A Theoretical Framework for Association Mining based on the Boolean Retrieval Model*

Peter Bollmann-Sdorra[1], Alaaeldin M. Hafez[2] and Vijay V. Raghavan[2]

Abstract. Data mining has been defined as the non- trivial extraction of implicit, previously unknown and potentially useful information from data. Association mining is one of the important sub-fields in data mining, where rules that imply certain association relationships among a set of items in a transaction database are discovered.

The efforts of most researchers focus on discovering rules in the form of implications between itemsets, which are subsets of items that have adequate supports. Having itemsets as both antecedent and precedent parts was motivated by the original application pertaining to market baskets and they represent only the simplest form of predicates. This simplicity is also due in part to the lack of a theoretical framework that includes more expressive predicates.

The framework we develop derives from the observation that information retrieval and association mining are two complementary processes on the same data records or transactions. In information retrieval, given a query, we need to find the subset of records that matches the query. In contrast, in data mining, we need to find the queries (rules) having adequate number of records that support them.

In this paper we introduce the theory of association mining that is based on a model of retrieval known as the Boolean Retrieval Model. The potential implications of the proposed theory are presented.

Keywords: Data Mining, Association Mining, Theory of Association Mining, Boolean Retrieval Model.

1 Introduction

Knowledge discovery in databases is the process of identifying useful and novel structure (model) in data [7, 8, 12]. Data Mining is considered as one of the main steps in the knowledge discovery process and it is concerned with algorithms used to extract potentially valuable patterns, associations, trends, sequences and dependencies in data. Other steps in knowledge discovery include data preparation, data selection and data cleaning. Association mining is one of the central tasks in data mining [7, 14, 19]. Association mining is the process that discovers dependencies among values of certain attributes on values of some other attributes [1-4,6, 12].

---

* This research was supported in part by the U.S. Department of Energy, Grant No. DE-FG02-97ER1220.

[1] Fachbereich Informatik, FR 6-9, Technical University of Berlin, Franklin Str. 28/29, 10857 Berlin.

[2] ahafez(raghavan)@cacs.louisiana.edu, The Center for Advanced Computer Studies, University of Louisiana, Lafayette, LA70504-4330. USA.

Data mining techniques can discover rules that most traditional business analysis and statistical techniques fail to deliver. Furthermore, the application of data mining techniques enhances the value of data by converting expensive volumes of data into valuable rules for future tactical and strategic business development. Unfortunately, most mining techniques focus on only a narrow view of the data mining problem [7, 8, 9, 13, 15]. Researchers focus on discovering rules in the form of implications between itemsets, which are subsets of items that have adequate supports [1-4, 6, 10, 11, 14, 18, 19]. Having frequent itemsets as both antecedent and precedent parts was motivated by the original application pertaining to market baskets and they represent only the simplest form of predicates. This simplicity is also due in part to the lack of a theoretical framework that includes more expressive predicates. Unlike information retrieval systems that are supported by a strong theoretical background [5, 16, 17], where it provides advanced capabilities that give the user the power to ask more sophisticated and pertinent questions. It empowers the right people by providing the specific information they need.

The framework we develop derives from the observation that information retrieval and association mining are two complementary processes on the same data records or transactions. In information retrieval, given a query, we need to find the subset of records that matches the query. In contrast, in data mining, we need to find the queries (rules) having adequate number of records that support them.

In this paper we introduce the theory of association mining that is based on a model of retrieval known as the Boolean Retrieval Model. The theory for association mining based on this model offers a number of insights:

- a Boolean query that uses only the AND operator (i.e. a conjunction) is analogous to an itemset,
- a general Boolean query (involving AND, OR or NOT operators) has interpretation as a generalized itemset,
- notions of support of itemsets and confidence of rules can be dealt with uniformly, and
- an event algebra can be defined, involving all possible transaction subsets, to formally obtain a probability space.

In section 2, we give the problem definition. The generalized itemsets and Boolean queries are introduced in section 3. In section 4, the rules and their response strengths are given. Finally, the paper is concluded in section 5.

2. Problem Definition

2.1 Notation

- I Set of all items $\{i_1, i_2, \ldots, i_n\}$
- 2^I Set of all possible transactions.
- t A transaction

- T Set of transactions $\{t_1, t_2, \ldots, t_m\}$
- f(t) Frequency of transaction t
- w'(t) Weight of t
- w(t) Normalized weight of t
- q A query; a Boolean expression on items I
- Q* Set of all queries q
- Q Set of all conjunctive queries
- A An item-set
- A_q The item-set of all items corresponding to $q \in Q$
- RS (q) The response set of query q
- R(q) The response of query q
- $SS(A_q)$ Support set of item-set A_q
- $S(A_q)$ Support of item-set A_q

2.2 Association Mining

Association mining was introduced by Agrawal et al.[1], it has emerged as a prominent research area. The association mining problem also referred to as the *market basket* problem can be formally defined as follows. Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of items as $S = \{s_1, s_2, \ldots, s_m\}$ be a set of transactions, where each transaction $s_i \in S$ is a set of items that is $s_i \subseteq I$. An *association rule* denoted by $X \Rightarrow Y$, where $X, Y \subset I$ and $X \cap Y = \Phi$, describes the existence of a relationship between the two itemsets X and Y.

Several measures have been introduced to define the *strength* of the relationship between itemsets X and Y such as support, confidence, and interest. The definitions of these measures, from a probabilistic model are given below.

I. *Support* $(X \Rightarrow Y) = P(X, Y)$, or the percentage of transactions in the database that contain both *X* and *Y*.

II. *Confidence* $(X \Rightarrow Y) = P(X, Y) / P(X)$, or the percentage of transactions containing *Y* in transactions those contain *X*.

III. *Interest*$(X \Rightarrow Y) = P(X, Y) / P(X)P(Y)$ represents a test of statistical independence.

2.3. Boolean Association Mining

Given a set of items $I = \{i_1, i_2, \ldots, i_n\}$, a transaction t is defined as a subset of items such that $t \in 2^I$, where $2^I = \{\varnothing, \{i_1\}, \{i_2\}, \ldots, \{i_n\}, \{i_1, i_2\}, \ldots, \{i_1, i_2, \ldots, i_n\}\}$. In reality, not all possible transactions might occur. For example, transaction $t = \varnothing$ is excluded.

Let $T \subseteq 2^I$ be a given set of transactions $\{t_1, t_2, \ldots, t_m\}$. Every transaction $t \in T$ has an assigned weight w'(t). Several possible weights could be considered,

w'(t)	=	1, for all transactions $t \in T$.
w'(t)	=	f(t), where f(t) is the frequency of transaction t, for all transactions $t \in T$, i.e., how many times the transaction t was repeated in our database.
w'(t)	=	|t| * g(t) for all transactions $t \in T$, where |t| is the cardinality of t, and g(t) could be either one of the weight functions w'(t)'s defined in (i) and (ii). In this case, longer transactions get higher weight.
w'(t)	=	v(t) * f(t) for all transactions $t \in T$, where v(t) could be the sum of the prices or profits of those items in t.

The weights w's are normalized to

$$w(t) = \frac{w'(t)}{\sum_{\forall t' \in T} w'(t')}, \text{ and } \sum_{\forall t \in T} w(t) = 1$$

Example 2.1: Let I = {beer, milk, bread} be the set of all items, where price(beer) = 5, price(milk) = 3, and price(bread) = 2. A set of transactions T along with their frequencies are given in the following table,

#	t	f(t)
1	{beer}	22
2	{milk}	8
3	{bread}	10
4	{beer, bread}	20
5	{milk, bread}	25
6	{beer, milk, bread}	15

Using weight definitions (i)-(iv), the corresponding weights are

(i)

T	{beer}	{milk}	{bread}	{beer,bread}	{milk,bread}	{beer,milk,bread}
w'(t)	1	1	1	1	1	1
w(t)	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$

(ii)

T	{beer}	{milk}	{bread}	{beer,bread}	{milk,bread}	{beer,milk, bread}
w'(t)	22	8	10	20	25	15
w(t)	0.22	0.08	0.1	0.2	0.25	0.15

(iii) using (ii) as g(t)

T	{beer}	{milk}	{bread}	{beer,bread}	{milk,bread}	{beer,milk,bread}
w'(t)	22	8	10	40	50	45
w(t)	0.13	0.05	0.06	0.23	0.27	0.26

(iv)

T	{beer}	{milk}	{bread}	{beer,bread}	{milk,bread}	{beer,milk, bread}
Price(t)	5	3	2	7	5	10
w'(t)	110	24	20	140	125	150
w(t)	0.19	0.04	0.04	0.25	0.22	0.26

3. Item-Sets and Their Representation as Queries

Item-sets describe items in transactions. In the following three definitions, we give the formal definitions for expressing item-sets as queries (i.e., logical expressions), the set of transactions making up the response set of a query, and support.

Definition 3.1: For a given set of items I, the set Q of all possible queries associated with item-sets created from I is defined as follows.

$i \in I \Rightarrow i \in Q$,

$q, q' \in Q \Rightarrow q \wedge q' \in Q$

These are all.

Definition 3.2: For any query $q \in Q$, the response set of q, RS(q), is defined as follows:

For all atomic $i \in Q$, $RS(i) = \{t \in T \mid i \in t\}$

$RS(q \wedge q') = RS(q) \cap RS(q')$

Definition 3.3: Let $q = (i_1 \wedge i_2 \wedge \ldots \wedge i_k)$ and A_q denote the item-set associated with q; that is, $A_q = \{i_1, i_2, \ldots, i_k\}$, the support of A_q is defined as

$$S(A_q) = \sum_{\forall\, t \in RS(q)} w(t), \text{ where } q = (i_1 \wedge i_2 \wedge \ldots \wedge i_k).$$

Lemma 3.1: The support set of A_q ; $SS(A_q)$, equals to RS(q).

Proof: Following the usual definition of a support set, the proof of lemma 3.1 is straightforward.

Example 3.1: Let w(t) be defined as in case (ii) in example 2.1. Definitions 3.1, 3.2 and 3.3 are illustrated in the following table.

	q	A_q	RS(q)	$S(A_q)$	$\frac{S(A_q)}{S(\{x,y,z\})}$
x	beer	{beer}	$\{t_1,t_4,t_6\}$	0.57	3.8
y	milk	{milk}	$\{t_2,t_5,t_6\}$	0.48	3.1
z	bread	{bread}	$\{t_3,t_4,t_5,t_6\}$	0.7	4.7
	$beer \wedge milk$	{beer,milk}	$\{t_6\}$	0.15	1.0
	$beer \wedge bread$	{beer,bread}	$\{t_4,t_6\}$	0.35	2.3
	$milk \wedge bread$	{milk,bread}	$\{t_5,t_6\}$	0.4	2.7
	$beer \wedge milk \wedge bread$	{beer,milk,bread}	$\{t_6\}$	0.15	1.0

Example 3.2: Let w(t) be defined as in case (iv) in example 2.1. The corresponding table is

	q	A_q	RS(q)	$S(A_q)$	$\frac{S(A_q)}{S(\{x,y,z\})}$
x	beer	{beer}	$\{t_1,t_4,t_6\}$	0.7	2.7
y	milk	{milk}	$\{t_2,t_5,t_6\}$	0.52	2.0
z	bread	{bread}	$\{t_3,t_4,t_5,t_6\}$	0.77	3.0
	beer$\wedge$milk	{beer,milk}	$\{t_6\}$	0.26	1.0
	beer$\wedge$bread	{beer,bread}	$\{t_4,t_6\}$	0.51	2.0
	milk$\wedge$bread	{milk,bread}	$\{t_5,t_6\}$	0.48	1.8
	beer$\wedge$milk$\wedge$bread	{beer,milk,bread}	$\{t_6\}$	0.26	1.0

By comparing the support values in examples 3.2 and 3.1, and because the weight values of larger transactions are considered, in example 3.2, all support values are higher than those in example 3.1. But when the relative support formula (i.e., $\frac{S(A_q)}{S(\{x,y,z\})}$) is used, all support values in example 3.1 are higher than those in example 3.2.

Lemma 3.2: For queries q, q_1, q_2 and q_3, the following axioms hold:

$RS(q \wedge q) = RS(q)$
$RS((q_1 \wedge q_2) \wedge q_3) = RS(q_1 \wedge (q_2 \wedge q_3))$
$RS(q_1 \wedge q_2) = RS(q_2 \wedge q_1)$

Example 3.3: $RS((x_1 \wedge x_2) \wedge (x_3 \wedge x_2)) = RS(x_1 \wedge x_2 \wedge x_3)$

In order to apply probabilities, we need a full set of algebra, and so, we need to redefine item-sets.

Definition 3.4: For a given set of items I, the set Q* of all possible queries is defined as follows.
$i \in I \Rightarrow i \in Q^*$,
$q, q' \in Q^* \Rightarrow q \wedge q' \in Q^*$
$q, q' \in Q^* \Rightarrow q \vee q' \in Q^*$
$q \in Q^* \Rightarrow \neg q \in Q^*$
These are all.

Definition 3.5: For any query $q \in Q^*$, the response set of transactions, R (q) is defined as

For all $i \in Q^*$, $RS(i) = \{t \in T \mid i \in t\}$
$RS(q \wedge q') = RS(q) \cap RS(q')$
$RS(q \vee q') = RS(q) \cup RS(q')$
$RS(\neg q) = T - RS(q)$

In analogy to lemma 3.2, we may apply all axioms of Boolean algebra on elements of Q*. Consequently, we can show that response sets of equivalent Boolean expressions are equal.

Theorem 3.1: If q is a transformation of q' that is obtained by applying the rules of Boolean algebra, then

$$RS(q)= RS(q')$$

Each $q \in Q^*$ can be considered as a generalized itemset. The itemsets investigated in earlier works only consider $q \in Q$.

Lemma 3.3: $\{RS(q) \mid q \in Q^*\}=2^T$
Proof:
(i) $\{RS(q) \mid q \in Q^*\} \subseteq 2^T$ is trivial
(ii) We need to prove $2^T \subseteq \{RS(q) \mid q \in Q^*\}$.
Let $T_s = \{t_1, t_2, \ldots, t_k\} \in 2^T$, where $t_1, t_2, \ldots, t_k$ are transactions. Let $t_i = \{x_1, x_2, \ldots, x_p\}$ and $\{y_1, y_2, \ldots, y_q\} = I - T_s$. Now, if $q_i = x_1 \wedge x_2 \wedge \ldots \wedge x_p \wedge \neg y_1 \wedge \neg y_2 \wedge \ldots \wedge \neg y_q$, then $RS(q_i) = t_i$. Now, let $q = q_1 \vee q_2 \vee \ldots \vee q_k$, then $RS(q) = T_s$.

Let P be a function, where $P : 2^T \rightarrow [0,1]$. For RS(q), $q \in Q^*$, we define

$$P(RS(q)) = R(q)$$

P is well defined because,

$$RS(q_1) = RS(q_2) \;\Rightarrow\; R(q_1) = R(q_2)$$

Theorem 3.2: $(T, 2^T, P)$ is a probability space.

Proof: Check Kolmogaroff axioms.

Lemma 3.3 implies that for every subset T_s of T, such that the cardinality of T_s is greater than or equal to the minimum support, there exists at least one query $q \in Q^*$, where q is frequent. The generated queries, which are generalized itemsets, could have different complexities and lengths. In order to only generate understandable queries, new restrictions or measures, such as, compactness and simplicity, should be introduced.

4. Rules and Their Response Strengths

Let q and q' be conjunctive queries in Q.

Definition 4.1: The confidence of a rule $A_q \Rightarrow A_{q'}$ is defined as $\frac{R(q \wedge q')}{R(q)}$, (assume $A_q \cap A_{q'} = \phi$)

Definition 4.2: The interest of a rule $A_q \Rightarrow A_{q'}$ is defined as $\frac{R(q \wedge q')}{R(q) * R(q')}$, (assume $A_q \cap A_{q'} = \phi$)

Definition 4.3: The support of a rule $A_q \Rightarrow A_{q'}$ is defined as
$S(A_q \Rightarrow A_{q'}) = R(\neg q \vee q')$

Lemma 4.1: For a rule $A_q \Rightarrow A_{q'}$, $S(A_q \Rightarrow A_{q'}) = 1 - R(q) + R(q \wedge q')$

Proof:

$$\begin{aligned}
SS(A_q \Rightarrow A_{q'}) &= RS(\neg q \vee q') \\
&= RS(\neg q) \cup RS(q') \\
&= (T - RS(q)) \cup RS(q') \\
&= T - (RS(q) - RS(q \wedge q'))
\end{aligned}$$

$$S(A_q \Rightarrow A_{q'}) = 1 - R(q) + R(q \wedge q')$$

Lemma 4.2: For a rule $A_q \Leftrightarrow A_{q'}$,
$S(A_q \Leftrightarrow A_{q'}) = 1 - R(q) - R(q') + 2 * R(q \wedge q')$

Proof:

$$\begin{aligned}
SS(A_q \Leftrightarrow A_{q'}) &= RS(\neg q \vee q') \cap RS(q \vee \neg q') \\
&= (RS(\neg q) \cup RS(q')) \cap (RS(q) \cup RS(\neg q')) \\
&= ((T - RS(q)) \cup RS(q')) \cap ((RS(q) \cup (T - RS(q')) \\
&= (T - (RS(q) \cup RS(q'))) \cup (RS(q) \cap RS(q'))
\end{aligned}$$

$$S(A_q \Leftrightarrow A_{q'}) = 1 - R(q) - R(q') + 2 * R(q \wedge q')$$

For a query (or a rule) $A_q \Rightarrow A_{q'}$, the confidence measure is given in definition 4.1, and the support measure is given in definition 4.2 and lemma 4.1. Although the two measures look similar, as explained below, having the support measure instead of the confidence measure could also give information about the interest of a rule.

Let $\delta = R(q) - R(q \wedge q')$. The confidence and support of $A_q \Rightarrow A_{q'}$ are

$$conf(A_q \Rightarrow A_{q'}) = 1 - \frac{\delta}{R(q)}$$

$$S(A_q \Rightarrow A_{q'}) = 1 - \delta$$

In $conf(A_q \Rightarrow A_{q'})$, if $R(q)$ is increased and the value of δ is fixed (i.e., all new transactions are having both *q* and *q'*), the ratio $\frac{\delta}{R(q)}$ will get smaller, and

$\boldsymbol{conf}$ $(A_q \Rightarrow A_{q'})$ will get larger. If we have such a case, where all new transactions are having both q and q', then the rule $A_q \Rightarrow A_{q'}$ should not be considered as an interesting rule. Thus, in the support measure, we not only consider the confidence of a rule but also the interest of this rule.

5. Conclusions

In this paper we introduce the theory of association mining that is based on a model of retrieval known as the Boolean Retrieval Model. The framework we develop derives from the observation that information retrieval and association mining are two complementary processes on the same data records or transactions. In information retrieval, given a query, we need to find the subset of records that matches the query, while in data mining, we need to find the queries (rules) having adequate number of records that support them.

In the traditional association mining formulation, only the conjunction operator is used in forming itemsets (queries). Based on the theory of Boolean retrieval, we generalize the itemset structure by using all Boolean operators. By introducing the notion of support of generalized itemsets, a uniform measure for both itemsets and rules (generalized itemsets) has been developed. Support of a generalized itemset is extended to allow transactions to be weighted so that they can contribute to support unequally. This is achieved by the introduction of weight functions.

Since every subset of *T* has at least one generalized itemset, there can be very large number of frequent generalized itemsets, many of which could have complex structures. In order to only generate understandable queries, new restrictions or measures, such as, compactness and simplicity, should be introduced.

References

[1] R. Agrawal, T. Imilienski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. of the ACM SIGMOD Int'l Conf. On Management of data, May 1993.

[2] R. Agrawal, and R. Srikant, "Fast Algorithms for Mining Association Rules," Proc. Of the 20^{th} VLDB Conference, Santiago, Chile, 1994.

[3] R. Agrawal, J. Shafer, "Parallel Mining of Association Rules," IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, Dec. 1996.

[4] C. Agrawal, and P. Yu, "Mining Large Itemsets for Association Rules," Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 1997.

[5] A. Bookstein and W. Cooper, "A General Mathematical Model for Information Retrieval Systems," Library Quarterly, 46(2), 153-167.

[6] S. Brin, R. Motwani, J. Ullman, and S. Tsur, "Dynamic Itemset Counting and Implication Rules for Market Basket Data," SIGMOD Record (SCM Special Interset Group on Management of Data), 26,2, 1997.

[7] S. Chaudhuri, “Data Mining and Database Systems: Where is the Intersection,” Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 1997.

[8] U. Fayyad, “Mining Databases: Towards Algorithms for Knowledge Discovery,” Data Engineering Bulletin 21(1): 39-48 (1998)

[9] U. Fayyad: Editorial. Data Mining and Knowledge Discovery 2(1): 5-7 (1998)

[10] U. Fayyad, “Data Mining and Knowledge Discovery in Databases: Implications for Scientific Databases,” SSDBM 1997: 2-11

[11] H. Mannila, “Inductive databases and condensed representations for data mining,” Proc. International Logic Programming Symposium, 1997, pp. 21--30.

[12] H. Mannila, “Methods and problems in data mining,” In Proceedings of the International Conference on Database Theory, Delphi, Greece, January 1997. Springer-Verlag.

[13] H. Mannila, “Data mining: machine learning, statistics, and databases,” Eight International Conference on Scientific and Statistical Database Management, Stockholm. 1996.

[14] H. Mannila, H. Toivonen, and A. Verkamo, “Efficient Algorithms for Discovering Association Rules,” AAAI Workshop on Knowledge Discovery in databases (KDD-94) , July 1994.

[15] A. Netz, S. Chaudhuri, J. Bernhardt and U. Fayyad, “Integration of Data Mining with Database Technology,” VLDB 2000: 719-722

[16] S. E. Robertson, “Theories and Models in Information Retrieval,” Journal of Documentation, 33, 126-149.

[17] G. Salton, Automatic Text Processing, Reading, MA: Addison Wesley.

[18] M. Zaki and M. Ogihara, “Theoretical foundations of association rules,” In 3rd ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, June 1998.

[19] M. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, “ New Algorithms for Fast Discovery of Association Rules,” Proc. Of the 3rd Int’l Conf. On Knowledge Discovery and data Mining (KDD-97), AAAI Press, 1997.