TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU VÀ PHƯƠNG PHÁP KHAI PHÁ LUẬT KẾT HỢP TRONG CƠ SỞ DỮ LIỆU

● NGUYỄN THỊ VIỆT HÀ

TÓM TẮT:

Ngày nay, sự phát triển vượt bậc việc ứng dụng công nghệ thông tin ở hầu hết các lĩnh vực khiến lượng dữ liệu được lưu trữ ngày càng lớn. Cách khai thác dữ liệu truyền thống không còn đáp ứng đầy đủ những yêu cầu, những thách thức mới. Do đó, kỹ thuật phát hiện tri thức trong cơ sở dữ liệu ra đời nhằm giải quyết bài toán khai phá dữ liệu.

Từ khóa: Dữ liệu, khai phá, cơ sở, công nghệ thông tin, lưu trữ, chuyển đổi.

1. Đặt vấn đề

Thuật ngữ Khai phá dữ liệu (KPDL) ra đời vào cuối những năm 80 thế kỷ trước. Có nhiều định nghĩa khác nhau về KPDL, nhưng diễn đạt một cách dễ hiểu thì KPDL là quá trình tìm kiếm những thông tin (tri thức) có ích, tiềm ẩn và mang tính dự đoán trong các khối cơ sở dữ liệu (CSDL) lớn.

Mục đích việc phát hiện tri thức từ dữ liệu KPDL là cốt lõi của quá trình khám phá tri thức gồm có các giải thuật KPDL chuyên dùng, dưới một số quy định về hiệu quả tính toán chấp nhận được. KPDL nhằm tìm ra những mẫu mới, những thông tin tiềm ẩn mang tính dự đoán chưa được biết đến, có khả năng mang lại lợi ích cho người sử dụng và KPDL là tìm ra các mẫu được quan tâm nhất tồn tại trong CSDL, nhưng chúng lại bị che giấu bởi một số lượng lớn dữ liệu.

Ngày nay, công nghệ thông tin phát triển đồng nghĩa với việc phát triển các phần mềm ứng dụng. Phần mềm KPDL là một công cụ phân tích dùng để phân tích dữ liệu. Nó cho phép người sử dụng phân tích dữ liệu theo nhiều góc nhìn khác nhau, phân loại dữ liệu theo những quan điểm riêng biệt và tổng kết các mối quan hệ đã được bóc tách.

Hiện kỹ thuật KPDL đang được áp dụng một cách rộng rãi trong nhiều lĩnh vực kinh doanh và đời sống khác nhau, như: Thương mại (phân tích dữ liệu bán hàng và thị trường, phân tích đầu tư, quyết định cho vay, phát hiện gian lận,...); Thông tin sản xuất (điều khiển và lập kế hoạch, hệ thống quản lý, phân tích kết quả thử nghiệm,...); Thông tin khoa học (dự báo thời tiết,...); CSDL sinh học (ngân hàng gen,...); Khoa học địa lý (dự báo động đất,...); Trong y tế, marketing, ngân hàng, viễn thông, du lịch, internet...

Những gì thu được từ KPDL thật đáng giá. Điều đó được chứng minh bằng thực tế: Chẩn đoán bệnh trong y tế dựa trên kết quả xét nghiệm đã giúp cho bảo hiểm y tế phát hiện ra nhiều trường hợp xét nghiệm không hợp lý; Trong dịch vụ viễn thông đã phát hiện ra những nhóm người thường xuyên gọi cho nhau bằng mobile và thu lợi hàng triệu USD; IBM Suft-Aid đã áp dụng KPDL vào phân tích các lần đăng nhập Web vào các trang liên

quan đến thị trường để phát hiện sở thích khách hàng, từ đó đánh giá hiệu quả của việc tiếp thị qua Web và cải thiện hoạt động của các Website; trang Web mua bán qua mạng cũng tăng doanh thu nhờ áp dụng KPDL trong việc phân tích sở thích mua bán của khách hàng.

2. Các bước khám phá tri thức và phương pháp chính trong khai phá dữ liệu

2.1. Các bước khám phá tri thức

Bước 1- Trích chọn dữ liệu (data selection): Là bước trích chọn những tập dữ liệu cần được khai phá từ các tập dữ liệu lớn (databases,data warehouses).

Bước 2 - Tiền xử lý dữ liệu (data preprocessing): Là bước làm sạch dữ liệu (xử lý dữ liệu không đầy đủ, dữ liệu nhiễu, dữ liệu không nhất quán,...), rút gọn dữ liệu (sử dụng các phương pháp thu gọn dữ liệu, histograms, lấy mẫu,...), rời rạc hoá dữ liệu (dựa vào histograms, entropy, phân khoảng,..). Sau bước này, dữ liệu sẽ nhất quán, đầy đủ, được rút gọn và được rời rạc hoá.

Bước 3- Biến đổi dữ liệu (data transformation): Là bước chuẩn hoá và làm mịn dữ liệu để đưa dữ liệu về dạng thuận lợi nhất nhằm phục vụ cho các kỹ thuật khai phá ở bước sau.

Bước 4- Khai phá dữ liệu (data mining): Đây là bước quan trọng và tốn nhiều thời gian nhất của quá trình khám phá tri thức, áp dụng các kỹ thuật khai phá (phần lớn là các kỹ thuật của machine learning) để khai phá, trích chọn được các mẫu (pattern) thông tin, các mối liên hệ đặc biệt trong dữ liệu.

Bước 5- Đánh giá và biểu diễn tri thức (knowledge representation & evaluation): Dùng các kỹ thuật hiển thị dữ liệu để trình bày các mẫu thông tin (tri thức) và mối liên hệ đặc biệt trong dữ liệu đã được khai phá ở bước trên biểu diễn theo dạng gần gũi với người sử dụng như đồ thị, cây, bảng biểu, luật,... Đồng thời, bước này cũng đánh giá những tri thức khai phá được theo những tiêu chí nhất định.

Trong giai đoạn KPDL, có thể cần sự tương tác của người dùng để điều chỉnh và rút ra các tri thức cần thiết nhất. Các tri thức nhận được cũng có thể được lưu và sử dụng lại.

2.2. Các phương pháp chính trong khai phá dữ liệu

2.2.1. Phương pháp luật kết hợp

Một trong những chủ đề phổ biến của KPDL là khai phá luật kết hợp. Mục đích của khai phá luật kết hợp là xác định mối quan hệ, sự kết hợp giữa các mục dữ liệu (item) trong một CSDL lớn.

2.2.2. Phương pháp cây quyết định

Mô tả tri thức dạng đơn giản nhằm phân các đối tượng dữ liệu thành một số lớp nhất định. Các nút của cây được gán nhãn là tên các mục dữ liệu, các cạnh được gán các giá trị có thể của các mục dữ liệu, các lá mô tả các lớp khác nhau. Các đối tượng được phân lớp theo các đường đi trên cây, qua các cạnh tương ứng với các giá trị của mục dữ liệu tới lá.

2.2.3. Phương pháp K-Mean

Có nhiều phương pháp được sử dụng trong phân cụm, phương pháp k-Mean được coi là các kỹ thuật cơ bản của phân cụm. Với phương pháp này sẽ chia tập có n đối tượng thành k cụm sao cho các đối tượng trong cùng một cụm thì giống nhau, các đối tượng khác cụm thì khác nhau.

2.2.4. Các phương pháp dựa trên mẫu

Phương pháp này sử dụng khai phá chuỗi theo thời gian (Sequential temporal patterns). Xét về mặt kỹ thuật thì tương tự như KPDL bằng luật kết hợp nhưng có thêm tính thứ tự và tính thời gian. Một luật mô tả mẫu tuần tự có dạng tiêu biểu X -> Y phản ánh sự xuất hiện của biến cố X sẽ dẫn đến việc xuất hiện kế tiếp biến cố Y. Hướng tiếp cận này được ứng dụng nhiều trong lĩnh vực tài chính và thị trường chứng khoán bởi chúng có tính dự báo cao.

3. Khai phá luật kết hợp trong cơ sở dữ liệu

3.1. Khái niệm

Khai phá luật kết hợp (KPLKH) là một kỹ thuật quan trọng của KPDL. Mục tiêu nhằm phát hiện mối quan hệ giữa các mục dữ liệu trong CSDL. Mô hình đầu tiên của bài toán KPLKH là mô hình nhị phân (hay còn gọi là mô hình cơ bản) được R. Agrawal, T. Imielinski và A. Swami đề xuất vào năm 1993, xuất phát từ nhu cầu phân tích dữ liệu của cơ sở dữ liệu giao tác, phát hiện các mối quan hệ giữa các tập mục hàng hóa (Itemsets) đã bán được tại các siêu thị. Việc xác định các quan hệ này không phân biệt vai trò khác nhau cũng như không dựa vào các đặc tính dữ liệu vốn có của các mục dữ liệu mà chỉ dựa vào sự xuất hiện cùng lúc của chúng.

3.2. Khai phá luật kết hợp

Bài toán KPLKH có thể phát biểu như sau: Cho

cơ sở dữ liệu giao tác DB, ngưỡng độ hỗ trợ tối thiểu minsup và ngưỡng độ tin cậy tối thiểu minconf.

Yêu cầu: Tìm tất cả các luật kết hợp $X \rightarrow Y$ trên cơ sở dữ liệu DB sao cho $sup(X \rightarrow Y) \geq$ minsup và $conf(X \rightarrow Y) \geq$ minconf.

KPLKH này được gọi là bài toán cơ bản hay bài toán nhị phân, vì ở đây, giá trị của mục dữ liệu trong cơ sở dữ liệu là 0 hoặc 1 (xuất hiện hay không xuất hiện).

Bài toán KPLKH trong CSDL chia thành hai bài toán con:

(1) Tìm tất cả các tập mục thường xuyên: Một tập mục là thường xuyên được xác định qua tính độ hỗ trợ và thoả mãn độ hỗ trợ cực tiểu.

(2) Sinh ra các luật kết hợp từ các tập mục thường xuyên đã tìm được thỏa mãn độ tin cậy tối thiểu cho trước.

Khi KPLKH trong CSDL DB thì mọi khó khăn nằm ở bài toán thứ nhất là tìm tập mục thường xuyên.

3.3. Các cách tiếp cận khai phá tập mục thường xuyên

Bài toán khai phá tập mục thường xuyên có thể chia thành hai bài toán nhỏ: Tìm các tập mục ứng viên và tìm các tập mục thường xuyên. Tập mục ứng viên là tập mục mà ta hy vọng là tập mục thường xuyên, phải tính độ hỗ trợ của nó để kiểm tra. Tập mục thường xuyên là tập mục có độ hỗ trợ lớn hơn hoặc bằng ngưỡng hỗ trợ tối thiểu cho trước. Ta có thể phân chúng theo 2 tiêu chí:

- Phương pháp duyệt qua không gian tìm kiếm.
- Phương pháp xác định độ hỗ trợ của tập mục.

Phương pháp duyệt qua không gian tìm kiếm được phân làm 2 cách: duyệt theo chiều rộng (Breadth First Search - BFS) và duyệt theo chiều sâu (Depth First Search - DFS).

Duyệt theo chiều rộng là duyệt qua cơ sở dữ liệu gốc để tính độ hỗ trợ của tất cả các tập mục ứng viên có (k-1) mục trước khi tính độ hỗ trợ của các tập mục ứng viên có k mục. Với cơ sở dữ liệu có n mục dữ liệu, lần lặp thứ k phải kiểm tra độ hỗ trợ của tất cả $C_n^k = \frac{n!}{k!(n-k)!}$ tập mục ứng viên có k mục.

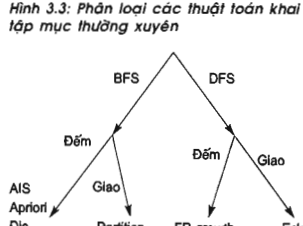
Duyệt theo chiều sâu là duyệt qua CSDL đã được chuyển đổi thành cấu trúc cây, quá trình duyệt gọi đệ quy theo chiều sâu của cây.

Với CSDL có n mục dữ liệu, không gian tìm kiếm có tất cả 2^n tập con, rõ ràng đây là bài toán NP khó, do vậy cần phải có phương pháp duyệt thích hợp, tỉa nhanh các tập ứng viên.

Phương pháp xác định độ hỗ trợ của tập mục X được chia làm hai cách: cách thứ nhất là đếm số giao tác chứa X trong CSDL. Cách thứ hai là tính phần giao các tập chứa định danh của các giao tác chứa X.

Đã có rất nhiều thuật toán tìm tập mục thường xuyên được công bố, có thể phân theo Hình 3.3 như sau:

Hình 3.3: Phân loại các thuật toán khai phá tập mục thường xuyên



3.4. Các thuật toán điển hình khai phá tập mục thường xuyên

Có hai thuật toán điển hình khai phá tập mục thường xuyên là Thuật toán Apriori và Thuật toán FP- Growth. Trong đó, Thuật toán Apriori tiêu biểu cho phương pháp sinh ra các tập mục ứng viên và kiểm tra độ hỗ trợ của chúng và Thuật toán FP-Growth, đại diện cho phương pháp không sinh ra tập mục ứng viên, cơ sở dữ liệu được nén lên cấu trúc cây, sau đó khai phá bằng cách phát triển dần các mẫu trên cây này.

3.4.1. Thuật toán Apriori

Apriori là thuật toán khai phá tập mục thường xuyên do R. Agrawal và R. Srikant đề xuất vào năm 1993. Thuật toán Apriori còn là nền tảng cho việc phát triển nhiều thuật toán khai phá tập mục thường xuyên khác về sau.

Giả sử các mục dữ liệu trong mỗi giao tác được lưu theo trật tự từ điển. Thuật toán sử dụng các ký hiệu sau: (Xem bảng).

Bảng: Thuật toán sử dụng các ký hiệu

Tập k mục:	Chức năng
L_k	Tập các k-tập mục thường xuyên (với độ hỗ trợ tối thiểu minsup). Mỗi phần tử của tập này có 2 trường:
	i) Tập mục (itemsets)
	ii) Độ hỗ trợ (count)
C_k	Tập các k-tập mục ứng viên (các tập mục thường xuyên tiềm năng). Mỗi phần tử của tập này có 2 trường:
	i) Tập mục (itemsets)
	ii) Độ hỗ trợ (count)

Ý tưởng chính của thuật toán như sau: Sinh ra các tập mục ứng viên từ các tập mục thường xuyên ở bước trước, sử dụng kỹ thuật "tỉa" để bỏ đi những tập mục ứng viên không thoả mãn ngưỡng hỗ trợ cho trước. Cơ sở của kỹ thuật này là tính chất Apriori: Bất kỳ tập con nào của tập mục thường xuyên cũng phải là tập mục thường xuyên. Vì vậy, các tập mục ứng viên gồm k mục có thể được sinh ra bằng cách kết nối các tập mục thường xuyên có (k-1) mục và loại bỏ tập mục ứng viên nếu nó có chứa bất kỳ một tập con nào không phải là thường xuyên.

Thuật toán duyệt cơ sở dữ liệu nhiều lần. Mỗi lần duyệt, thuật toán thực hiện hai bước: bước kết nối và bước tỉa. Trong lần lặp thứ k, thuật toán nối hai (k-1) - tập mục để sinh ra k - tập mục, sử dụng tính chất Apriori để tỉa các tập ứng viên.

3.4.2. Thuật toán FP_growth

Thuật toán Apriori có chi phí lớn nhưng lại kém hiệu quả. Để khắc phục nhược điểm này, J. Han, J Pei, Y. Yin và R. Mao đề xuất thuật toán FP-growth. Thuật toán FP-growth được xây dựng với 3 kỹ thuật chính:

(1) Nén dữ liệu thích hợp vào một cấu trúc cây gọi là cây FP-tree. Chỉ có các 1-tập mục (1-item) ở trong cây và các nút của cây được sắp xếp để các nút xuất hiện thường xuyên hơn có thể dễ dàng chia sẻ với các nút xuất hiện ít hơn.

(2) Thực hiện phương pháp khai phá phát triển (growth) từng đoạn dựa trên cây FP-tree gọi là phương pháp FP-growth.

(3) Kỹ thuật tìm kiếm được dùng ở đây là dựa vào sự phân chia, "chia để trị", phân rã nhiệm vụ khai phá thành các nhiệm vụ nhỏ hơn.

Thuật toán FP-growth do nén toàn bộ CSDL lên một cấu trúc dữ liệu nhỏ hơn là cây FP-tree nên tránh được việc duyệt nhiều lần CSDL (thuật toán chỉ duyệt cơ sở dữ liệu 2 lần). Tiếp theo thuật toán khai phá cây bằng cách phát triển dần các mẫu mà không sinh các tập mục ứng viên, do đó tránh được khối lượng tính toán lớn. Phương pháp FP- growth đã chứng tỏ được tính hiệu quả của nó và có thể thực hiện khai phá cho cả các mẫu ngắn và dài, nhanh hơn thuật toán Apriori, luôn chỉ cần duyệt CSDL 2 lần.

Thuật toán FP- growth thực hiện như sau: Đầu tiên, thuật toán duyệt CSDL lần thứ nhất để tính độ hỗ trợ của từng mục (đếm số lần xuất hiện của từng mục).

Tiếp đến, những mục không đủ độ hỗ trợ bị loại. Các mục còn lại được sắp theo thứ tự giảm dần của độ hỗ trợ (cũng tức là giảm dần theo số lần xuất hiện trong CSDL), ta nhận được danh sách L các mục đã sắp.

Duyệt CSDL lần thứ hai, với mỗi giao tác t, loại các mục không đủ độ hỗ trợ, các mục còn lại theo thứ tự giống như xuất hiện trong L (tức là thứ tự giảm dần theo độ hỗ trợ) được cất vào cây FP-tree.

Phần tiếp theo thuật toán khai phá tìm các mẫu thường xuyên trên cây FP-tree đã xây dựng mà không cần duyệt lại CSDL nữa.

Để hiểu phương pháp này làm việc thế nào, ta xét khai phá CSDL giao tác DB sau với độ hỗ trợ tối thiểu minsup = 3/5.

Thuật toán kinh điển Apriori tìm tập mục thường xuyên theo cách sinh ra các ứng cử viên và duyệt CSDL để kiểm tra, thuật toán FP-growh không khai phá theo cách của thuật toán Apriori mà nén các giao tác của CSDL lên cấu trúc cây FP-Tree, sau đó thực hiện khai phá trên cây này. Thuật toán sinh luật từ tập mục thường xuyên cũng đã được trình bày cụ thể.

3.5. Các hướng chính mở rộng của Khai phá luật kết hợp

Lĩnh vực KPLKH cho đến nay đã được nghiên cứu và phát triển theo nhiều hướng khác nhau. Các hướng chính mở rộng là:

- Luật kết hợp nhị phân (Binary association rule): Là hướng nghiên cứu đầu tiên của luật kết

hợp. Theo dạng luật kết hợp này thì các items chỉ được quan tâm là có hay không có xuất hiện trong cơ sở dữ liệu giao tác (Transaction database). Thuật toán tiêu biểu nhất của khai phá dạng luật này là thuật toán Apriori.

- Luật kết hợp có thuộc tính số và thuộc tính hạng mục: Các CSDL thực tế thường có các thuộc tính đa dạng (như nhị phân, số, mục (categorial)...) chứ không nhất quán ở một dạng nào cả. Vì vậy, để KPLKH trong các CSDL này các nhà nghiên cứu đề xuất một số phương pháp rời rạc hóa nhằm chuyển CSDL cần khai phá về dạng nhị phân để có thể áp dụng các thuật toán đã có. Luật kết hợp với thuộc tính được đánh trọng số trong CSDL thường không có vai trò như nhau. Một số mục dữ liệu quan trọng và được chú trọng hơn các mục dữ liệu khác. Vì vậy, trong quá trình tìm kiếm, các luật từ mục dữ liệu được đánh trọng số theo mức độ xác định nào đó, ta thu được những luật "hiếm" (tức là có độ hỗ trợ thấp nhưng mang nhiều ý nghĩa).

Thuật toán kinh điển Apriori tìm tập mục thường xuyên theo cách sinh ra các ứng cử viên và duyệt CSDL để kiểm tra, thuật toán FP-growh không khai phá theo cách của thuật toán Apriori mà nén các giao tác của CSDL lên cấu trúc cây FP-Tree, sau đó thực hiện khai phá trên cây này. Thuật toán sinh luật từ tập mục thường xuyên cũng đã được trình bày cụ thể.

Ngoài một số phương pháp chính trong KPDL đã trình bày ở trên, còn có những biến thể của KPLKH:

- Luật kết hợp tiếp cận theo hướng tập thô (mining association rule base on rough set): tìm kiếm luật kết hợp dựa trên lí thuyết tập thô.

Luật kết hợp nhiều mức (multi-level association ruls): với cách tiếp cận luật kết hợp này sẽ tìm kiếm thêm những luật có dạng: mua máy tính PC thì mua hệ điều hành Window AND, mua phần mềm văn phòng Microsoft Office,...

- Luật kết hợp mờ (fuzzy association rule): Với những khó khăn gặp phải khi rời rạc hóa các thuộc tính số, các nhà nghiên cứu đề xuất luật kết hợp mờ khắc phục hạn chế đó và chuyển luật kết hợp về một dạng gần gũi hơn.

- Khai phá luật kết hợp song song (parallel mining of association rule): Nhu cầu song song hóa và xử lý phân tán là cần thiết vì kích thước dữ liệu ngày càng lớn nên đòi hỏi tốc độ xử lý phải được đảm bảo.

Những mở rộng, biến thể của KHLKH trên đây cho phép ta tìm kiếm luật kết hợp một cách linh hoạt trong những cơ sở dữ liệu lớn. Ngoài ra, còn một số khái niệm mở rộng của các luật kết hợp, đó là: Luật kết hợp định lượng, Luật kết hợp tổng quát,... Việc KPLKH dựa trên các khái niệm mở rộng này cho phép phát hiện nhiều luật kết hợp mà các thuật toán KPLKH cơ sở không tìm thấy. Ví dụ, với luật kết hợp định lượng cho phép người ta phát biểu một luật có dạng như sau:"Nếu các khách hàng mua ít nhất 3 mặt hàng A thì cũng mua từ 5 đến 10 mặt hàng B". Bên cạnh đó, các nhà nghiên cứu còn chú trọng đề xuất các thuật toán nhằm tăng tốc quá trình tìm kiếm luật kết hợp trong CSDL.

4. Thực nghiệm khai phá luật kết hợp Chương trình ứng dụng KPLKH đã thực hiện thành công, cho kết quả tìm tập mục thường xuyên và luật kết hợp từ CSDL bán hàng tại siêu thị tỉnh Bắc Giang.

Kết quả thực nghiệm KPDL trên tệp Input.txt đã khẳng định những vấn đề lý thuyết trong KPLKH đã trình bày ở trên.

Qua thực nghiệm với các ngưỡng độ hỗ trợ và độ tin cậy khác nhau đã nhận thấy rằng: Khi độ hỗ trợ càng thấp, số tập mục thường xuyên tìm thấy càng nhiều, độ tin cậy càng cao, sinh ra càng ít luật kết hợp.

Kết quả khai phá tập mục thường xuyên và luật kết hợp do chương trình thực nghiệm tìm được đã hỗ trợ rất tốt cho các nhà quản lý siêu thị trong việc tổ chức kinh doanh.

5. Kết luận

KPDL là một trong những kỹ thuật quan trọng, mang tính thời sự đối với nền CNTT thế giới hiện nay. Sự bùng nổ thông tin cùng với sự phát triển và ứng dụng ngày càng rộng rãi của CNTT trong mọi lĩnh vực đã khiến nhu cầu xử lý những khối dữ liệu khổng lồ để kết xuất ra những thông tin, tri thức hữu ích cho người sử dụng một cách tự động, nhanh chóng và chính xác, trở thành nhân tố quan trọng hàng đầu cho mọi thành công của các tổ chức và cá nhân. KPDL đang được áp dụng một cách rộng rãi trong nhiều lĩnh vực kinh doanh và đời sống. Trong thực tế, có rất nhiều tổ chức và công ty lớn trên thế giới đã áp dụng kỹ thuật KPDL vào các hoạt động sản xuất - kinh doanh của mình và thu được những lợi ích to lớn ■

TÀI LIỆU THAM KHẢO:

1 Vũ Đức Thi (1997), Cơ sở dữ liệu-Kiến thức và thực hành, Nhà xuất bản Thống kê, Hà Nội.

2.Vũ Đức Thi (1999), Thuật toán trong tin học. Nhà xuất bản Khoa học và kỹ thuật, Hà Nội.

3.Lê Tiến Vương (1996), Nhập môn cơ sở dữ liệu quan hệ, Nhà xuất bản Khoa học và kỹ thuật, Hà Nội.

4. Demetrovics J, Thi V.D (1988), Some results about functional dependencies, Acta Cybernetical 8,3,273-278.

5. Demetrovics J, Thi V.D (1988), Relations and minumal keys, Acta Cybernetical 8,3,279-285.

Ngày nhận bài: 12/8/2019
Ngày phản biện đánh giá và sửa chữa: 22/8/2019
Ngày chấp nhận đăng bài: 30/8/2019

Thông tin tác giả:
NGUYỄN THỊ VIỆT HÀ
Trưởng khoa Khoa học cơ bản
Trường Cao đẳng Công nghệ và Kinh tế Công nghiệp

AN OVERVIEW ON DATA MINING AND THE ASSOCIATION RULE IN DATA MINING

● **NGUYEN THI VIET HA**
Dean, Faculty of Fundamental Sciences
Industrial Economics and Technology College

ABSTRACT:

Today, the amount of data is growing quickly due to the rapid information technology development in most fields. The traditional data mining methods are no longer meet new requirements and challenges of information technology advancements. Hence, the knowledge discovery technique was developed to solve data mining problems.

Keywords: Data, mining, database, information technology, storage, conversion.