Extraction of Coverings as Monotone DNF Formulas*

Kouichi Hirata† Ryosuke Nagazumi‡** Masateru Harao†

† Department of Artificial Intelligence
‡ Graduate School of Computer Science and Systems Engineering
Kyushu Institute of Technology
Kawazu 680-4, Iizuka 820-8502, Japan
† {hirata,harao}@ai.kyutech.ac.jp
‡ nagazumi@dumbo.ai.kyutech.ac.jp

Abstract. In this paper, we extend monotone monomials as large itemsets in association rule mining to *monotone DNF formulas*. First, we introduce not only the minimum support but also *the maximum overlap*, which is a new measure how much all pairs of two monomials in a monotone DNF formula commonly cover data. Next, we design the algorithm *dnf_cover* to extract *coverings as monotone DNF formulas* satisfying both the minimum support and the maximum overlap. In the algorithm *dnf_cover*, first we collect the monomials of which support value is not only more than the minimum support but also less than the minimum support as *seeds*. Secondly we construct the coverings as monotone DNF formulas, by combining monomials in seeds under the minimum support and the maximum overlap. Finally, we apply the algorithm *dnf_cover* to bacterial culture data.

1 Introduction

The purpose of *data mining* is to extract hypotheses to explain a database. An *association rule* is one of the most famous forms of hypotheses in data mining or association rule mining [1, 6, 7, 12]. In order to extract association rules from a transaction database, the algorithm Apriori, introduced by Agrawal *et al.* [2, 3], extracts *large itemsets* as sets of variables satisfying the *minimum support* for the transaction database. Then, by combining variables in each large itemset, we can extract association rules satisfying both the minimum support and the *minimum confidence* for the transaction database.

The disadvantage of Apriori, however, is that, if we extract association rules that explain a transaction database nearly overall, then the extracted large itemsets only reflect the data with very high frequency and they are not interesting

* This work is partially supported by Japan Society for the Promotion of Science, Grants-in-Aid for Encouragement of Young Scientists (B) 15700137 and for Scientific Research (B) 13558036. Also this paper will be published in Proc. 6th International Conference on Discovery Science (DS2003), Lecture Notes in Artificial Intelligence **2483**, 165–178, 2003 ©Springer-Verlag.

** Current address: Zenrin Co., Ltd.

in general. Furthermore, if we deal with a transaction database with one class *class*, it is natural to extract an association rule $x_1 \wedge \cdots \wedge x_n \rightarrow$ *class* with the consequence *class* rather than the association rules constructed from the rule generation in [2, 3], for example, $x_2 \wedge \cdots \wedge x_n \rightarrow x_1$, from a large itemset $\{x_1, \ldots, x_n\}$. In order to extract the association rules with the consequence *class* that explain the transaction database nearly overall, in this paper, we regard a large itemset as a *monotone monomial* and extend it to a *monotone DNF formula* as a disjunction of monotone monomials.

It is a problem for the above purpose that there exist extremely many monotone DNF formulas satisfying only the minimum support. Then, in this paper, we assume that each monotone monomial in a monotone DNF formula should cover data in a transaction database *without overlapping* preferably. Hence, we introduce a new measure *overlap* as the cardinality of an *overlap set*. Here, the overlap set of two monomials t and s is the set of ID's of data that t and s commonly cover, and the overlap set of a monotone DNF formula $f = t_1 \vee \cdots \vee t_m$ is the union of overlap sets of each two terms t_i and t_j in f $(1 \leq i \neq j \leq m)$. We call a monotone DNF formula satisfying the minimum support and the maximum overlap a *covering as monotone DNF formulas*.

Based on the above two measures, we design the algorithm *dnf_cover* to extract coverings as monotone DNF formulas from a transaction database, by extending the algorithm Apriori. In this paper, we adopt the minimum support more than 70%. In the algorithm *dnf_cover*, first we collect monotone monomials of which frequency is not only more than the minimum support, which are *coverings as monotone monomials*, but also less than the minimum support as *seeds*. Secondly we construct monotone DNF formulas by combining monomials in seeds under the minimum support and the maximum overlap. Here, we use the monotonicity of overlap sets in order to reduce a search space.

Finally, we give the empirical results by applying the algorithm *dnf_cover* to bacterial culture data, which are full version of data in [9–11]. We use two kinds of such data, one is MRSA (methicillin-resistant Staphylococcus aureus) data with 118 records and another is Anaerobes data with 1064 records. Both of them consist of data with 93 attributes. Then, we evaluate the number and the length of extracted coverings and investigate the extracted coverings from a medical viewpoint.

2 Coverings as Monotone DNF Formulas

For a set S, $|S|$ denotes the cardinality of S.

Let X be a finite set and we call an element of X a *variable*. A *monotone monomial* in X is a finite conjunction of variables in X. For a monotone monomial $t = x_1 \wedge \cdots \wedge x_n$, we denote the number n of variables in t by $|t|$. Sometimes we identify a monotone monomial $x_1 \wedge \cdots \wedge x_n$ with a set $\{x_1, \ldots, x_n\}$ of variables in X. A *monotone DNF formula* in X is a finite disjunction of monotone monomials in X.

A *transaction database* in X is a set $\mathcal{D}$ of pairs of a natural number and a finite set of variables in X, that is, $\mathcal{D} = \{(tid, T_{tid}) \mid T_{tid} \subseteq X\}$ [2,3]. Here, the natural number *tid* and the set T_{tid} of variables are called a *transaction ID* and a *transaction*, respectively. For an attribute-value database, the form of "attribute = attribute value" is regarded as a variable in X. Sometimes we omit the statement "in X" in monotone monomials, monotone DNF formulas and transaction databases.

We introduce two measures, a usual measure *frequency* defined as the cardinality of a *cover set* and a new measure *overlap* defined as the cardinality of an *overlap set*.

Definition 1. Let $\mathcal{D}$ be a transaction database. Then, the *cover set* $cvs_{\mathcal{D}}(t)$ of a monomial t for $\mathcal{D}$ is defined in the following way.

$$cvs_{\mathcal{D}}(t) = \{tid \mid (tid, T_{tid}) \in \mathcal{D}, t \subseteq T_{tid}\}.$$

Furthermore, the *cover set* $cvs_{\mathcal{D}}(f)$ of a monotone DNF formula $f = t_1 \vee \cdots \vee t_m$ for $\mathcal{D}$ is defined as the union of cover sets of all monomials in f.

$$cvs_{\mathcal{D}}(f) = \bigcup_{i=1}^{m} cvs_{\mathcal{D}}(t_i).$$

The *frequency* of f in $\mathcal{D}$ is defined as $|cvs_{\mathcal{D}}(f)|$ and denoted by $freq_{\mathcal{D}}(f)$.

Definition 2. Let $\mathcal{D}$ be a transaction database. Then, the *overlap set* $ols_{\mathcal{D}}(t)$ of monomials t and s for $\mathcal{D}$ is defined in the following way.

$$ols_{\mathcal{D}}(t, s) = cvs_{\mathcal{D}}(t) \cap cvs_{\mathcal{D}}(s).$$

Furthermore, the *overlap set* $ols_{\mathcal{D}}(f)$ of a monotone DNF formula $f = t_1 \vee \cdots \vee t_m$ in $\mathcal{D}$ is defined as the union of all overlap sets of two distinct monomials in f.

$$ols_{\mathcal{D}}(f) = \bigcup_{1 \leq i \neq j \leq m} ols_{\mathcal{D}}(t_i, t_j).$$

The *overlap* of f in $\mathcal{D}$ is defined as $|ols_{\mathcal{D}}(f)|$ and denoted by $ol_{\mathcal{D}}(f)$.

For δ $(0 < \delta \leq 1)$ and η $(0 < \eta \leq 1)$, we say that a monotone DNF formula f is a *covering as monotone DNF formulas* of $\mathcal{D}$ under the minimum support δ and the maximum overlap η if f satisfies that $freq_{\mathcal{D}}(f) \geq \delta|\mathcal{D}|$ and $ol_{\mathcal{D}}(f) \leq \eta|\mathcal{D}|$. In particular, we say that a monotone monomial t is a *covering as monotone monomials* of $\mathcal{D}$ under the minimum support δ if t satisfies that $freq_{\mathcal{D}}(f) \geq \delta|\mathcal{D}|$. Furthermore, a covering as either monotone monomials or monotone DNF formulas of $\mathcal{D}$ is called a *covering* of $\mathcal{D}$ simply.

3 Extraction of Coverings

In this section, we design the algorithm *dnf_cover* to extract coverings of transaction databases under the minimum support δ and the maximum overlap η, by extending Apriori [2, 3].

First note that the minimum support is very small (less than 2%) in Apriori [2, 3]. On the other hand, our purpose is to find coverings reflecting much data in a transaction database than large itemsets of Apriori, so our minimum support is much larger than one of Apriori. In this paper, we set the minimum support to more than 70% in the empirical results in Section 4.

In order to extract coverings under the minimum support δ and the maximum overlap η, we design the algorithm *dnf_cover* described as Figure 1. This algorithm consists of two phases, a *conjunction phase* to construct L_k by Apriori and a *disjunction phase* to monotone DNF formulas from monotone monomials.

The difference between a conjunction phase in *dnf_cover* and Apriori is that we collect monomials not satisfying the minimum support as *seeds* and set them to a variable *SEED* in a conjunction phase. A monotone monomial $t \wedge x$ is collected in *SEED* if $freq_{\mathcal{D}}(t) \geq \delta|\mathcal{D}|$ and $freq_{\mathcal{D}}(t \wedge x) < \delta|\mathcal{D}|$.

Furthermore, in order to avoid to collect monotone monomials with low frequency in seeds, in this paper, we introduce the *minimum monomial support* σ. In *dnf_cover*, we collect the monomial t in *SEED* satisfying that $\sigma|\mathcal{D}| \leq freq_{\mathcal{D}}(t) < \delta|\mathcal{D}|$. If $\sigma = 0$, then R coincides with R' in a conjunction phase of *dnf_cover*.

Theorem 1 (The monotonicity of $ol_{\mathcal{D}}$). For monotone DNF formulas f and g, it holds that $ol_{\mathcal{D}}(f) \leq ol_{\mathcal{D}}(f \vee g)$.

Proof. By the definition, it holds that $ol_{\mathcal{D}}(f \vee g) = |ols_{\mathcal{D}}(f \vee g)|$. Suppose that f and g be monotone DNF formulas $t_1 \vee \cdots \vee t_m$ and $t_{m+1} \vee \cdots \vee t_n$ $(1 \leq m < n)$. Then, $f \vee g = t_1 \vee \cdots \vee t_m \vee t_{m+1} \vee \cdots \vee t_n$. By the definition, the following statement holds.

$$\begin{aligned} ols_{\mathcal{D}}(f \vee g) &= \bigcup_{1 \leq i \neq j \leq n} ols_{\mathcal{D}}(t_i, t_j) \\ &= ols_{\mathcal{D}}(f) \cup \bigcup_{1 \leq i \leq n, m+1 \leq j \leq n} ols_{\mathcal{D}}(t_i, t_j). \end{aligned}$$

Hence, it holds that $ols_{\mathcal{D}}(f) \subseteq ols_{\mathcal{D}}(f \vee g)$, that is, $ol_{\mathcal{D}}(f) \leq ol_{\mathcal{D}}(f \vee g)$. □

Theorem 2. Let t_i $(1 \leq i \leq m)$ be monomials, f a monotone DNF formula $t_1 \vee \cdots \vee t_m$, and g a monotone DNF formula $f \vee t$. Then, the following equation holds.

$$ols_{\mathcal{D}}(g) = ols_{\mathcal{D}}(f) \cup \bigcup_{i=1}^{m} ols_{\mathcal{D}}(t_i, t).$$

Proof. By regarding t as t_{m+1}, the following statement holds.

```
procedure dnf_cover(D, δ, η)
    /* D: a transaction database, X: a set of variables,
        δ: minimum support, η: maximum overlap,
        σ: minimum monomial support */
L_0 ← ∅; L_1 ← X; k ← 0; SEED ← ∅;
while L_k ≠ L_{k+1} do begin /* conjunction phase */
    k ← k + 1; C_{k+1} ← ∅;
    forall t ∈ L_k such that |t| = k do begin
        forall (tid, T_tid) ∈ D do
            if t ⊆ T_tid then
                forall lexicographically larger variables x ∈ L_k
                        than all variables in t do
                    C_{k+1} ← C_{k+1} ∪ {t ∧ x};
    end /* forall*/
    R ← {t ∈ C_{k+1} | freq_D(t) < δ|D|};
    R' ← {t ∈ C_{k+1} | σ|D| ≤ freq_D(t) < δ|D|};
    L_{k+1} ← L_k ∪ (C_{k+1} − R);
    SEED ← SEED ∪ R';
end /* while */
DNF_1 ← L_k; l ← 0; S_0 ← ∅; S_1 ← SEED;
while S_{l+1} ≠ ∅ do begin /* disjunction phase */
    l ← l + 1; DNF_{l+1} ← ∅; S_{l+1} ← ∅;
    forall f ∈ S_l do begin
        forall lexicographically larger elements t ∈ SEED
                than all monomials in f do
            if ol_D(f ∨ t) ≤ η|D| then
                S_{l+1} ← S_{l+1} ∪ {f ∨ t};
                if freq_D(f ∨ t) ≥ δ|D| then
                    DNF_{l+1} ← DNF_{l+1} ∪ {f ∨ t};
    end /* forall */
end /* while */
return ⋃_{i=1}^{l} DNF_i;
```

Fig. 1. The algorithm *dnf_cover* to extract coverings from $\mathcal{D}$

$$ols_{\mathcal{D}}(f \vee t) = \bigcup_{1 \leq i \neq j \leq m+1} ols_{\mathcal{D}}(t_i, t_j)$$
$$= ols_{\mathcal{D}}(f) \cup \bigcup_{i=1}^{m} ols_{\mathcal{D}}(t_i, t_{m+1}).$$

Hence, the statement holds. □

By Theorem 1, once a monotone DNF formula does not satisfy the maximum overlap, no monotone DNF formula obtained by adding monotone monomials to it satisfies the maximum overlap. Then, in a disjunction phase, we construct monotone DNF formulas in the following way.

> While a monotone DNF formula f satisfies the maximum overlap, we connect $t \in SEED$ to f by a disjunction $\vee$. If $f \vee t$ satisfies the minimum support, then add $f \vee t$ to coverings.

Note that, in the construction of S_2, we can obtain the overlap sets $ols_{\mathcal{D}}(t \vee s)$ for each $t, s \in SEED$ such that $t \vee s$ satisfying the maximum overlap. Then, by Theorem 2, the overlap $ol_{\mathcal{D}}(f)$ for $f \in S_l$ $(l \geq 3)$ can be obtained by the overlap sets of each pairs of elements in $SEED$ in S_2.

Example 1. Consider the transaction database $\mathcal{D}$ in Figure 2. Furthermore, assume that the minimum support and the maximum overlap are 80% and 25%, respectively. Also the minimum monomial support σ is set to 0%.

tid	T_{tid}
1	a, c, e, f
2	b, c, e
3	c, e, f
4	a, b, c, f
5	d, e

Fig. 2. A transaction database $\mathcal{D}$

A conjunction phase in *dnf_cover* constructs DNF_1 and $SEED$ as Figure 3. We fix the order in $SEED$ from top-down.

Next, in a disjunction phase, S_l and DNF_l $(l \geq 2)$ as Figure 4 are constructed from the $SEED$. Here, the value *freq* and *ol* in the column "determ." mean that the formula does not satisfy the minimum support and the maximum overlap, respectively, and the value $\bullet$ means that the formula satisfies both. Thus, the algorithm *dnf_cover* adds the formula with *freq* and $\bullet$ to S_l and the formula with $\bullet$ to DNF_l.

Hence, all of the extracted coverings of $\mathcal{D}$ by the algorithm *dnf_cover* is described as follows.

$DNF_1 : c,\ e$
$DNF_2 : a \vee (c \wedge e),\ b \vee f,\ b \vee (c \wedge e),$
$d \vee f,\ d \vee (c \wedge e),$
$DNF_3 : a \vee b \vee d,\ a \vee d \vee (c \wedge e),$
$b \vee d \vee f,\ b \vee d \vee (c \wedge e).$

monotone monomial t	$cvs_{\mathcal{D}}(t)$
c	$1, 2, 3, 4$
e	$1, 2, 3, 5$

monotone monomial t	$cvs_{\mathcal{D}}(t)$
a	$1, 4$
b	$2, 4$
d	5
f	$1, 3, 4$
$c \wedge e$	$1, 2, 3$

Fig. 3. DNF_1 (left) and *SEED* (right) for $\mathcal{D}$

4 Empirical Results from Bacterial Culture Data

In this section, we give the empirical results by applying the algorithm *dnf_cover* to bacterial culture data, which are full version in [11]. We use two kinds of such data, one is MRSA (methicillin-resistant Staphylococcus aureus) data with 118 records and another is Anaerobes data with 1082 records. Both of them consist of data between four years (from 1995 to 1998) with 93 attributes. In this paper, we transform them to transaction databases to applying *dnf_cover*.

In the following tables, δ and η denote the maximum support and the minimum overlap, respectively. Also *max_vars* denotes the maximum number of variables in monotone monomials for each extracted covering. Furthermore, we identify DNF_i with its cardinality $|DNF_i|$. In this section, the minimum monomial support is fixed to 10%.

4.1 MRSA data

The number of extracted coverings from MRSA data by *dnf_cover* is described as Figure 5.

In general, by decreasing the minimum support and by increasing the maximum overlap, the number and the length of extracted coverings are increasing, because decreasing the minimum support is corresponding to increasing the number of elements in *SEED* and DNF_i and increasing the maximum overlap is to decreasing the number of elements in DNF_i. Furthermore, for $\delta = 70\%$ or 80%, $i = 1, 2$ and 3 are corresponding to the largest cardinality of DNF_i under $\eta = 5\%, 10\%$ and 15%, respectively.

Figure 6 describes a part of extracted coverings from MRSA data under the minimum support 80% and the maximum overlap 10%, where the subscript

	formula f	$cvs_{\mathcal{D}}(f)$	$ols_{\mathcal{D}}(f)$	determ.
DNF_2	$a \vee b$	$1,2,4$	4	*freq*
	$a \vee d$	$1,4,5$	$\emptyset$	*freq*
	$a \vee f$	$1,3,4$	1	*freq*
	$a \vee (c \wedge e)$	$1,2,3,4$	1	$\bullet$
	$b \vee d$	$2,4,5$	$\emptyset$	*freq*
	$b \vee f$	$1,2,3,4$	4	$\bullet$
	$b \vee (c \wedge e)$	$1,2,3,4$	2	$\bullet$
	$d \vee f$	$1,3,4,5$	$\emptyset$	$\bullet$
	$d \vee (c \wedge e)$	$1,2,3,5$	$\emptyset$	$\bullet$
	$f \vee (c \wedge e)$	$1,2,3,4$	$1,3$	*ol*
DNF_3	$a \vee b \vee d$	$1,2,4,5$	4	$\bullet$
	$a \vee b \vee f$	$1,2,3,4$	$1,4$	*ol*
	$a \vee b \vee (c \wedge e)$	$1,2,3,4$	$1,2,4$	*ol*
	$a \vee d \vee f$	$1,3,4,5$	$1,4$	*ol*
	$a \vee d \vee (c \wedge e)$	$1,2,3,4,5$	1	$\bullet$
	$a \vee f \vee (c \wedge e)$	$1,2,3,4$	$1,3$	*ol*
	$b \vee d \vee f$	$1,3,4,5$	4	$\bullet$
	$b \vee d \vee (c \wedge e)$	$1,2,3,4,5$	2	$\bullet$
	$b \vee f \vee (c \wedge e)$	$1,2,3,4$	$2,4$	*ol*
	$d \vee f \vee (c \wedge e)$	$1,2,3,4,5$	$1,3$	*ol*
DNF_4	$a \vee b \vee d \vee f$	$1,2,3,4,5$	$1,4$	*ol*
	$a \vee b \vee d \vee (c \wedge e)$	$1,2,3,4,5$	$1,2,4$	*ol*
	$a \vee d \vee f \vee (c \wedge e)$	$1,2,3,4,5$	$1,2,3,4$	*ol*

Fig. 4. S_l and DNF_l for $\mathcal{D}$

δ	70%			80%			90%		
η	5%	10%	15%	5%	10%	15%	5%	10%	15%
DNF_1	150	150	150	7	7	7	1	1	1
DNF_2	121	1243	2293	4	93	227	2	3	5
DNF_3	9	1108	9325	3	46	356	1	7	51
DNF_4	14	133	2464	2	42	182	1	8	27
DNF_5	7	200	2734	4	9	189	–	–	25
DNF_6	6	178	3486	–	2	140	–	–	8
DNF_7	–	54	2662	–	–	73	–	–	–
DNF_8	–	–	89	–	–	9	–	–	–
total	307	3066	23203	20	199	1183	5	19	117
max_vars	7	7	7	2	3	3	1	1	1

Fig. 5. The number of extracted coverings from MRSA data.

DNF	frequency
$((\texttt{Cep1} = \texttt{R}) \wedge (\texttt{PcS} = \texttt{R}))_{78.8} \vee (\texttt{dis} = \texttt{13})_{11.0}$	81.4%
$((\texttt{PcS} = \texttt{R}) \wedge (\texttt{VCM} = \texttt{S}) \wedge (\texttt{beta} = \texttt{0}))_{79.7} \vee (\texttt{dis} = \texttt{13})_{11.0}$	80.5%
$(\texttt{PcB} = \texttt{R})_{79.7} \vee (\texttt{dis} = \texttt{34})_{11.9}$	83.1%
$(\texttt{LCM} = \texttt{R})_{79.7} \vee (\texttt{ward} = \texttt{3})_{10.2}$	80.5%
$(\texttt{CBP} = \texttt{R})_{23.7} \vee (\texttt{year} = \texttt{95})_{50.0} \vee (\texttt{year} = \texttt{96})_{13.6}$	80.5%
$(\texttt{CBP} = \texttt{R})_{23.7} \vee (\texttt{year} = \texttt{95})_{50.0} \vee (\texttt{year} = \texttt{97})_{16.1}$	80.5%

Fig. 6. A part of extracted coverings from MRSA data under the minimum support 80% and the maximum overlap 10%.

number of each monotone monomial denotes its frequency (%). Then, *dnf_cover* extracts the drug-resistant for MRSA, that is, the resistant to benzilpenicillins ($\texttt{PcB} = \texttt{R}$), synthetic penicillins ($\texttt{PcS} = \texttt{R}$) and 1st generation cephems ($\texttt{Cep1} = \texttt{R}$). Furthermore, *dnf_cover* extracts not only the disease information that is cranial nerve ($\texttt{dis} = \texttt{13}$) or nephrostomy ($\texttt{dis} = \texttt{34}$) but also the information that the department is an internal medicine ($\texttt{dept} = \texttt{1}$) or the ward is 3 ($\texttt{ward} = \texttt{3}$). They are possible to be a key of emerging infection.

As another sensitivity of antibiotics, *dnf_cover* extracts the coverings containing the resistant to lincomycins ($\texttt{LCM} = \texttt{R}$), which implies the fact that lincomycins take no effect for MRSA. Also the last two coverings in Figure 6 contain the resistant to carbapenems ($\texttt{CBP} = \texttt{R}$). In particular, we can extract the covering $(\texttt{CBP} = \texttt{S})_{55.1} \vee (\texttt{year} = \texttt{98})_{20.3}$ under the minimum support 70% and the maximum overlap 10%, so these coverings are possible to imply the drug-resistant change for four years.

4.2 Reducing a search space

In *dnf_cover*, we have already introduced the minimum monomial support σ as a threshold to reduce a search space. In this section, we also introduce another threshold called the *maximum monomial support* τ. Then, we replace $R' \leftarrow \{t \in C_{k+1} \mid \sigma|\mathcal{D}| \leq \mathit{freq}_{\mathcal{D}}(t) < \delta|\mathcal{D}|\}$ in *dnf_cover* with

$$R' \leftarrow \{t \in C_{k+1} \mid \sigma|\mathcal{D}| \leq \mathit{freq}_{\mathcal{D}}(t) \leq \tau|\mathcal{D}|\}.$$

Hence, *dnf_cover* outputs the coverings as monotone DNF formulas of which monotone monomial is *uniformly* frequent, that is, of which monotone monomial t always satisfies that $\sigma|\mathcal{D}| \leq \mathit{freq}_{\mathcal{D}}(t) \leq \tau|\mathcal{D}|$.

Figure 7 describes the number of extracted coverings from MRSA data by *dnf_cover* with the maximum monomial support τ. Here, $\tau = \delta$ means the results without τ, which are the same results described by Figure 5.

Note that $|DNF_1|$ is the same result without τ, because the construction of DNF_1 is independent from the introduction of τ to *dnf_cover*. On the other hand, if $\tau = 50\%$, then $|DNF_i|$ for $i \geq 3$ ($\eta = 5\%$), $i \geq 4$ ($\eta = 10\%$) and $i \geq 5$ ($\eta = 15\%$) are the same results without τ, respectively. If $\tau = 30\%$, then $|DNF_i|$ for $i \geq 7$ is the same result without τ. Hence, *dnf_cover* with the maximum

δ	70%								
η	5%			10%			15%		
τ	δ	50%	30%	δ	50%	30%	δ	50%	30%
DNF_1	150	150	150	150	150	150	150	150	150
DNF_2	121	3	0	1243	4	0	2293	4	0
DNF_3	9	9	0	1108	80	0	9325	223	1
DNF_4	14	14	3	133	133	32	2464	1089	108
DNF_5	7	7	6	200	200	109	2734	2734	1304
DNF_6	6	–	–	178	178	177	3486	3486	3017
DNF_7	–	–	–	54	54	54	2662	2662	2662
DNF_8	–	–	–	–	–	–	89	89	89
total	307	189	162	3066	799	522	23203	10437	7331
max_vars	7	6	6	7	6	6	7	6	6

δ	80%								
η	5%			10%			15%		
τ	δ	50%	30%	δ	50%	30%	δ	50%	30%
DNF_1	7	7	7	7	7	7	7	7	7
DNF_2	4	1	–	92	1	0	227	1	0
DNF_3	3	3	–	46	24	0	356	55	0
DNF_4	2	2	–	42	42	0	182	145	0
DNF_5	4	4	–	9	9	1	189	189	17
DNF_6	–	–	–	2	2	1	140	140	25
DNF_7	–	–	–	–	–	–	73	73	73
DNF_8	–	–	–	–	–	–	9	9	9
total	20	17	7	199	85	9	1183	619	131
max_vars	2	2	2	3	2	2	3	2	2

δ	90%								
η	5%			10%			15%		
τ	δ	50%	30%	δ	50%	30%	δ	50%	30%
DNF_1	1	1	1	1	1	1	1	1	1
DNF_2	2	0	–	2	0	–	5	0	–
DNF_3	1	1	–	7	1	–	51	1	–
DNF_4	1	1	–	8	8	–	27	24	–
DNF_5	–	–	–	–	–	–	25	25	–
DNF_6	–	–	–	–	–	–	8	8	–
total	5	3	1	19	10	1	117	59	1
max_vars	1	1	1	1	1	1	1	1	1

Fig. 7. The number of extracted coverings from MRSA data by introducing the maximum monomial support τ.

monomial support reduces the number of extracted coverings consisting of a few monotone monomials from MRSA data.

4.3 Anaerobes data

In this section, we apply *dnf_cover* to Anaerobes data with 1082 records[1] larger than MRSA data with 118 records. Figure 8 describes the number of extracted coverings from Anaerobes data by *dnf_cover*. Note that, under the minimum support 70% or the maximum overlap 15%, we cannot extract coverings from Anaerobes data.

δ	80%		90%	
η	5%	10%	5%	10%
DNF_1	7	7	1	1
DNF_2	9	15	1	3
DNF_3	6	44	2	5
DNF_4	0	9	–	–
DNF_5	1	3	–	–
DNF_6	–	2	–	–
total	23	78	4	9
max_vars	3	3	1	1

Fig. 8. The number of extracted coverings from Anaerobes data.

Under the minimum support 80% and the maximum overlap 10%, 40 coverings contain the resistant to benzilpenicillins, anti-pseudomonas penicillins ($\texttt{PcAP} = \texttt{R}$), 1st generation cephems, 2nd generation cephems ($\texttt{Cep2} = \texttt{R}$), 3rd generation cephems ($\texttt{Cep3} = \texttt{R}$), lincomycins and macrolides ($\texttt{ML} = \texttt{R}$). In particular, 29 coverings are redundant for the sensitivity of antibiotics such as $(\texttt{PcB} = \texttt{R}) \vee (\texttt{PcB} = \texttt{S})$ and 11 coverings are nonredundant. A part of extracted nonredundant coverings is described in Figure 9.

On the other hand, Anaerobes consist of 13 species. In particular, we pay our attention to Bacteroides spp. data with 524 records, Fusobacterium spp. data with 154 records, Prevotella spp. data with 165 records and Streptococcus spp. data with 157 records, and refer them to `Bact`, `Fuso`, `Prev` and `Stre`, respectively. Then, the number and examples of extracted coverings from them by *dnf_cover* are described as Figure 10 and 11, respectively. Here, the maximum overlap is fixed to 10%.

With increasing the size of database under the same minimum support and maximum overlap, the number of extracted coverings by the algorithm *dnf_cover*

[1] The number of data is different from [8], because we simply extract data of which bacterium is Anaerobes from the original data, while data in [8] has been obtained by cleaning our data.

DNF	frequency
$(\texttt{Cep1} = \texttt{S})_{45.2} \vee (\texttt{PcB} = \texttt{R})_{52.0}$	87.5%
$(\texttt{Cep2} = \texttt{R})_{10.6} \vee (\texttt{year} = \texttt{95})_{42.4} \vee (\texttt{year} = \texttt{96})_{36.2}$	80.9%
$(\texttt{Cep2} = \texttt{S})_{73.9} \vee (\texttt{Cep3} = \texttt{R})_{14.7}$	81.5%
$(\texttt{LCM} = \texttt{R})_{31.0} \vee (\texttt{ML} = \texttt{S})_{59.9}$	84.1%
$(\texttt{LCM} = \texttt{S})_{57.6} \vee (\texttt{ML} = \texttt{R})_{29.2}$	81.7%

Fig. 9. A part of extracted nonredundant coverings from Anaerobes data under the minimum support 80% and the maximum overlap 10%.

data	Bact			Fuso			Prev			Stre		
records	524			154			165			157		
δ	70%	80%	90%	70%	80%	90%	70%	80%	90%	70%	80%	90%
DNF_1	24	7	7	239	44	8	87	15	3	5	1	1
DNF_2	116	22	5	789	283	10	525	33	5	34	2	1
DNF_3	287	43	13	1494	159	19	887	57	19	202	25	2
DNF_4	39	4	–	396	28	0	156	16	4	118	25	2
DNF_5	44	2	–	230	9	0	134	4	–	83	12	1
DNF_6	2	2	–	28	17	2	2	–	–	2	2	2
total	512	80	25	3176	540	39	1791	125	31	444	67	9
max_vars	4	3	3	7	5	3	6	4	2	2	1	1

Fig. 10. The number of extracted coverings from Bact, Fuso, Prev and Stre under the maximum overlap 10%.

data	*DNF*	frequency
Bact	$((\texttt{CBP} = \texttt{S}) \wedge (\texttt{CP} = \texttt{S}) \wedge (\texttt{PcB} = \texttt{R}) \wedge (\texttt{TC} = \texttt{S}))_{74.6} \vee (\texttt{ctr} = \texttt{2})_{12.8}$	77.7%
	$(\texttt{LCM} = \texttt{R})_{42.3} \vee (\texttt{ML} = \texttt{S})_{56.9}$	90.9%
	$(\texttt{LCM} = \texttt{S})_{50.2} \vee (\texttt{ML} = \texttt{R})_{36.3}$	83.9%
Fuso	$((\texttt{CBP} = \texttt{S}) \wedge (\texttt{CP} = \texttt{S}) \wedge (\texttt{Cep1} = \texttt{S}) \wedge (\texttt{Cep2} = \texttt{S}) \wedge (\texttt{PcAP} = \texttt{S})$ $\wedge (\texttt{PcB} = \texttt{S}))_{64.3} \vee (\texttt{PcB} = \texttt{R})_{16.2} \vee (\texttt{age} = \texttt{10s})_{12.3}$	84.4%
	$(\texttt{LCM} = \texttt{R})_{26.0} \vee (\texttt{ML} = \texttt{S})_{53.9}$	73.4%
Prev	$((\texttt{CBP} = \texttt{S}) \wedge (\texttt{CP} = \texttt{S}) \wedge (\texttt{Cep2} = \texttt{S}) \wedge (\texttt{Cep3} = \texttt{S}) \wedge (\texttt{LCM} = \texttt{S}))_{67.3}$ $\vee (\texttt{ML} = \texttt{R})_{10.9}$	75.8%
	$(\texttt{LCM} = \texttt{R})_{10.8} \vee ((\texttt{ML} = \texttt{S}) \wedge (\texttt{PcAP} = \texttt{S}))_{64.3}$	72.6%
Stre	$(\texttt{LCM} = \texttt{R})_{10.3} \vee (\texttt{ML} = \texttt{S})_{58.1} \vee (\texttt{age} = \texttt{10s})_{12.3}$	71.0%

Fig. 11. A part of extracted coverings from Bact, Fuso, Prev and Stre under the minimum support 70% and the maximum overlap 10%.

tends to decrease by Figure 5 and 8, while its tendency is not correct exactly by Figure 10. Furthermore, we pay our attention to the sensitivity of antibiotics in the extracted coverings (*cf.* Figure 11).

1. In extracted coverings from `Bact`, the resistant to benzilpenicillins, 1st generation cephems, 2nd generation cephems and 3rd generation cephems occurs frequently. Also the resistant to anti-pseudomonas penicillins occurs in 4 redundant coverings. Furthermore, the resistant to lincomycins and macrolides occurs in 4 coverings, in which 2 coverings are redundant and 2 coverings are described in Figure 11.
2. In extracted coverings from `Fuso`, the resistant to benzilpenicillins, 1st generation cephems and 3rd generation cephems occurs frequently. Also the resistant to macrolides occurs in 5 redundant coverings.
3. In extracted coverings from `Prev`, the resistant to lincomycins and macrolides occurs frequently. Also the resistant to 1st generation cephems occurs in 34 coverings and the resistant to benzilpenicillins occurs in 5 redundant coverings. Other sensitivity of antibiotics is susceptibility.
4. In extracted coverings from `Stre`, all of the sensitivity of antibiotics are susceptibility, except lincomycins.

Note here that the occurrence of the susceptible to carbapenems (`CBP` = `S`) in Figure 11 is implied by the fact that carbapenems take effect for Anaerobes. Also the above statement for `Stre` is implied by the fact that any drug takes effect for Streptococcus spp. Furthermore, the information that the age of patients is 10's is interesting, because our bacterial culture data mainly consist of information for older patients.

5 Conclusion

In this paper, we have extended monotone monomials as large itemsets in Apriori to monotone DNF formulas, and formulated the coverings as monotone DNF formulas by introducing two measure, the minimum support and the maximum overlap. Then, we have designed the algorithm *dnf_cover* to extract the coverings as monotone DNF formulas. Finally, we have given the empirical results by applying the *dnf_cover* to MRSA data and Anaerobes data. In particular, we have succeeded to extract some valuable coverings from a medical viewpoint.

It is one of the advantage of the algorithm *dnf_cover* that we give no upperbound of both the number of monotone monomials and the number of variables in monotone monomials in monotone DNF formulas. For example, by using k minimal multiple generalizations by Arimura *et al.* [5], we can design the algorithm to extract coverings as *monotone k-term DNF formulas*, where k is the upperbound of the number of monomials. Instead of them, we give the upperbound of the overlap (and the minimum and the maximum monomial supports), and reduce a search space for the extraction of monotone DNF formulas.

In this paper, we only implement the prototype to the algorithm *dnf_cover*, so it is a future work to improve the efficiency of the implementation. In particular,

Agrawal *et al.* [2, 3] have designed AprioriTid in order to decrease the number of accesses to a transaction database, so it is necessary to improve our *dnf_cover* according to AprioriTid.

Figure 5, 8 and 10 in Section 4 claim that the appropriate number of our bacterial culture data from which *dnf_cover* extracts many nonredundant coverings is less than about 500. This claim is concerned with 93 attributes of our data. It is a future work to clear the relationship between the number of attributes and such appropriate number of data for various databases.

From a medical viewpoint, it is interesting how coverings can be extracted from Staphylococci and Enterococci data, in particular, by the difference of samples between the blood and others. It is an important future work.

From a viewpoint of Algorithmic/Computational Learning Theory, it is well known that monotone DNF formulas are learnable with equivalence and membership queries [4]. It is a future work to analyze the relationship between the learnability and the extraction of monotone DNF formulas, and to incorporate the learning algorithm with *dnf_cover*.

Acknowledgment

The authors would thank to Kimiko Matsuoka in Osaka Prefectural General Hospital and Shigeki Yokoyama in Koden Industry Co., Ltd. for the valuable comments from a medical viewpoints in Section 4 and 5.

References

1. J.-M. Adamo: *Data mining for association rules and sequential patterns: Sequential and parallel algorithms*, Springer, 2001.
2. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. I. Verkamo: *Fast discovery of association rules*, in [7], 307–328.
3. R. Agrawal, R. Srikant: *Fast algorithms for mining association rules in large databases*, Proc. of 20th VLDB, 487–499, 1994.
4. D. Angluin: *Queries and concept learning*, Machine Learning **2**, 319–342, 1988.
5. H. Arimura, T. Shinohara, S. Otsuki: *Finding minimal generalizations for unions of pattern languages and its application to inductive inference from positive data*, Proc. 11th STACS, LNCS **775**, 649–660, 1994.
6. S. Džeroski, N. Lavrač (eds.): *Relational data mining*, Springer, 2001.
7. U. M. Fayyed, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (eds.): *Advances in Knowledge Discovery and Data Mining*, AAAI/MIT Press, 1996.
8. K. Matsuoka, M. Fukunami, S. Yokoyama, S. Ichiyama, M. Harao, T. Yamakawa, S. Tsumoto, K. Sugawara: *Study on the relationship of patients' diseases and the occurrence of Anaerobes by using data mining techniques*, Proc. International Congress of the Confederation of Anaerobes Societies **186** (1Xa-P2), 2000.
9. E. Suzuki: *Mining bacterial test data with scheduled discovery of exception rules*, in [10], 34–40.
10. E. Suzuki (ed.): Proc. International Workshop of KDD Challenge on Real-World Data (KDD Challenge 2000), 2000.

11. S. Tsumoto: *Guide to the bacteriological examination data set*, in [10], 8–12. Also available at `http://www.slab.dnj.ynu.ac.jp/challenge2000`.
12. C. Zhang, S. Zhang: *Association rule mining*, LNAI **2308**, 2002.