D-008

データベースのスキーマ情報を活用した機械学習
Machine Learning Using Database Schemas

志村 薫
Kaoru Shimura

杉浦 健人
Kento Sugiura

石川 佳治
Yoshiharu Ishikawa

1 はじめに

ビッグデータ時代の今日，企業などのデータベースでは大量のデータが蓄積，管理されている．これら大量のデータから知識を発見しビジネスなどに活用するために，データベース中のデータを用いた機械学習への需要が高まっている．

しかし，データベース中のデータを用いた機械学習ではデータベース特有のスキーマ情報を機械学習に活かせていない．データベース管理システム（database management system: DBMS）はデータ本体のみならず，データ間の制約や従属性などさまざまなスキーマ情報も管理している．しかし，現状の機械学習ではこれらを活用しきれておらず，一部のテーブルに対する結合の省略 [1] のような限定的な応用に留まっている．

一般にデータベース中のデータは正規化もしくはデータウェアハウスに基づく解析処理のため，表 1 のように複数のテーブルに分割されており，学習用データへの変換には高コストなテーブルの結合が必要となる．従来ではすべてのテーブルを結合していたが，Kumar らは外部キー制約を活用することで一部のテーブルの結合を省略する手法を提案した [1]．この手法ではスタースキーマに従ったテーブル群を前提としており，ディメンションテーブルに対するファクトテーブルのタプル比を用いて，結合を省略した場合の学習モデルの汎化性能の低下具合を見積もる．タプル比がしきい値以上の場合は結合の省略が可能であると判断し，ファクトテーブルの外部キーを対応するディメンションテーブルの全属性の代わりに機械学習の特徴量として，結合を省略する．

例えば，表 1 の月・商品・店舗ごとの販売個数のデータセットを用いて，今後の販売個数を回帰分析により予測することを考える．従来では表 1 (a) の販売テーブル，表 1 (b) の商品テーブル，表 1 (c) の店舗テーブルのすべてを結合し，（販売時期，カテゴリ，単価，規模，県，市，区）を特徴量として用いた．一方，Kumar らの手法では仮にしきい値を 1000 とすると，商品テーブルのタプル比は $\frac{6000000}{10000} = 600 < 1000$，店舗テーブルのタプル比は $\frac{6000000}{100} = 60000 > 1000$ であるため，（販売時期，カテゴリ，単価，店舗 ID）を特徴量とし，店舗テーブルの結合を省略する．

このように，Kumar らが提案した外部キー制約を用いた結合省略手法はスキーマ情報を活用した機械学習の好例である一方，課題も残っている．

■未活用なスキーマ・統計情報の存在　Kumar らはスキーマ情報として外部キー制約を用いたが，スキーマ情報には従属性など他の情報も存在する．また，索引やヒストグラム，各属性値の分布などの情報を用いた学習データの選定も考えられる．

■テーブル単位の結合の省略　外部キー制約を用いる場合，結合の省略はテーブル単位で行われる．しかし，例えばテーブル中のある属性は学習に有効だが別の属性は有効でないというように，属性単位で結合を行うか決定する方が望ましい場合も考えられる．

これら課題を踏まえ，本稿ではデータベースのスキーマ情報を最大限に活用した機械学習フレームワークについて検討し，スキーマ情報に基いた学習データの選択・加工により学習の効率化を目指す．

表 1: 月・商品・店舗ごとの販売個数のデータセット

(a) 販売テーブル（ファクトテーブル）

販売 ID	販売時期	販売個数	商品 ID	店舗 ID
#1	2019/04	300	#1	#1
#2	2019/04	200	#1	#2
#3	2019/04	100	#1	#3
-	-	-	-	-
#6000000	2019/09	250	#10000	#100

(b) 商品テーブル（ディメンションテーブル）

商品 ID	カテゴリ	単価（円）
#1	食品	500
#2	文具	100
#3	衣類	1000
-	-	-
#10000	食品	700

(c) 店舗テーブル（ディメンションテーブル）

店舗 ID	規模	県	市	区
#1	大	愛知	名古屋	千種
#2	中	愛知	名古屋	緑
#3	小	愛知	名古屋	緑
-	-	-	-	-
#100	中	神奈川	横浜	緑

名古屋大学大学院情報学研究科　Graduate School of Informatics, Nagoya University

2 要素技術

本章では提案手法における要素技術として，確率的関数従属性と列指向データベースについて述べる．

2.1 確率的関数従属性

スキーマ情報を活用し切れていないという課題に対し，関数従属性の活用について検討する．関数従属性は，ある属性 X の値が決まると別の属性 Y の値が一意に定まるという性質であり，$X \rightarrow Y$ と表記する．

特に，本研究では関数従属性を確率的に拡張した確率的関数従属性 [2–6] に着目する．関数従属性は属性間の関係を表す有用な情報であるが，一方でデータベース中の全タプルが従属性の条件を満たさなければならないという厳しい条件を持ち，一部の例外的なタプルによって関数従属性が成り立たないケースも起こりうる．そこで，確率的関数従属性では従属性が成り立つ割合を確率で表す．例えば，表 1 (c) では ID 100 以外のタプルにおいて関数従属性 ⟨ 区名 → 市名 ⟩ が成り立っているとする．この場合，関数従属性を ⟨ 区名 → 市名 : 0.99 ⟩ のように確率的に表すことで，従属性の成り立つ割合をシステムが認識可能になる．

2.2 列指向データベース：

より柔軟な結合の省略を実現するために，列指向データベースの利用を検討する．列指向データベースは列単位でデータが格納されたデータベースであり，代表的な実装としては C-Store [7] や MonetDB [8] が挙げられる．列指向データベースでは一般的な行指向データベースと異なり列単位でデータが格納されているため，特に列単位の集約処理などが効率的に行える．一方で，行単位の処理が必要なデータ更新などの OLTP（online transaction processing）処理の効率は悪い．

本研究では列指向データベースを用いることで，列単位の結合の省略を考える．上述したように列指向データベースには OLTP 処理の効率が悪いという問題もあるが，本研究ではデータウェアハウスのようにデータ更新がほとんど起きないような状況のデータに対する機械学習を想定し，列単位の結合や効率的な集約処理などの利点を主に活用する．

3 活用方針

本章では関連研究や要素技術を踏まえ，機械学習におけるスキーマ情報の活用方針案について述べる．

3.1 カテゴリデータから量的データへの近似的変換

統計的に扱いやすくするため，カテゴリデータは量的データへと変換されることがあり，代表的な変換手法としては交互最小二乗法（alternating least squares method: ALS）などが存在するが，収束するまで繰り返しパラメータを推定するため，計算コストが大きいという問題がある．

そこで，DBMS が保持する従属性などのスキーマ情報や各属性値の分布などの統計情報を基に，カテゴリデータから量的データへの変換を近似的に行うことで，計算コストを抑えるといった活用方針が考えられる．

3.2 特徴選択の近似的実行

学習の効率化や予測モデルの解釈性向上のため，特徴量全体の中から意味のある一部の特徴量のみを選択し，学習に用いる特徴選択という手法がある．特徴選択は 2 種類に大別され，特徴量の部分集合を用いて実際に学習，評価を行い最適な特徴量を選択するラッパー法と目的変数と各特徴量との情報利得や Gini 係数などの統計的指標を用いて選択するフィルタ法が存在する．

特に，フィルタ法については統計的指標を用いるという性質上，DBMS と相性が良いと考えられる．DBMS が保持する従属性などのスキーマ情報や各属性値の分布などの統計情報を特徴選択の指標として用いることで，特徴選択を近似的に実行するといった活用方針が考えられる．

4 おわりに

本稿ではデータベースのスキーマ情報を活用した機械学習に着目した．関連研究として外部キー制約を活用した結合省略手法について，概要を述べ，課題として未活用なスキーマ・統計情報の存在，テーブル単位の結合の省略を挙げた．また，提案手法における要素技術として確率的関数従属性，列指向データベースについて考察した．そして，これらを踏まえた上でスキーマ情報の活用方針案として，カテゴリデータから量的データへの近似的変換，特徴選択の近似的実行について述べた．今後はこれらの方針に従い機械学習フレームワークを実装し，実験を行うことでスキーマ情報の活用による学習の効率化具合を計測したい．

謝辞

本研究の一部は，科研費 16H01722 および 19K21530 による．

参考文献

[1] A. Kumar, J. Naughton, J. M. Patel, and X. Zhu, “To join or not to join?: Thinking twice about joins before feature selection,” in *Proc. SIGMOD*, pp. 19–34, 2016.
[2] Z. Abedjan, L. Golab, and F. Naumann, “Profiling relational data: A survey,” *VLDB J.*, vol. 24, no. 4, pp. 557–581, 2015.
[3] J. Liu, J. Li, C. Liu, and Y. Chen, “Discover dependencies from data — a review,” *IEEE TKDE*, vol. 24, no. 2, pp. 251–264, 2010.
[4] S. Ma, L. Duan, W. Fan, C. Hu, and W. Chen, “Extending conditional dependencies with built-in predicates,” *IEEE TKDE*, vol. 27, no. 12, pp. 3274–3288, 2015.
[5] W. Fan, F. Geerts, J. Li, and M. Xiong, “Discovering conditional functional dependencies,” *IEEE TKDE*, vol. 23, no. 5, pp. 683–698, 2010.
[6] G. Cormode, L. Golab, K. Flip, A. McGregor, D. Srivastava, and X. Zhang, “Estimating the confidence of conditional functional dependencies,” in *Proc. SIGMOD*, pp. 469–482, 2009.
[7] M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. Madden, E. O’Neil, P. O’Neil, A. Rasin, N. Tran, and S. Zdonik, “C-store: A column-oriented DBMS,” in *Proc. VLDB*, pp. 553–564, 2005.
[8] MonetDB: Home: `https://www.monetdb.org/` (accessed: June 17, 2019).