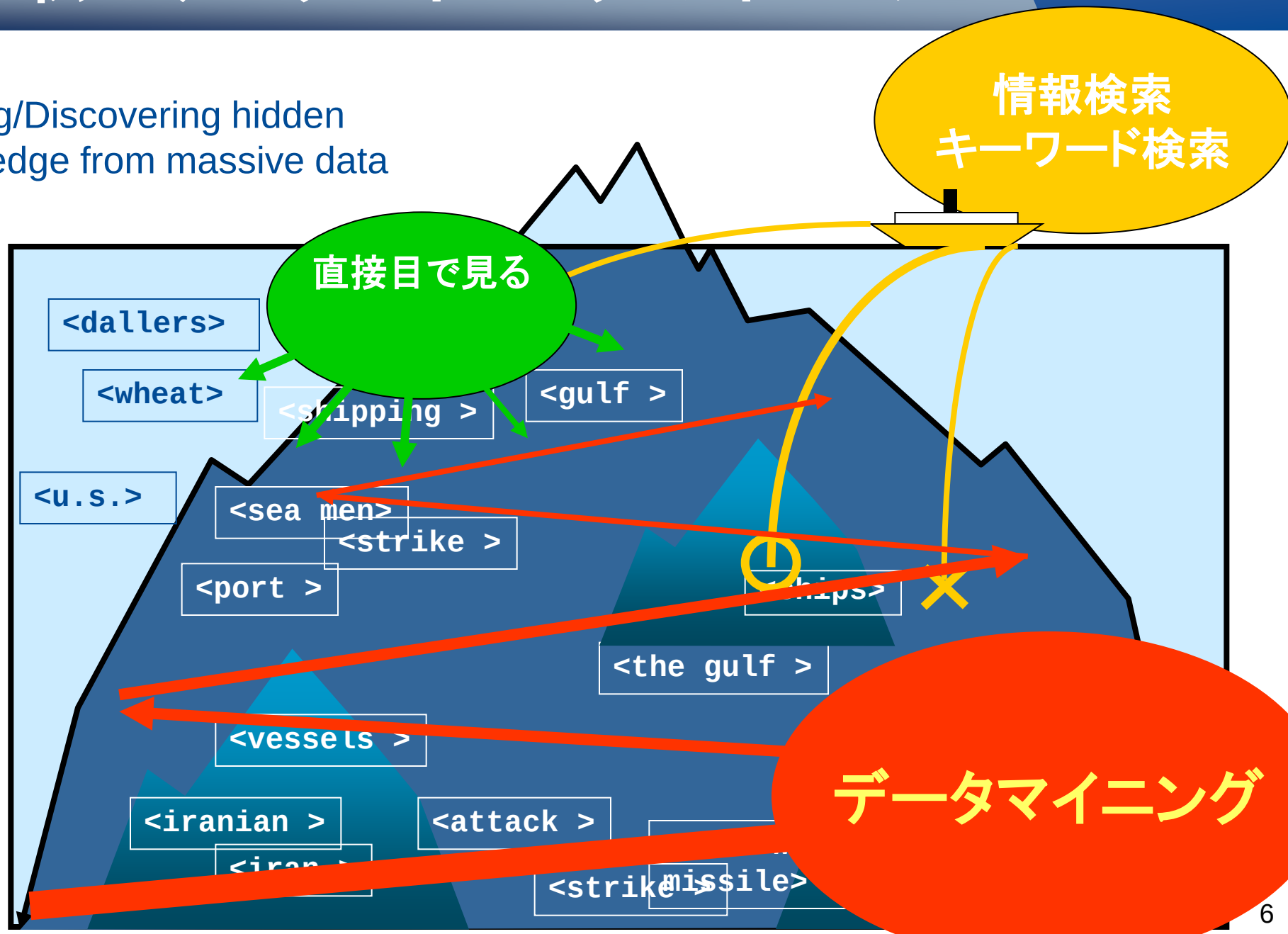
Backgrounds

The whole process of Data Mining

- 1.Understanding the domain of data
- 2.Preprocessing of data sets
- 3. Mining of patterns（Data Mining in narrow sense)
- 4.Analysis of discovered patterns
- 5.Use of the analyzed results

私のデータマイニングのイメージ

Finding/Discovering hidden knowledge from massive data



Backgrounds: 4テラバイトのデータ

4テラバイト = 4,000,000,000,000バイト

- 400字詰めA4原稿用紙で309km^2
  ＝ 山手線の内側の約3倍の面積
- 1秒間に10文字入力すると12,683年
- CD 6153枚 ＝ 音楽として聴くと316日間
- 記憶装置からの読み込みに 15.2時間
- ちょっとした計算(N^2時間)
  2.2 GFLOPS × 32PE のスーパーコンピュータで
  800万年以上かかる。

•高速なアルゴリズムが必要！！！

データマイニングの動向

パターン発見

- トランザクションデータから共通して出現する規則性を発見する
- 頻出パターン発見 [Agrawal et a. ‘94]
- 最適化マイニング [森下 ’96, ’98, ‘00]

予測学習・自動分類

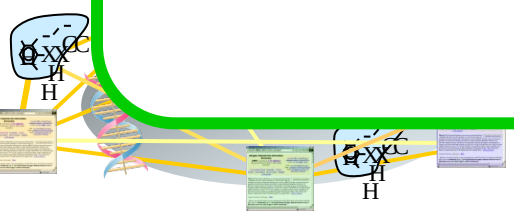
- 不完全なデータから，未知の規則を学習する
- SVM [Vapnik ‘96],
- Boosting [Shapire & Kearns ‘96]
- C4.5 [Quinlan ‘96]

構造マイニング

- 非定型構造データから特徴的な部分構造を規則性を発見する
- グラフマイニング[Washio & Motoda ‘00], [Zaki ’02], [Uno, Asai, Arimura, ’02, ‘03]

クラスタリング

- データを類似したものどうしグルーピングする.
- 大規模・不完全なデータからの高速クラスタリング
- K-means, CLARANS, DBSCAN



確率モデリング

- 高次元大規模データから不確実な現象を予測・モデル化する
- ベイジアンネットワーク [Pearl ’90s]
- HMM [Asai], MCMC, ベイズ推定・MDL・AIC

新しいタイプのデータマイニング

- テキストマイニング
  - 自然言語テキスト
  - 情報抽出
  - 意味マイニング
- ストリームマイニング
  - センサー監視
  - 近似統計処理

情報知識ネットワーク特論
Data Mining 1:
データマイニングと頻出集合発見

有村 博紀, 喜田拓也
北海道大学大学院 情報科学研究科 コンピュータサイエンス専攻
email: {arim,kida}@ist.hokudai.ac.jp
http://www-ikn.ist.hokudai.ac.jp/ikn-tokuron/
http://www-ikn.ist.hokudai.ac.jp/~arim

.

Frequent Itemset Mining Definitions

Backgrounds

Frequent Itemset Mining

- Finding all "frequent" sets of elements (items) appearing no more than σ times in a given transaction data base.
- Introduced by Agrawal and Srikant [VLDB'94]
- One of the most popular data mining problem
- Basis for more complicated / sophisticated data mining problems

動機：結合ルールマイニング

- Association Rule Mining [Agrawal 1993/1994]
  - Finding combination of “items” frequently appearing in a given database

- トランザクションデータ
- バスケットデータ
- 二値データベース

- レコード/タプル
- バスケット

ID	Chips	Mustard	Sausage	Softdrink	Beer
001	1	0	0	0	1
002	1	1	1	1	1
003	1	0	1	0	0
004	0	0	1	0	1
005	0	1	1	1	1
006	1	1	1	0	1
007	1	0	1	1	1
008	1	1	1	0	0
009	1	0	0	1	0
010	0	1	1	0	1

- カラム/属性
- アイテム

トランザクション／レコードの意味
「レコード003の顧客は，ポテトチップとソーセージを一緒に買った」

動機：結合ルールマイニング

- Frequent Itemset **X** = { Mustard, Sausage, Beer }
  - with support/frequency 40%

ID	Chips	Mustard	Sausage	Softdrink	Beer
001	1	0	0	0	1
002	1	1	1	1	1
003	1	0	1	0	0
004	0	0	1	0	1
005	0	1	1	1	1
006	1	1	1	0	1
007	1	0	1	1	1
008	1	1	1	0	0
009	1	0	0	1	0
010	0	1	1	0	1

- **アイテム集合 X**
- **Xの出現リスト**
  Occ(X) = {002, 005, 006, 010}
- **Xの頻度（サポート）**
  freq(X) = |Occ(X)| = 4/10 = 40%

アイテム集合 X の意味
X =「マスタードとソーセージ，ビールを一緒に買う人」が全体の40%いた

頻出アイテム集合マイニング

Frequent Itemset Mining Problem

- Given: A database DB over a set Σ of items, and a number $\sigma \geqq 0$ called “minsup” (minimum support)
- Problem: Find all itemsets $X \subseteq \Sigma$ appearing in no less than σ records of DB

14

Definitions: Database

- A set $\Sigma = \{ 1, ..., n \}$ of items (elements)
- Transaction database
  - A set $T = \{ t_1, ..., t_m \}$ of subsets of Σ
  - Each subset $t \subseteq \Sigma$ is called a tuple (record)

Alphabet of items

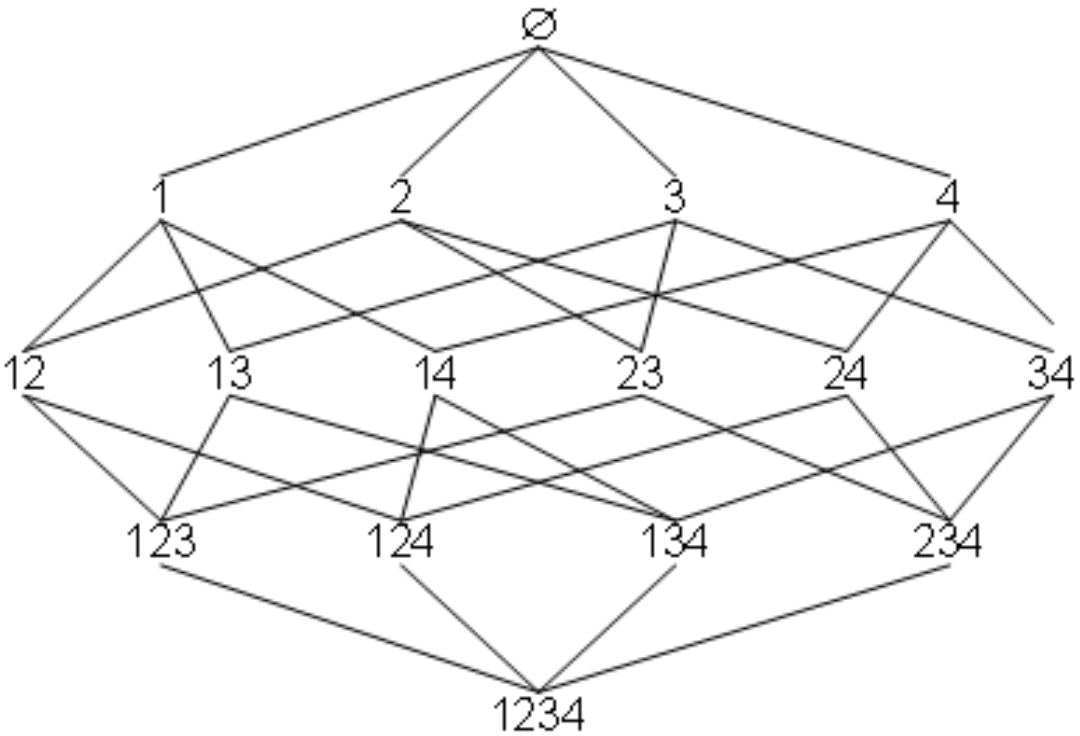
$I = \{1,2,3,4\}$

Transaction database

id	tuples
1	1, 3
2	2, 4
3	1, 2, 3, 4
4	1, 2, 4

Definitions: Itemset lattice

- Item set: any subset $X \subseteq \boldsymbol{\Sigma} = \{ 1, \ldots, n \}$
- (Item) set lattice $\mathcal{L} = (2^{\boldsymbol{\Sigma}}, \subseteq)$
  - The power set $2^{\boldsymbol{\Sigma}} = \{ X : X \subseteq \boldsymbol{\Sigma} \}$
  - The subset relation $\subseteq$ over $2^{\boldsymbol{\Sigma}}$



Example:
The set lattice
for$\Sigma = \{1,2,3,4\}$

Definitions: Frequent sets

- An itemset X appears in a tuple t: $X \subseteq t$
- The occurrence of X in a database ***T***:
  $Occ(X, \boldsymbol{T}) = \{ t \in \boldsymbol{T} : X \subseteq t \}$
- The frequency of X: $Fr(X, T) = |\, Occ(X, T) \,|$
- Minimum support (minsup): an integer $0 \leqq \sigma \leqq |\boldsymbol{T}|$
- X is σ-frequent (frequent) in T if $Fr(X, T) \geqq \sigma$.

Alphabet of items

$I = \{A,B,C,D\}$

Occurrences and frequences of itemsets

Occ(**3**, T) = {1, 3}
Fr(**3**, T) = 2

Occ(**24**, T) = {2, 3, 4},
Fr(**24**, T) = 3

Transaction database

id	tuples
1	1, 3
2	2, 4
3	1, 2, 3, 4
4	1, 2, 4

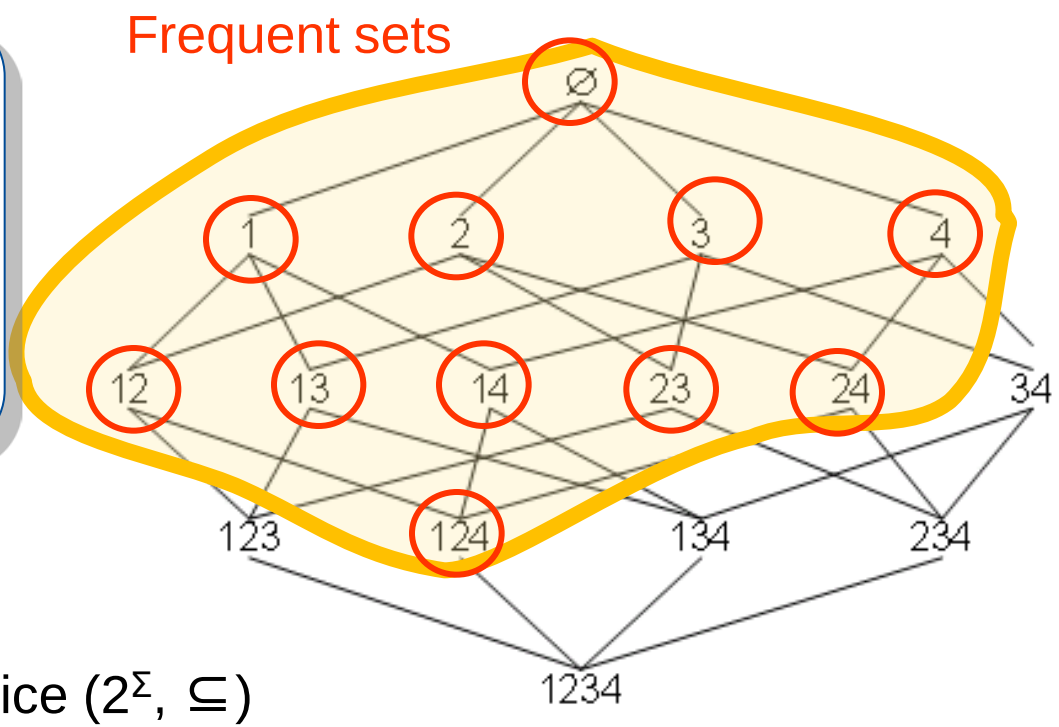
17

Definitions: Frequent sets

- The occurrence of X in a database ***T***:
  $Occ(X, T) = \{ t \in T : X \subseteq t \}$
- X is σ-frequent (frequent) in T if $Fr(X, T) = | Occ(X, T) | \geqq \sigma$.

minsupσ= **2**

Frequent sets
∅,
1, 2, 3, 4,
12, 13, 14,
23, 24, 124



	1	2	3	4	5
t1	○		○		
t2		○		○	
t3	○	○	○	○	
t4		○	○		○
t5	○	○		○	

database

The itemset lattice $(2^{\Sigma}, \subseteq)$

Definitions: Problem

Frequent Itemset Mining Problem

- Given: A transaction database $\boldsymbol{T}$ and a non-negative integer $0 \leqq \sigma \leqq |\boldsymbol{T}|$
- Task: Enumerate **all "frequent" itemsets X** in $\boldsymbol{T}$ that have frequency at least σ ($Fr(\mathbf{X}) \geqq \sigma$)

- $\mathscr{F}$: the class of all σ-frequent itemsets
- The number $|\mathscr{F}|$ of solutions is exponential in the number $\boldsymbol{n}$ of items.
- a typical enumeration problem.

頻出アイテム集合マイニング

Frequent Itemset Mining Problem

- Given: A database DB over a set Σ of items, and a number $\sigma \geqq 0$ called “minsup” (minimum support)
- Problem: Find all itemsets $X \subseteq \Sigma$ appearing in no more than σ records of DB

Applications to more sophisticated data mining problems

- Association rule [Agrawal, Srikant ‘94]
  - {Mustard, Sausage, Beer => PotateChips } **with frequency** 40%
- Optimized classification rule [Sese & Morishita PODS’90]
  - **If** gene0001 **&** gene0012 **then** diabetes **with classification error** 8.5%
- Itemset boosting/SVM [Saigo, Uno, Tsuda Bioinformatics **23(18)** 2007]
  - Learning a linear classifier over itemsets as composite features
- Weighted substructure mining [Nowozin, Tsuda, Uno, Kudo, Bakir, CVPR’07]
  - Application to image processing

機械学習アルゴリズムへの応用

ブースティング (Boosting) [Freund, Shapire 1996]

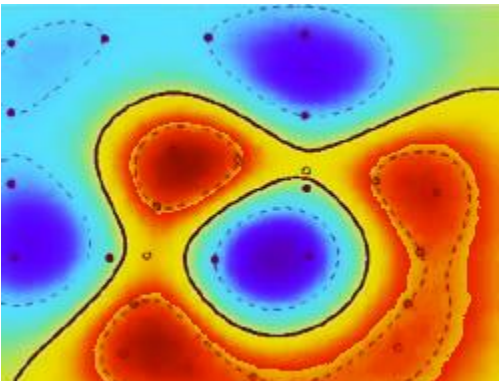
- 多数の機械学習アルゴリズムを統合して高精度予測
- オンライン予測の理論と深い関連

SVM (Support Vector Machines) [Vapnik 1996]

- マージン最大化による現在の state-of-the-art methods
- カーネル法を用いた高次元空間と多様なデータへの拡張

これらの学習アルゴリズムと，重みつきアイテム集合マイニング
を組み合わせて利用できる

キーワード: Boosting, SVM, オンライン予測, ニューラルネット, 計算学習理論
国際会議: NIPS, ICML, COLT, ALT



- V. Vapnik, Statistical Learning Theory, Wiley, 1998. (textbook)
- N. Cristianini and J. Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, Cambridge, 2000. (textbook)
- Y. Freund and R. E. Schapire, A decisiontheoretic generalization of on-line learning and an application to boosting, JCSS, 55, 119-139, 1997. (AdaBoost)
- 金森, 畑埜, 渡辺,ブースティング: 学習アルゴリズムの設計技法, 森北出版 (text book)

21

Computational Complexity of Data Mining Algorithms

How to model efficient data mining algorithms?

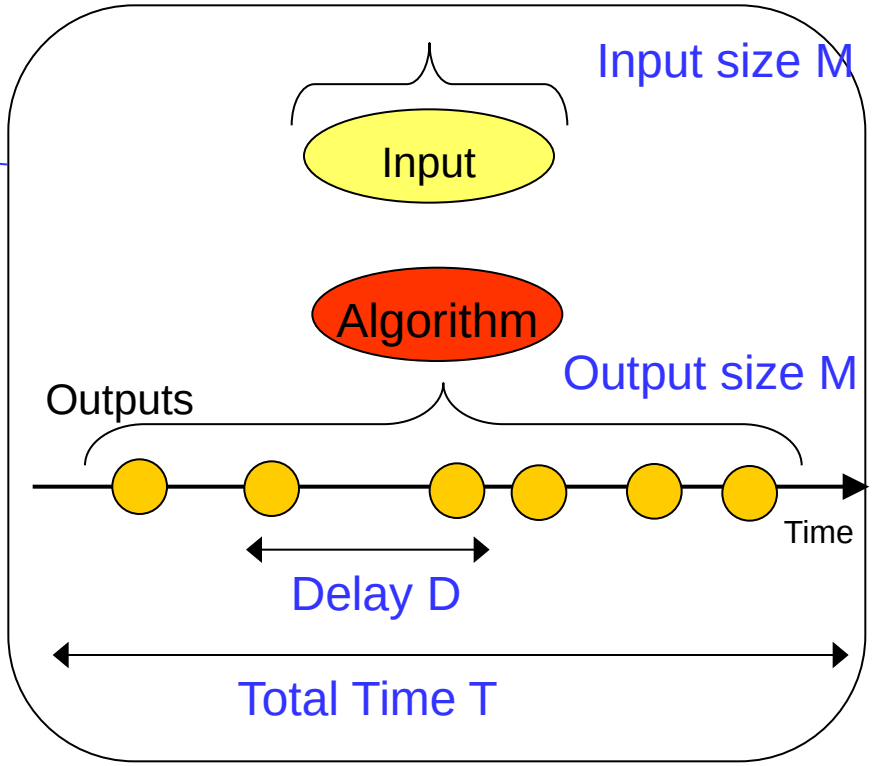
- **Light-weight**
- **High-throughput**

Computational Complexity of Data Mining Algorithms

Modeling data mining as enumeration

- **Idea: Mesure the computation time per solution**

- **Output-polynomial (OUT-POLY)**
  - Total time = poly(N, M)
- **polynomial-time enumeration, or amortized polynomial-delay (POLY-ENUM)**
  - Amotized delay is poly(Input), or
  - Total time = M·poly(N)
- **polynomial-delay (POLY-DELAY)**
  - Maximum of delay is poly(Input)



\+ **polynomial-space (POLY-SPACE)**

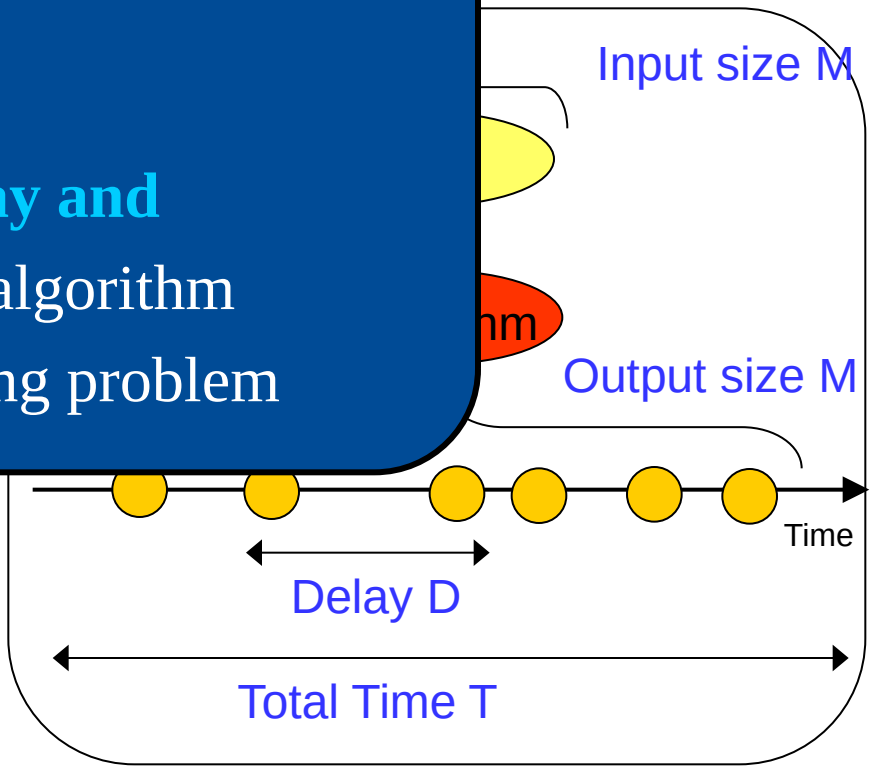
Modeling data mining as enumeration

- **Idea: Mesure the computation time per solution**

Ultimate Goal:

To design
polynomial delay and
polynomial space algorithm
for a given data mining problem

- **Output-p**
  - Total t
- **polynom**
  (POLY-E
  - Amotized
  - Total time i utput·poly(Input)



- **polynomial-delay (POLY-DELAY)**
  - Maximum of delay is poly(Input)

+ polynomial-space (POLY-SPACE)

Frequent Itemset Mining Algorithms

Apriori Algorithm (BFS algorithm)

目次

1回: データマイニングと頻出集合発見

- データマイニング
- 頻出集合マイニング

ポイント

- データマイニングとは何か？
- 列挙アルゴリズム
- 高速なアルゴリズムの設計