

6. ПРОДУКЦИОННИ МОДЕЛИ

6.1. Въведение.

При продукционните модели знанията се представят във вида

Условие \rightarrow Следствие

Пример: Ако времето е хубаво, Иван отива на риболов.

Продукциите отразяват релацията по най-близък за човека начин. Те са относително близки до логическите модели и дават възможност да се организират процедури за извод. В същото време продукционните модели не са така формализирани и ограничени както логическите модели.

Всяка продукция може да се разглежда като правило за извършване на определени действия.

6.2. Общ вид на продукция.

В най-обобщен вид една продукция изглежда по следния начин:

(i) Q; P; A \Rightarrow B; N;

Първата съставка **i** е име на продукцията. То я отделя от останалите продукции.

Името може да бъде:

- число (например пореден номер в базата знания);
- дума или набор от думи, които отразяват същността на продукцията. (например “търсене на път в дървовиден граф”).

Втората съставка **Q** е област за приложение на продукцията.

Знанията се групират по сфери на приложение. Това дава възможност да се осъществи по-ефективно търсене в базата от знания. За да се осъществи по-ефективно търсене, се използва областта за приложение на продукциите.

Третата съставка **P** е условие за приложимост на продукцията.

Често това условие е в предикатна форма. Продукцията може да се използва, само ако това условие е истинно.

Четвъртата съставка **A \Rightarrow B** е ядро на продукцията. Знакът \Rightarrow се интерпретира като “следва”.

Ядрото може да има различни форми. Най-често;
АКО A ТО B (if A then B).

Възможни са и други варианти, например:

АКО A ТО B₁, в противен случай B₂ (if A then B₁ else B₂)

Ядрото може да се интерпретира по различен начин.

Например:

-логическо интерпретиране – ако условието е истинно, то истинно е и B; ако условието не е истинно, то за истинността на B нищо не може да се каже.

-A е условие за изпълнението на действието B.

Примери за продукции.

(Път с максимално тегло в граф) (i); Графът е дървовиден и всеки възел е означен с тегло (Q); Графът е бинарен(P);

Ако $x \geq y$, избира се x, в противен случай се избира y. Да се повтори ядрото, ако избраният възел не е краен(ядро).

(Спускане до най-дълбоката площадка в пещера (i)); Всички разклонения водят надолу (Q); На всяка площадка има не повече от две разклонения (P); Ако $x \geq y$, избира се x, в противен случай се избира y; Ако достигнатата площадка не е последна, да се повтори ядрото(ядро).

Разгледаните до тук ядра могат да се нарекат прости. Едно сложно ядро може да има следния вид:

Ако A₁, A₂, ..., A_n, то (B₁, B₂, ..., B_n)

И от двете страни смисълът е конюнкция.

Последната пета съставка **N** е постусловие на продукцията.

Постусловието се активира след като продукцията се изпълни.

Постусловията описват действия и процедури, които трябва да следват изпълнението на продукцията.

6.3. Видове ядра на продукцията.

Ядрата могат да бъдат детерминирани или недетерминирани:

При детерминираните ядра ако условието е изпълнено, дясната част се изпълнява обезателно.

При недетерминираните ядра са възможни варианти.

Ако **A**, то е възможно **B**.

Възможността може да се изрази по различен начин, например чрез:

- **вероятност**

Ако **A**, то с вероятност **p** да се реализира **B**.

- **семантична оценка** за възможност

Ако **A**, то *с увереност* да се изпълни **B**.
почти сигурно и т.н.

Възможна е и алтернативност:

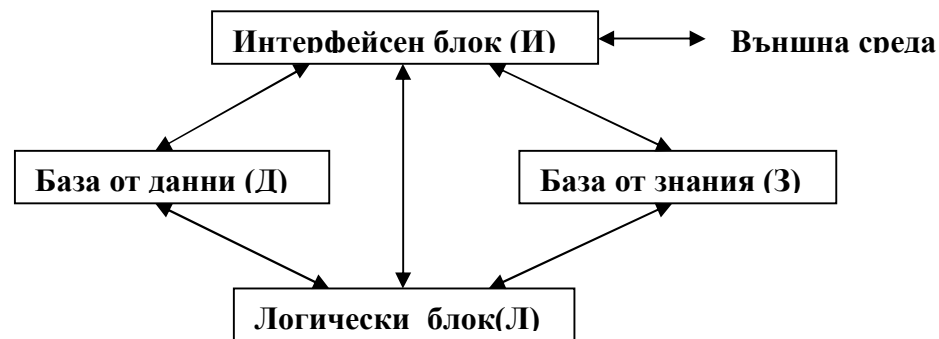
Ако **A**, то по-често **B₁**, а по-рядко **B₂**.

В такива случаи са необходими тегловни коефициенти за избор (вероятностна оценка, лингвистична оценка, експертни оценки).

Някои продукции могат да представляват прогнозиращи продукции. Например

Ако **A**, то с вероятност **p** може да се очаква **B**.

Източник и приемник на информацията за продукцията би могъл да бъде всеки от блоковете на следната схема.



Възможните варианти личат от следната таблица.

ВА	И	Д	З	Л
И		*		*
Д	*	*	*	*
З	*	*	*	*
Л	*		*	*

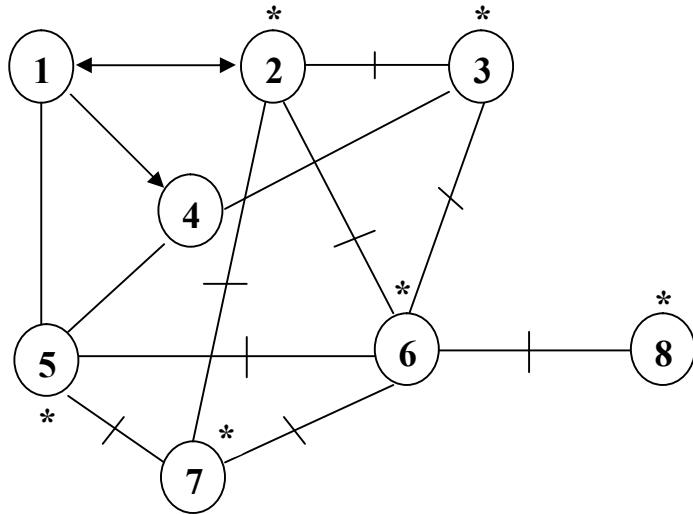
Записът $A_x \Rightarrow B_y$ означава, че информацията се взема от блок **x**, а резултатът от действието **B** се изпраща в блок **y**.

Например $A_D \Rightarrow B_D$.

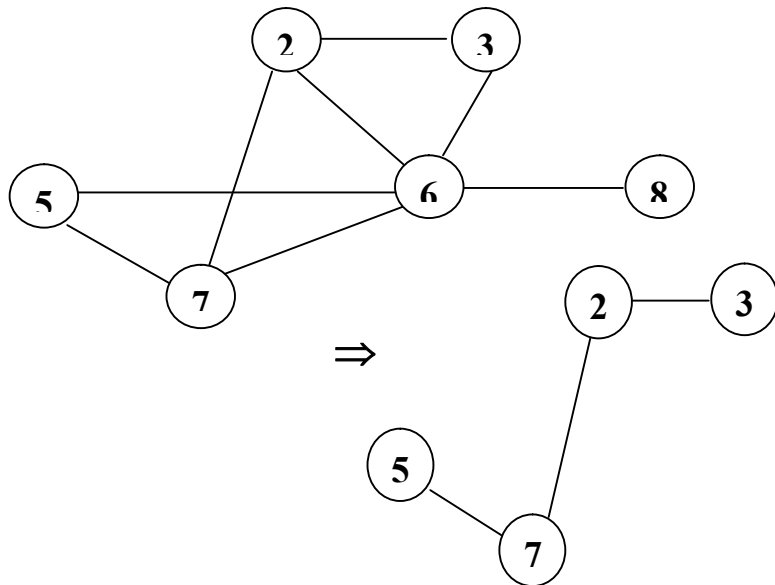
Най-често $A_Z \Rightarrow B_Z$ - замяна на един фрагмент от базата знания с друг фрагмент (получен чрез B_Z). За да се задейства тази продукция е необходимо в БД да бъде намерен фрагмента A_Z . (търсене по образец).

$A_D \Rightarrow B_Z$ - намиране на закономерности по емпирични данни.

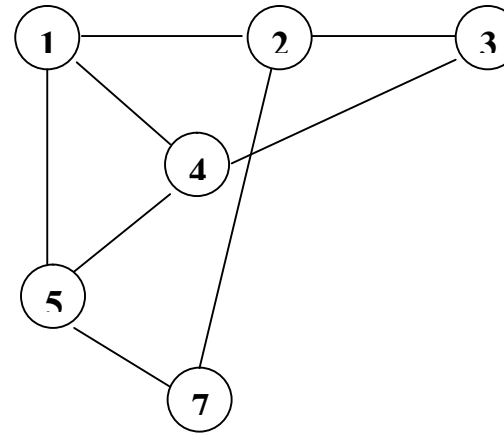
Пример: Пътна мрежа.



Селища 6 и 8 след построяване на язовир.
Търсене на фрагмент и замяна с нов фрагмент.



Резултатът има следния вид:



6.4. Основен обработващ цикъл

БД съдържа конкретни стойности за всички величини-характеристики на обектите, които са достъпни за всички продукции както при проверка на стойността, така и за нейното изменение.

Обработката следва следната схема:

- 1) Проверка за изпълнение на активиращите условия (левите страни) на всички продукции. Определя се множеството продукции, които могат да се изпълнят. Те се конкурират помежду си.
- 2) По определена стратегия се избира една от конкуриращите се продукции. (Всички останали продукции не се изпълняват).

3) Изпълнява се избраната продукция като с това се променя състоянието на базата от данни. Ако не е налице условието за край следва отново 1.

При изпълнението на първата стъпка от обработващия цикъл се открояват два момента:

- *обхождане на продукциите в базата знания и*
- *търсене на стойности в базата данни.*

От рационалното им организиране зависи ефективността на системата.

Стратегии за обхождане на продукции. В това отношение възможните варианти са много и ще споменем само два примера.

1) Използува се следната идея – най-често използваната продукция е най-полезна. За да работи успешно тази идея, необходимо продукциите да бъдат независими една от друга. Те се подреждат в базата знания по честотата на използване и всеки път се обхождат в този ред. Ако се добави и обратна връзка по “успешно прилагане” на продукциите се получава обучаваща се система, която се усъвършенствува в процеса на работата си.

2) Принцип на най-дългото условие. Този принцип се основава на “здравия смисъл” - частните случаи са по-полезни, тъй като отчитат по-голям брой условия. За да работи този принцип, необходимо е подреждане на продукциите в реда от по-частни към по-обща.

Търсенето в базата данни отнема значително време, поради което е необходимо да се предприемат мерки за рационалното му организиране. Ето някои от използваните варианти за организиране на този процес:

- последователно търсене; има най-ниска ефективност;
- търсене чрез индексване на базата данни;
- разбиване на подбази от данни и въвеждане на правила за избор на подбаза.
- разбиване на продукциите на групи по област на приложение и формиране на база данни за всяка подгрупа;

- вътрешно структуриране на БД и проверка само на онези правила, в чиито условия съществуват обекти, променени в предходната итерация и т.н.

За оптимизиране изпълнението на първата стъпка могат да се използват следните идеи:

* Много продукции имат общи условия или части от условия. Следователно проверката на тези условия може да се извърши само един път.

* При изпълнение на продукции само малка част от данните се променят. Следователно може да се проверява само променената част от данните, тъй като резултатите от проверката на другите данни са както при предишната итерация.

При изпълнението на втората стъпка от обработващия цикъл се използват разнообразни **стратегии за избор на продукция**. Най-често стратегиите за избор на продукция са известни предварително и са твърдо заложили. Използуват се следните стратегии за избор:

- **по номер**-избира се продукцията, която има най-малък номер (например пореден номер)

- **по “възраст” на данните** - избира се продукцията, чиито активиращи данни са били най-скоро променени, или които тя ще промени.

- **по успешно участие** - избира се продукцията, за която водената статистика показва, че най-често е използвана при довел до успех търсене на решение.

За изпълнението на третата стъпка ще споменем само, че възможността за **паралелно изпълнение** на всички активирани продукции може да доведе до случай, когато две от тях изменят едни и същи фрагменти от БД по различен начин. Необходимо е такива конфликти да се откриват и да се решават, което може да намали ефективността от паралелното изпълнение в значителна степен. Когато две или няколко продукции изменят едни и същи

даннипо различен начин, те образуват конфликтно множество продукции. При избор на продукция от конфликтно множество продукции може да се постъпи по някой от следните начини:

- избира се първата срещната продукция;
- избира се продукцията, която при този извод не е използвана.

Конфликтните множества от продукции могат да се намалят чрез добавяне на нови условия. Но тогава много от нововъведените продукции могат да се окажат взаимнозависими

6.5. Управление на системата от продукции.

При управлението на система от продукции трябва да се отчитат следните възможности:

- липсват предписания за реда за изпълнение на продукциите;
- има изисквания към реда за изпълнение на продукциите; В този случай е необходима информация коя продукция след коя ще следва.
- смесен вариант; В този случай за някои продукции е определен ред за изпълнение. Като частен случай в постусловието се посочва името на продукцията, която трябва да се изпълни като следваща.

Продукциите могат да бъдат управлявани

- Централизирано чрез изградена система за управление на избора.
- Децентрализирано от текущото състояние на БД.

При работата със система от продукции се използват разнообразни техники.

Използване на метапродукции. Метапродукциите се изпълняват първи и при активирането си определят реда за изпълнение на останалите продукции.

Принцип на “дъската за обяви”. В оперативната памет се оформя специална област, наречена “дъска за обяви”.

Изпълняваните продукции четат “обявите” от черната дъската и записват резултатите от своята работа също върху нея.

Принцип на приоритетния избор. За целта въз основа на важността на продукциите за дадена предметна област за всяка от тях се задава приоритет. Този приоритет се нарича статичен приоритет. Подреждането на продукциите по важност не е лека работа и се предлага от експертите. За усъвършенствуване управлението на системата се създава механизъм за поддържане на динамични приоритети, които се изработват в процеса на работа на системата.

Управление от имената. За имената на продукциите е зададена система от правила (например граматика), която стеснява кръга на активираниите продукции, които трябва да се изпълняват.

Продукциите могат да бъдат изпълнявани в прав и в обратен ред.

Прав ред - проверяват се условията за А и се изпълнява В.

Обратен ред – за да се изпълни В, необходимо е да се получат условията за А.

Срещат се наименованията прави (Forward) и обратни (Backward) продукции.

В продукционните системи изводът може да бъде реализиран в два варианта - прав извод (forward chaining) и обратен извод (backward chaining), които се наричат още съответно:

- извод, управляван от данните (data-driving inference)
- извод, управляван от целта (goal-driving inference)

Нека повторим **правия извод:**

1. Търсят се продукции, които могат да се задействат при дадено състояние на БД .
2. Избира се една от тях.
3. Изпълнява се тази продукция и ако не е достигната до целта отново се преминава към 1.

Обратен извод:

1. Намират се продукциите, които имат за дясна част търсената цел.

2. Избира се една от тях и параметрите от лявата ѝ част, които не са намерени в базата данни се формират като нови цели (подцели).

3. Действа се така докато се стигне до удовлетворяване от БД на всички формирани подцели или се изясни, че това е невъзможно.

При неуспешен край може да следва връщане назад (backtracking) и избор на ново правило, чиято дясна част е търсената цел.

6.7. Представяне на неточни знания.

В повечето случаи експертите представят своите знания с различна степен на увереност. За изразяване на този факт се използват различни подходи.

Използуване на вероятности

Вероятността да се избере обекта А от множеството $S=\{A,B,C,D,\dots\}$ от обекти се бележи с $P(A)$.

Вероятността да се избере обекта В, ако преди това е бил избран обекта А се бележи с $P(B|A)$ и се нарича условна вероятност.

Пример.

В- човекът е болен от грип.

А- човекът има висока температура.

$P(A)$ – доколко хората вдигат висока температура.

$P(B)$ – доколко хората боледуват от грип.

$P(A|B)$ – доколко високата температура се дължи на заболяване от грип;

Шансът за събитието А - $O(A)$ - се изчислява по следната формула:

$$O(A) = \frac{P(A)}{1 - P(A)}$$

Условният шанс да се случи А при случило се В:

$$O(B | A) = \frac{P(A | B)}{P(A | \bar{B})} \cdot O(B)$$

$$\alpha S = \frac{O(B | A)}{O(B)} \quad \text{и} \quad \alpha N = \frac{O(B | \bar{A})}{O(B)}$$

Величината αS е наречен фактор за достатъчност и означава шансът да се появи В при изпълнение на А.

Величината αN е наречен фактор за необходимост и означава шансът да се появи В при неизпълнение на А. (ЕС – prospector)

Да приведем няколко примера за използване на вероятности съвместно с продукции.

If A and B and C then D(0.8)

Ако А, В и С са истинни, то D е изпълнено с вероятност 0.8

If A(0.7) and B(0.6) and C(1.0) then D(0.8)

Вероятността за изпълнението на D се получава по формулата:

$$\min(0.7, 0.6, 1.0) * 0.8 = 0.48 \text{ (ЕС MYSIN)}$$

Използуване на лингвистична променлива

Например при оценката на повредата на сгради след земетресения може да се състави следната лингвистична

променлива(отгоре е цифровата оценка, а отдолу съответната ѝ лингвистична стойност):



Използване на коефициент на увереност (CF)

Коефициентът на увереност CF е число в диапазона [-1,1]. Стойностите означават -1 невярно, 0-няма сведения за истинността, 1- вярно;

Коефициенти на увереност могат да се задават както за променливи, така и за изрази. Нека покажем някои формули за изчисляване на коефициента на увереност за логически изрази:

Логически израз	Формула за коефициент на увереност
If x then y	$q(y) = q(x)$
If not x then y	$q(y) = 1 - q(x)$
If x ₁ and x ₂ then y	$q(y) = \min(q(x_1), q(x_2))$
If x ₁ or x ₂ then y	$q(y) = \max(q(x_1), q(x_2))$

До прага CF = 0.2 правилата се пренебрегват, тъй като се смята, че не могат да се използват.

Ако няколко правила завършват с едно и също заключение

Правило 1: if A then D (CF1=0.9)

Правило 2. if B and C then D (CF2 =0.9)

При CFA = 0.6, CFB = 0.8 и CFC = 0.9 след изчисление се получава:

$$\text{За първото правило } CF1D = 0.6 * 0.9 = 0.54$$

$$\text{За второто правило } CF2D = \min(0.8, 0.9) * 0.7 = 0.56$$

Окончателно за едновременното действие на двете правила:

$$CFD = CF1D + CF2D - CF1D * CF2D = 0.54 + 0.56 - 0.54 * 0.56 \approx 0.8$$

6.7. Предимства и недостатъци на продукционните системи.

Към предимствата на продукционните системи могат да се отнесат:

- Те са удобен начин за представяне на човешките знания.
- Притежават естественост при представяне на експертно знание.
- Системата е модулна и с редки изключения отстраняването и добавянето на продукция не изменя останалите.
- Всяка продукция е самостоятелна, независимо от останалите знания. Създаването и актуализацията се извършват лесно. Върху БЗ могат да работят едновременно няколко души.
- Налице е ясна идея за реализиране на обяснения.
- Поддават се в повечето случаи на структуриране.
- Съчетават се много добре с мрежовите модели.
- Наличие на естествен паралелизъм и асинхронност, което е от значение за новите поколения компютри.
- Простота на системите, които редактират знанията, плюс възможността за създаване на програми, които извличат (чрез “разпитване”) знания от експертите.

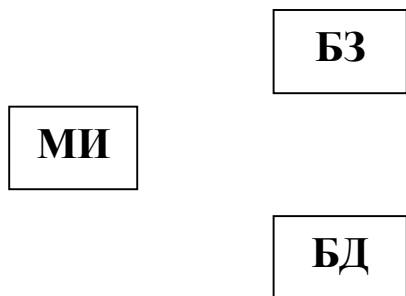
Към по-съществените недостатъци на продукционните системи могат да се причислят следните:

- При голям брой продукции проверката за непротиворечивост се затруднява;
- Съществени са и трудностите за проверка дали системата работи коректно (възможностите за избор на една от многото активирани продукции).
- В лявата част не може да се поставят дизюнкции

if $(a \vee b)$ then q . В този случай трябва да се запишат две независими продукции;

if a then q
if b then q

6.8. Пример към “продукционни системи”



Съдържание на БД:

Величини от тип Boolean

- $b_{1,i}$ има / няма
- $b_{2,i}$ е / не е
- $b_{3,i}$ притежава / не притежава
- $b_{4,i}$ прави / не прави
- $b_{5,i}$ е индивид / не е индивид

Многозначни величини (set)

$I_{1,i}$ (1,2,3,4,5)

$I_{2,i}$ (1,2,3,4,5,6,7)

Съдържание на БЗ:

1. $b_{1,1} \rightarrow b_{2,1}$
2. $b_{4,5} \rightarrow b_{2,1}$
3. $b_{4,1} \rightarrow b_{2,2}$
4. $b_{1,3} \wedge b_{1,4} \rightarrow b_{2,2}$
5. $b_{2,1} \wedge b_{1,5} \rightarrow b_{2,3}$
6. $b_{2,1} \wedge b_{4,6} \rightarrow b_{2,3}$
7. $b_{2,1} \wedge b_{2,2} \wedge I_{1,1}[1] \wedge I_{1,1}[3] \rightarrow b_{5,1}$

8. $b_{2,1} \wedge b_{2,2} \wedge I_{1,1}[4] \wedge I_{1,1}[1] \rightarrow b_{5,2}$
9. $b_{2,3} \wedge b_{1,6} \wedge b_{1,7} \wedge I_{1,1}[3] \rightarrow b_{5,3}$
10. $b_{2,3} \wedge I_{1,1}[4] \rightarrow b_{5,4}$
11. $b_{2,4} \wedge b_{1,6} \wedge b_{1,7} \wedge b_{4,3} \rightarrow b_{5,5}$
12. $b_{2,4} \wedge b_{4,3} \wedge I_{1,1}[2] \wedge b_{4,7} \rightarrow b_{5,6}$
13. $b_{2,4} \wedge b_{4,3} \rightarrow b_{5,7}$
14. $b_{1,8} \rightarrow b_{2,4}$
15. $b_{4,3} \wedge b_{4,4} \rightarrow b_{2,4}$

Действия с тези конструкции:

1) Какво следва от $b_{1,1} = true$

Преглед на БЗ.

$b_{2,1} = true$ от 1.

Системата отговаря $b_{2,1} = true$ и продължава.

Намиране на най-близко обкръжение

ако и $b_{1,5} \rightarrow b_{2,3}$ (5)

ако и $b_{4,6} \rightarrow b_{2,3}$ (6)

ако и $b_{2,2}$ и $I_{1,1}[1]$ и $I_{1,1}[3] true \rightarrow b_{5,1}$ (7)

Намиране на по-отдалечено обкръжение

ако и $b_{2,2}$ и $I_{1,1}[4]$ и $I_{1,1}[1] true \rightarrow b_{5,2}$

Избира най-близкото обкръжение и задава въпрос:

Въпрос на системата:

$? b_{i,5} = true$

При отговор да
Системата отговаря $b_{2,3} = true$;
системата със
($b_{1,1}, b_{2,1}, b_{1,5}, b_{2,3}$)

При отговор не
следва въпрос от
системата
 $? b_{4,6} = true$

търси ново обкръжение

Не пита за $b_{4,6}$,
защото не води
до нов резултат.
 $b_{2,3}$

(Не може да се приеме, че и $b_{4,6}=true!$ Внимание с импликацията)

- 7. $b_{2,2}$ и $I_{1,1}[1]$ и $I_{1,1}[3] \rightarrow b_{5,1}$
- 8. $b_{2,2}$ и $I_{1,1}[4]$ и $I_{1,1}[1] \rightarrow b_{5,2}$
- 10. $I_{1,1}[4] \rightarrow b_{5,4}$

Въпрос $?I_{1,1}[4] = true$

Започва с нещо, което е само в дясната част.

Интерпретация 1.

$b_{1,1}$ – козина	$b_{4,1}$ – яде месо	$b_{4,6}$ – преживя
$b_{1,3}$ – остри зъби	$b_{4,2}$ – яде трева	$b_{4,7}$ – плува
$b_{1,4}$ – остри нокти	$b_{4,3}$ – лети	
$b_{1,6}$ – дълга шия	$b_{4,4}$ – снася яйца	
$b_{1,7}$ – дълги крака	$b_{4,5}$ – дава мляко	

$I_{1,1}$ – оцветеност (светлокафяво, черно-бяло, черни ивици)
Подклас

$b_{2,1}$ – бозайник	$b_{5,1}$ – леопард
$b_{2,2}$ – хищник	$b_{5,2}$ – тигър
$b_{2,3}$ – копитно	$b_{5,3}$ – жираф
$b_{2,4}$ – птица	$b_{5,4}$ – забра

$b_{5,5}$ – щраус
 $b_{5,6}$ – пингвин
 $b_{5,7}$ – албатрос

$b_{1,i}$ – има / няма
 $b_{2,i}$ – е / не е
 $b_{3,i}$ – притежава / не притежава
 $b_{4,i}$ – прави / не прави

$I_{1,i}$ – оцветеност $I_{1,1}$ (бяло, черно, кафяво)
 $I_{1,2}$ (светлокафяво, тъмнокафяво, на ивици, кафяво)
 $I_{2,i}$ – покритие (козина, перушина, люспи, гола кожа)
 $I_{3,i}$ –

Интерпретация 2

$b_{1,1}$ – висока температура	$b_{3,1}$ – възпалени сливици
$b_{1,3}$ – главоболие	$b_{3,2}$ – апендикс
$b_{1,4}$ – световъртеж	$b_{3,3}$ – херния
$b_{1,5}$ – петна по кожата	
$b_{1,6}$ –	$b_{4,1}$ – повръща
$b_{1,7}$ –	$b_{4,2}$ –
$b_{1,8}$ –	
$b_{2,1}$ – заболяване	$b_{5,1}$ – изследване на
$b_{2,2}$ – инфекция	$b_{5,2}$ – да се приеме в болница
$b_{2,3}$ – зараза	$b_{5,3}$
$b_{2,4}$ –	$b_{5,4}$
	$b_{1,5}$
	$b_{1,6}$
	$b_{1,7}$
$I_{1,2}$ – възраст (дете, младеж, възрастен, напреднала възраст)	
$I_{1,1}$ – боледувал от (шарка,, скарлатина,)	