

クラス間共通パターンとその効率的発見アルゴリズム

北澤 聖也[†] 棚原 亮[†] 亀谷 由隆[†]

[†]名城大学理工学部情報工学科

1 はじめに

クラスラベル付きのトランザクション集合が与えられたとき、興味あるクラスに関連のあるパターンを発見するタスクとして識別パターン発見[1]がある。例えば、生物分類データが与えられたとき、哺乳類は脊椎があり、母乳を飲むといったパターンを見つけることができる。しかし、脊椎があるというパターンは脊椎動物全てに共通することである。そのため、本研究ではデータベースの中の様々なクラスの組み合わせを考え、それらに共通して出現するパターンを効率よく見つける手法を提案する。

2 提案手法

2.1 クラス間共通パターン

パターン発見の対象とするデータセットはクラスラベル付きのトランザクション集合 $D = \{\tau_1, \tau_2, \dots, \tau_N\}$ である。トランザクション数は $|D| = N$ である。 D 中に出現するクラスラベル集合を C 、アイテム集合を X とおく。 c はクラスラベルであり、 $c \in C$ が成り立つ。また、 t はアイテム集合であり、 $t \subseteq X$ が成り立つ。

今回探したいパターンを $\pi = \langle c, x \rangle$ とする。 c とはクラスラベル集合であり、 $c \subseteq C$ が成り立つ。 x とはアイテム集合であり、 $x \subseteq X$ が成り立つ。 $\pi = \langle c, x \rangle$ はクラス集合 c にアイテム集合 x が共通して現れることを意味しており、クラス間共通パターン（あるいは単に共通パターン）と呼ぶ。また、 c と x に対し、 c のいずれかに所属するトランザクションの数を $N(c)$ 、 x を含むトランザクションの数を $N(x)$ 、さらに、 c のいずれかに所属し x を含むトランザクションの数を $N(c, x) = \sum_{c \in C} N(c, x)$ と定義し、 $N(c, x)$ をクラス c において、パターン x に適合したトランザクション数と定義する。このとき、再現率 $p(x | c) = N(c, x) / N(c)$ 、適合率 $p(c | x) = N(c, x) / N(x)$ となる。本研究では共通パターンの評価値としてこれらの調和平均である、F 値 ($F(\pi) = F(c, x) = 2p(c | x)p(x | c) / (p(c | x) + p(x | c))$) を用いる。そして、F 値の上位 k 個以下のパターンを求めるこことを考える。

2.2 探索木とその性質

素朴に識別パターン発見アルゴリズムを拡張しようとすれば、 $c \subseteq C$ となるような c を正クラス、 c ではないクラス集合を負クラスとする。そして、正クラスを興味のあるクラスとして識別パターンを見つける。例えば、FP-Growth を基にした場合、この方法では必要以上に FP-tree を構築することになり無駄が多い。

そこで提案手法では、各アイテム集合 x を深さ優先で探索しながら、各 x に対し全てのクラスの組み合わせ c を深さ優

先探索することを考える。図 1 に示すように x については接尾探索木[1]を考え、 c については接頭探索木[1]を考える。なお、この探索木は $X = \{A, B, C\}$, $C = \{c_1, c_2, c_3, c_4\}$ に対する木であり、探索木のクラスは $c_i = i$ で表示している。また図 1 の右方にある破線で囲まれた三角形は $\{A\}$ の下にある接頭探索木と同じものを省略したものである。

各アイテム集合 x を探索するときは、各ノードに各クラス c のトランザクション数 $N(c, x)$ を記録した FP-tree を構築しながら探索を行う。そして、記録された $N(c, x)$ に基づき、確率計算のみで評価値を求める。各クラス集合 c について探索するときは、注目した x における再現率の降順でクラスを探索しパターン $\langle c, x \rangle$ を生成する。例えば図 1 では、 $p(x | c_1) \geq \dots \geq p(x | c_4)$ が成り立っている。ここで、全ての $c' \in c$ について $p(x | c) \leq p(x | c')$ が成り立つ $c \notin c'$ となる c に対し、

$$\begin{aligned} p(x | c \cup \{c\}) &= \frac{p(c)p(x | c) + \sum_{c' \in c} p(c')p(x | c')}{p(c) + \sum_{c' \in c} p(c')} \\ &\leq p(x | c) \end{aligned} \quad (1)$$

であることは容易に示せるため、クラス集合の拡大については逆単調性が成り立つと言える。

また、クラス集合 c を c によって拡大したときの適合率は、単に元の適合率の和となる。

$$p(c \cup \{c\} | x) = p(c | x) + p(c | x) \quad (2)$$

クラス集合の接頭探索木において、現在注目するクラス集合 c を拡大する際に、追加することのないクラス集合を c_{out} とおく。例えば現在 $\{c_2\}$ について探索している場合、 $c_{out} = \{c_1\}$ となる。この c_{out} は枝刈りの際に用いる。

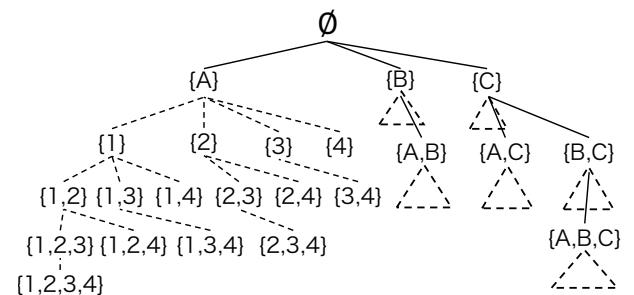


図 1: 提案方法の探索木

2.3 枝刈り

本手法では、分枝限定法に基づいて評価値の高いパターンから最大 k 個を出力するため、大きさ k の候補リストを用意し、評価値の降順で格納していく。まず、候補リストの k 番目のパターンを π_k とし、現在探索中のパターンを $\pi = \langle c, x \rangle$ と置き、クラスの組み合わせについて探索することを考える。探索が深くなるにつれてクラス集合が拡大するため、式 2 より適合率は 1 に近づく。しかし、2.2 節で導入した c_{out} が空でない場合、

Efficient Mining of Inter-Class Common Patterns from Labeled Transactions

[†] Seiya Kitazawa

[†] Ryo Tochihara

[†] Yoshitaka Kameya

Department of Information Engineering, Faculty of Science and Technology, Meijo University (†)

$p(\mathbf{c}_{out}, \mathbf{x})$ の分が適合率に加わることはない。つまり、 \mathbf{c} の拡大において適合率の最大値は $1 - p(\mathbf{c}_{out} | \mathbf{x})$ となる。クラス拡大時の逆単調性（式 1）も合わせて考えると、このパターン $\pi = \langle \mathbf{c}, \mathbf{x} \rangle$ のクラス拡大における上界は $\bar{F}(\pi) = 2(1 - p(\mathbf{c}_{out} | \mathbf{x}))p(\mathbf{x} | \mathbf{c}) / (1 - p(\mathbf{c}_{out} | \mathbf{x}) + p(\mathbf{x} | \mathbf{c}))$ で求められる。そのため、クラスの組み合わせを探索中 $\bar{F}(\pi) < F(\pi_k)$ であれば、枝刈りを行う。

一方、アイテム集合 \mathbf{x} の拡大においては、最大の再現率をもつクラス $c_{max} = \text{argmax}_{c \in C} P(\mathbf{x} | c)$ を考える。このとき、 $\bar{F}(c_{max}, \mathbf{x})$ は現在のアイテム集合 \mathbf{x} の拡大における上界であるため、この値が $F(\pi_k)$ 未満なら \mathbf{x} 以下を枝刈りできる。

2.4 冗長パターンの削除

異なる 2 つの候補パターン $\pi_1 = \langle \mathbf{c}_1, \mathbf{x}_1 \rangle, \pi_2 = \langle \mathbf{c}_2, \mathbf{x}_2 \rangle$ において、 $\mathbf{c}_1 \supseteq \mathbf{c}_2$ かつ $\mathbf{x}_1 \subseteq \mathbf{x}_2$ のとき、 π_1 は π_2 よりも一般的であると言い、 π_2 の方が特殊であると言う。冗長なパターンの削除は、生産性制約 [1] と飽和制約 [2] によって行う。生産性制約では π_1 が π_2 より一般的かつ $F(\pi_1) \geq F(\pi_2)$ のとき、 π_1 を残し π_2 を候補リストから削除する。飽和制約では π_2 が π_1 より特殊かつ $N(\pi_1) = N(\pi_2)$ のとき、 π_2 を残し、 π_1 を候補リストから削除する。提案手法では、探索中に生産性制約でチェックを行い、出力の直前で飽和制約でのチェックを行う。ただし、この 2 つの制約が競合した場合は飽和制約を優先させる。

3 実験

Zoo データセット¹を用いて提案手法の動作実験を行った。このデータは 101 個のクラスラベル付きのトランザクションから構成されている。個々のトランザクションが一つの生物種に対応しており、動物の特徴（母乳を飲む、卵を産む、など）がアイテムとして表現されている。また各生物種は 1~7 のクラス（1: 哺乳類、2: 鳥類、3: 爬虫類、4: 魚類、5: 両生類、6: 昆虫類、7: 甲殻類）に分類されている。

提案手法によって得られた共通パターンの一部を表 1 に示す。この結果は、出力パターン数 $k = 50$ で実行して得られた、上位 7 つの共通パターンである。例えば、1 番目のパターンは 哺乳類・鳥類・爬虫類・魚類・両生類に共通する特徴が「脊椎をもつ」ことを意味する。さらに、7 番目に F 値が 1 ではないパターンが出現しているが、これはクラス間で大まかに共通している特徴を柔軟に出力していることを示している。

表 1: Zoo データから得られる共通パターン

$F(\mathbf{c}, \mathbf{x})$	パターン $\langle \mathbf{c}, \mathbf{x} \rangle$
1.000	$\langle \{1, 2, 3, 4, 5\}, \{\text{backbone}\} \rangle$
1.000	$\langle \{1\}, \{\text{backbone, breath, milk}\} \rangle$
1.000	$\langle \{2\}, \{\text{backbone, breath, eggs, feathers, 2legs, tail}\} \rangle$
1.000	$\langle \{4, 5\}, \{\text{aquatic, backbone, eggs, toothed}\} \rangle$
1.000	$\langle \{4\}, \{\text{aquatic, backbone, eggs, fins, tail, toothed}\} \rangle$
1.000	$\langle \{5\}, \{\text{aquatic, backbone, breath, eggs, toothed}\} \rangle$
0.984	$\langle \{1, 3, 4, 5\}, \{\text{toothed}\} \rangle$

また、20News データセット² ([1] で用いた前処理済みのものを使用) を用いて提案手法の動作実験を行った。このデータはニュースグループのカテゴリをクラスとして分けられて

1 <https://archive.ics.uci.edu/ml/datasets/Zoo>

2 <http://qwone.com/~jason/20Newsgroups/>

おり、アイテムは投稿記事中の単語である。このデータセットでは、`articl` という中立的なアイテムがどのカテゴリにも多く含まれていたため、データセットから `articl` はストップワードとして削除した。得られた共通パターンの一部を表 2 に示す。この結果は、出力パターン数 $k = 20$ で実行して得られた、上位 7 つのパターンである。クラスの番号はそれぞれ、1: `rec.motorcycles`, 2: `comp.os.ms-windows.misc`, 3: `comp.windows.x`, 4: `sci.crypt`, 5: `rec.sport.baseball`, 6: `rec.sport.hockey`, 7: `alt.atheism`, 8: `soc.religion.christian`, 9: `talk.religion.misc` を表している。3 番目のパターンは `comp.os.ms-windows.misc` と `comp.windows.x` というカテゴリで window(s) に関する記述が多かったことを意味する³。

表 2: 20News データから得られる共通パターン

$F(\mathbf{c}, \mathbf{x})$	パターン $\langle \mathbf{c}, \mathbf{x} \rangle$
0.661	$\langle \{1\}, \{\text{dod}\} \rangle$
0.621	$\langle \{1\}, \{\text{bike}\} \rangle$
0.586	$\langle \{2, 3\}, \{\text{window}\} \rangle$
0.575	$\langle \{4\}, \{\text{encrypt}\} \rangle$
0.571	$\langle \{5, 6\}, \{\text{game}\} \rangle$
0.547	$\langle \{7, 8, 9\}, \{\text{god}\} \rangle$
0.539	$\langle \{8\}, \{\text{rutger}\} \rangle$

2.2 節で示した素朴な探索方法で Zoo データの共通パターンを発見したときとの探索性能の比較を表 3 (上) に示し、20News データでの探索性能の比較を表 3 (下) に示す。なお、出力結果は提案手法と変わらなかった。この結果より、提案手法は従来の方法より効率的であることが言える。

表 3: Zoo (上) および 20News (下) での性能比較

Zoo データ	素朴な探索	提案手法	比
探索回数 [回]	1,715	798	46.5%
探索時間 [ms]	66	26	39.4%

20News データ	素朴な探索	提案手法	比
探索回数 [回]	1,053,743	134,760	12.8%
探索時間 [ms]	78,966,255	79,798	0.1%

4 おわりに

本研究ではクラス間共通パターンとその効率的発見アルゴリズムを提案した。また、従来手法との比較を行い、探索性能が向上していることを確認した。今後の課題として、より大規模なデータについても同様の探索性能になるかの確認、F 値以外の評価値を使う場合の検討、クラスタ分析手法との組み合わせなどが挙げられる。

参考文献

- [1] 亀谷由隆、佐藤泰介: 最小サポート上昇法に基づく上位 k 関連パターン発見、人工知能学会データ指向構成マイニングとシミュレーション研究会予稿集, 2011.
 - [2] 宇野毅明、有村博紀: データインテンシブコンピューティング: その 2 頻出アイテム集合発見アルゴリズム、人工知能学会誌, Vol. 22, No. 3, pp. 425-436, 2007.
- 3 1 番目のパターンは `rec.motorcycles` においてオンライン同好会 DoD の記述が多かったことを意味する。