

INFERENCE ENGINE FOR DEVICES OF COLON CLASSIFICATION IN AI BASED AUTOMATED CLASSIFICATION SYSTEM

Pijushkanti Panigrahi

Reader & Head
Department of Library & Information Science
University of Calcutta
87/1 College Street
Kolkata – 700 073
West Bengal
Email : panigrahipk@yahoo.com

A. R. D. Prasad

Associate Professor
Documentation Research and Training Centre
Indian Statistical Institute
R V College - P. O
Bangalore - 560 059
Karnataka
E-mail : ardprasad@hotmail.com,
ardp@drtc.isibang.ac.in, ard@isibang.ac.in

This paper presents the use of various devices enunciated in Colon Classification (CC) to sharpen any isolate idea, especially when a new subject enters into the universe of subjects. Discusses their helpfulness in the automatic classification system, the Viswamitra, designed using Artificial Intelligence techniques with the help of PROLOG language. Discusses the procedure of building numbers for compound isolates using superimposition devices, alphabetical devices, geographic devices etc. Some suggestions are also made to rethink about the rule for alphabetical devices. For example, the system suggests to pick up the first three letters of a single worded term, first two letters from each word of a binomial term and the first letter from each word of a multinomial term to build isolate number using alphabetical device. Suggests some modifications in Colon Classification so that the concept of Analytico Synthetic Classification is used elaborately for information retrieval in the age of ontology.

INTRODUCTION

"The universe of information is ever-growing", is the fifth normative principle as enunciated by G Bhattacharyya in the light of Ranganathan's Five Laws of Library Science. The growth of information has paved the way for information explosion through generation of new scientific information by research, information analysis and consolidation and by other analogous activities. But the growth of information is responsible for barriers of large number including the phenomenon of small piece information lying buried in a vast mass of information preventing the easy access to the right information by right users [1]. This may result in unutilization of information

unless they are properly processed and arranged. To process and organize information, a classifier even finds difficulties as isolate numbers are not assigned in the enumerative classification schemes for newly generated ideas. Analytico synthetic classification schemes are right solution in these situations.

COLON CLASSIFICATION: AN OVERVIEW

An Enumerative Classification scheme tries to enumerate classification numbers for each and every subject (both Basic and Compound subjects) known to the designer and also it suggests numbers for some anticipated subjects which may come up in the future [2]. The disadvantages of enumerative schemes are, firstly, they provide a lengthy schedule of subjects. Secondly, the enumerated class numbers are built of a succession of semantically rich digits, not broken into meaningful facts by semantically poor digits. Thirdly, accommodating new subjects or thought contents of documents which are not enumerated, becomes difficult in the filiiary position vis-a-vis existing subjects [3]. Disadvantages of enumerative and almost enumerative schemes are eliminated by introducing the 'facet and isolate'* concept. This necessitates the conceiving of Faceted Classification scheme. It was Ranganathan who felt that classification schemes must not be enumerative in nature and he was trying to design a schedule consisting of

* Facet is a generic term used to denote any component - be it a basic subject (i.e. basic facet) or an isolate idea (i.e. isolate facet) - of a compound subject, and also its respective ranked forms, terms, and numbers. An isolate is any idea or idea complex to form a component of a subject but not by itself fit to be deemed to be a subject. Broadly isolates are two kinds - Special Isolates and Common Isolates.

'Schedule of Basic Subjects', 'Schedules of Common Isolates' which should be supported by 'Schedules of Special Isolates' for each specific basic subject. The class number for compound subjects can be built by accumulating numbers assigned in the different schedules of the basic subjects, common isolates and special isolates. The constructed class numbers will have connecting digits of a species different from the semantically rich digits used in the schedules of basic facets and the isolate facets. Thus the Colon Classification (CC) is devised by the great Indian library scientist S R Ranganathan.

In Freely Faceted Classification scheme, a compound basic subject is first analysed into its facets in the idea plane and later synthesized in the verbal plane and translated into the notational plane. For this reason, this scheme is also called as Analytico-Synthetic Classification. In this case, no fixed or rigid facet formula is used. At first, all facets are identified by proper analysis of the compound subject. This is done in the Idea Plane. Then one has to find out the facet terms and determine the sequence of those facets according to Principles and Postulates followed in the verbal plane. With the help of different schedules the facet terms are translated into notation i.e. Facet Numbers in the Notational Plane. Lastly, the facet numbers are synthesized into a class number using appropriate connecting digits. Hence each compound subject has its own right to determine its facet terms, facet sequence, facet number and class number etc. But every thing is guided in accordance with some basic principles and postulates applicable to any compound subject which is the main feature of an Analytico-Synthetic Classification, for example Colon Classification 7th Edition (CC7) [4].

Advantages of such schemes are their independent nature in building class numbers and they do not depend strictly either on enumerated classification number like enumerative schemes or on any rigid facet formula to be followed like rigidly faceted schemes. The three planes (Idea plane, Verbal Plane and Notational plane) will help in analyzing subject and in its synthesization for building a classification number. Various devices enunciated by S R Ranganathan for Colon Classification are

helpful to specify any isolate, especially when a new subject enters into the universe of subjects. These devices are essential in the formation of new isolates or sharpening of already existing ideas contained in the schedule and forming the numbers coextensive with them. Viswamitra, an automatic classification system developed for Colon Classification 7th edition (CC7) based on artificial intelligence (AI) techniques, is able to build classification numbers considering devices and generating numbers for them [5].

INFERENCE ENGINES FOR DEVICES

In the context of AI field, an inference engine is basically the code at the core of the system, which derives recommendations from the knowledge base and problem-specific data in working storage [6]. An inference engine technically encodes the algorithms used to derive the inference process in an expert system [7]. Thus, the inference engine for devices is the collection of rules written in Prolog language which are developed in the light of principles, postulates and canons for CC as enunciated by Ranganathan. On the other hand, the devices are used to sharpen or express the thought content of the document more distinctly by

- *forming an isolate number against a new idea which is not included in the schedule; or*
- *subdividing further an existing isolate idea to get its more distinct representation, which appears, through 'kind-of' or 'part-of' relationship with the existing one.*

Ranganathan suggested some devices like Superimposition Device for Compound Isolates, Alphabetical Device, Geographical Device, Subject Device, Chronological Device, Enumeration Device, Facet Device, Phase Device and Classic Device for Colon Classification [8]. Many of these are largely used in CC7.

Irrespective of individuals who are responsible for building class numbers of documents, right implementation of these devices will lead to the same isolate idea and isolate number (i.e. subjectivity of the individual may be avoided very easily). Again automating the classification

procedure helps in stereotype and mechanical implementation of devices. It gives the essence of automatic classification going with the analytical-synthetic nature of colon classification [9]. Inference engine for a few devices are demonstrated here, though the same for other devices can be developed in the same line of thinking easily using Prolog.

Superimposition Device to form Compound Isolate

In CC7, application of superimposition device has been shown in detail to form 'Compound Isolates'. This device facilitates in many ways like shortening a schedule by providing an enormous scope for building class number for those isolates, which are not in the schedule. When an isolate is not scheduled in a facet but can be regarded as the mutual denudation of two isolates, the compound isolate is constructed by using superimposition device. *An isolate in a schedule may be combined with another isolate in the same schedule. Such a combination of isolates is denoted by the term compound isolate* [10]. The device of forming compound isolate is called 'Superimposition Device for Compound Isolates'. Like other devices, the superimposition device is used to sharpen an isolate and to show its portion falling within another isolate of the same facet.

In building an isolate number using Superimposition Device, it is suggested to connect the isolate numbers by '-' (i.e. hyphen). With regard to the sequence of those isolates there are some guidelines.

- *The general convention, which became an important rule, is that the first occurring constituent isolate should form the first member of a superimposition device.*
- *If it is felt that the secondly occurred constituent isolate should be the first member to make the notation more intelligent, more helpful, then it must be the first member.*
- *If both the sequences of the constituent isolate got by above two rules give the intelligible isolate number and yield different meanings, the sequence should*

be determined by the meaning sought to be represented.

The superimposition device can be used for any basic subjects, special isolates or common isolates like space, time isolates. In the present work, the superimposition device has been implemented more effectively and successfully though the concerned predicates are confined to adopt the first rule only.

In the inference engine, the predicate '*super*' helps to form an isolate number using superimposition device [11]. This '*super*' predicate is called within the body of the '*compute*' rule

compute:-

```

[! recorded(out,Output,_) !],
 retrieveb(string, C, S1+N),
[! formula(C),
  super(C,S,S1,N1),
  not(var(S)),
  write(S),write(N1),
  write(Output,S),write(Output,N1) !],
 fail.

```

In the Medicine Schedule, the Superimposition Device is used widely for special isolates like personality, matter property, and energy schedule. For example, the isolates number for cells is 11 and blood is 35. But the number for '*Cells of Blood*' is not in the schedule. However, it can be built either as 11-35 or 35-11. Now the question is which one of the two is to be accepted, as both the numbers are found intelligible. Considering the convenience, the first is acceptable. Similarly '*Arteries of the brain*' is 37-72.

The following '*super*' rule is developed to form Compound Isolates using super imposition techniques

```

super(C+N,S,S1,N2):-
  recorded(done,C+N1,_),
  name(N,L),
  name(N1,L1),
  length(L1,Len),
  sublist(0,Len,L,L1), !,
  append(L1,L2,L),
  name(N2,L2),
  S=''.
super(C+N, S,S1,N):-
  recorded(done,C+N1,_),!,
  name(S,[45]).

```

super(C+N,S,S,N):-
recorda(done,C+N,_).

Please refer the Appendix included at the end of this paper to verify with out put of the system with regard to superimposition device.

Compound Time Isolates

Titles may contain two or more time isolate ideas, as the following [12].

Rainy new moon day ‘23-u
 New moon day in 1972 ‘N72-23
 Rainy new moon days in 1972‘N72-23-u

Or sometimes, the expressive title may contain a range of time.

1892-1972 AD ‘N72←M92.

However, variations in forming compound time isolates are observed in CC7. Their treatment in detail is studied through this research and will be discussed somewhere else. As this paper is mainly devoted to devices of CC7, only the use of superimposition concept is discussed here.

Any two isolates of time category are to be added as shown above. This type of compound time isolates are formed with the public time (specially year number) followed by ‘-’ (hyphen) and the ‘-’ should be followed by another feature time isolate. This sequence should be maintained always to form compound time isolate numbers. Two rules are framed under ‘compute’ predicate to build the compound time isolate [13].

```

compute:-
    !, recorded (out, Output, _)
    !, recorded (time, Term, _),
      write(Term),
      write(Output, Term) !,
    fail.
    
```

```

super(time+N, S, S1, N1):-
    name(N, L),
    append([96], L, L1),
    name(N2, L1),
    recorda(time, N2, Ref),
    N1=' ',
    S=' '.
    
```

The ‘*compute*’ predicate finds and writes the isolate number for Public time, then the ‘*super*’ predicate adds the indicator digit for time i.e. –‘– single inverted quote (ASCII value 96). On the other hand the following ‘*super*’ rule is to add other isolate numbers of the time category using superimposition concept i.e. using ‘-’ (hyphen, ASCII value 45).

```

super(time+N,S,S1,N1):-
    name(N,[H|T]),
    recorded(time,N2,Ref),
    erase(Ref),
    name(N,L),
    name(N2,L2), (
        append([96,48],L5,L2);
        append([96],L5,L2)
    ),
    append([96],L,L1),
    append(L1,[45],L3),
    append(L3,L5,L4),
    name(N5,L4),
    recorda(time,N5,R),
    S=' ',
    N1=' '.
    
```

As recommended in CC7, time isolates are categorized into two – ‘*time*’ to represent single and individual years, decades, centuries etc and ‘*time2*’ to represent featured time, lunar, solar time etc. Hence, two separate rules ‘*super*’ are incorporated. For building number of single and individual years, decades centuries etc in respect of time category ‘*time*’, the following ‘*super*’ is developed [14].

```

super(time+N,S,S1,N1):-
    name(N,L),
    forci(96,B),
    append([B],L,L1),
    name(N2,L1),
    recorda(time,N2,Ref),
    N1 = ' ',
    S=' '.
    
```

and to build isolate numbers for featured time, lunar, solar time of time category ‘*time2*’, the following ‘*super*’ has been used

```

super(time2+N,S,S1,N1):-
    recorda(time2,N,Ref),
    N1=' ', S=' '.
    
```

Actually, isolate number for civil time starts with roman capitals whose ASCII value is greater than that of numerals (which are used for lunar & solar

times) but lower than that of roman smalls (which are used for meteorological time divisions). CC7 suggests the reverse sequence of isolates in contrast of their appearance in the schedule i.e., First civil time then the other items of time may be added using super imposition device. The above super predicates add a 'zero' (ASCII 48) at the beginning of the civil time so that it appears always at the beginning. This is to fix the status sequence in the b-tree. But while computing and finally printing the isolate number, the system removes 'zero'.

Range of Time

When the expressive title contains a range of time, the '←' (backward arrow) is used as connecting digit. The end of an epoch is taken first, and then backward arrow followed by the beginning of that epoch is used to construct the isolate number. For example 1892-1972 AD is represented as 'N72←M92.

The following predicate *super(C, S, S1, N1)* is used to build the isolate number for a certain range of time. This is developed in the light of super imposition device [15].

```

super(time+N, S, S1, N1):-
    name(N, [H | T]),
        H>64,
        H<91,
        recorded(time, N2, Ref),
        erase(Ref),
        name(N, L),
        name(N2, L2),
        append([96], L5, L2),
        append([96], L, L1),
        append(L1, [27], L3),
        append(L3, L5, L4),
        name(N5, L4),
        recorda(time, N5, R),
        S='', N1=''.

```

The built-in predicates like *name*, *append*, *erase*, *recorda* are used to exploit their facilities. The *append* is used to add two list producing the third resultant one, *name* predicate converts the list into ASCII codes and vice versa, *recorda* is to store in

the b-tree, *erase* to remove the old reference from the memory [16]. In the Appendix included at the end of this paper, out put of the system with regard to superimposition device for time isolates are given.

Compound Space Isolates

The compound isolates for space may be formed, if necessary, by adding a host geographical area using a connecting digit '=' (i.e equal) sign [17].

Thus

```

Rivers of the World      1=zf7
Mountains of India      44=zm7
Valley of United Kingdom 56=zd4
and so on

```

As far as the expert system for the Superimposition Device is concerned, these have been taken into consideration. For this purpose, a separate knowledge base is developed for the schedule of environment division following the Prolog syntax [18]. As per the instruction of the schedule, the isolate number for geographical area and the special isolates are connected by '=' (equal sign) i.e. ASCII value 61. This rule is adapted in the inference engine using the following *super(space+N, S, S1, N)* predicate, where more or less the same programming techniques that are used for super-imposition device are adopted.

```

super(space+N, S, S1, N):-
    recorded(done, space+N1, _),!,
    name(S, [61]).

```

In addition to '=', the scope of adding different characters either '0' (zero), A, z or some other characters from some other schedule like environment divisions are observed. These individual geographical features may be connected using Alphabetical Device (AD) with a '0' (zero) preceding the AD number. Inference engine for Alphabetical Device is discussed in the next section. The following super rules are meant for compound space isolates.

```

super(space+N, S,S,N1):-
    name(N,[65|T]),           %A
    name(N1,[49,61,65|T]).   %1=A
super(space+N,S,S1,N):-
    name(N,[122|T]),         %z
    name(S,[46,49,61]), l.   %1=

```

In case of space A, 1=A, z, .1= are required to add as per the instruction in the schedule. The above segment of programming is used to do this. Please refer the Appendix included at the end of this paper to verify with out put of the system with regard to the use of devices for space schedule.

Alphabetical Device

The Alphabetical Device (AD) may be used where no other method of subdivision gives a better helpful sequence. The AD prescribes using the first or the first two or the first three etc., letters (all in caps) of the name of an entity, existential or conceptual, for the formation or the subdivision of an isolate. The number of letters used may be increased to the necessary extent [19].

Examples

Asia 4 India 44 West Bengal 4475
 Medinipur 4475=0M, which is a district of West Bengal

To differentiate this number from that of the district Malda the numbers may be assigned as

Malda	4475=0MA
Medinipur	4475=0ME

In case of binomial or multinomial terms i.e. a name with two or more words, the alphabetical device should be employed taking a suitable number of initial letters from each component word and connecting them by the indicator digit '+' with the next component [20]. For example, in the disease schedule of Medicine it is found that

4	Disease
42	Infection (i.e. disease caused by infection)
424	Bacteria (i.e. infected by bacteria)

For specific bacteria alphabetical device can be used e.g.

424 M+T Mycobacterium tuberculosis

(i.e. it represents the name of the specific bacteria which is responsible for infection)

In fact, the alphabetical device can be applied generally in respect of proper nouns and certain technical nomenclature, which are internationally nascent and have not yet been incorporated in the schedule. In addition, CC7 suggests the use of letters 'J' and 'P' instead of 'I' and 'O' respectively

[21]. This is to avoid the confusion that arises between the characters O and 0 (zero) and I and 1 (one). However, it is believed that confusion may arise when numbers are written manually, but if it is generated by computer then it will be clearly understandable by user, computer and /or even by barcode system.

To standardize the construction of an isolate number, Viswamitra always picks up the first three letters in a single worded term, the first two letters from each word of a binomial term and the only first letter from each word of a multinomial term. This small modification would not hurt the theoretical view rather it will help in building standardized classification numbers. The present work is carried out considering the following decisions.

- *For single worded term, the first three letters are taken as alphabetical device;*
- *For binomial (two worded terms), first two letters of each component word are picked up connecting them with the indicator digit '+'.*
- *For multinomial (three or more worded terms), only the first letter of each component word is picked up separating them with the indicator digit '+'.*

In the Inference Engine, use of Alphabetical Device is mainly categorized into two (i) Alphabetical Device for other than space schedule and (ii) Alphabetical Device for space schedule.

Alphabetical Device for other than Space Schedule

When the system fails to find the isolate number for any isolate idea from the knowledge base, the system should build that isolate number being guided by 'ifneeded(Object, Device)' predicate. The second variable (i.e., 'Device') can have any one of the following values depending on the requirement - 'alphadevice' or 'geographicdevice' or 'subjectdevice' or 'endalphadevice' etc. as suggested in Viswamitra [22]. For example:

If there is any instruction in the CC7 schedule like 'Use Alphabetical Device' or 'Further subdivided by

AD' etc., this instruction is converted into a Prolog fact as

ifneeded (Object, alphadevice).

Disease schedule of medicine does not include many diseases, which can be further subdivided using Alphabetical Device. As for *Encephalitis*, no number has been assigned in the schedule. But as it is a disease caused by virus infection, the number can be built as

4 disease 42 infection 423 virus 423ENC
encephalitis (By AD)

Similarly, *blood dysentery* is a type of dysentery but there is no number for it, though its broader term '*dysentery*' has got an isolate number like '474'. The number of *blood dysentery* may be formed as

474 dysentery

474BL+DY blood dysentery (By AD)

The knowledge base must contain information about where to use alphabetical devices. It can be represented as

cc7(encephalitis, part_of, virus).
ifneeded(encephalitis, alphadevice).

and

cc7(blood^dysentery, part_of, dysentery).
ifneeded(blood^dysentery, alphadevice).

In both cases, the inference engine first uses inheritance mechanism to get the number for the super concept i.e., virus or dysentery. Then it builds the actual number with the help of predicate '*alphadevice (Object, V)*'. For other than space category, the respective '*alphadevice*' predicate is

alphadevice(Object, V):-
has_value(Object, category, Cat),
Cat != space,
has_value(Object, part_of, Superconcept),
has_value(Superconcept, number, No),
alphanotspace(Object, No, V).

This '*alphadevice*' is again supported by another predicate '*alphanotspace (Object, No, V)*'. Three separate '*alphanotspace*' are written following the

above three consideration for single worded term, binomial and multinomial term [23]. This sequence is in the reverse order in the inference engine. That means, at first the system looks for multinomial, then binomial and lastly for single worded terms.

For Multinomial term,

alphanotspace(T1^T2^T3, No, V) :-
name(No, Ascii),
name(T1, [F_]), name(T2, [G_]), name(T3, [H_]),
F1 is F-32,
G1 is G-32,
H1 is H-32,
append(Ascii, [F1, 43, G1, 43, H1], V1),
name(V, V1).

At first, the system splits the term, and then collects the first letter of each term (i.e. through *F, G, H* variables) converts them to upper case by subtracting 32 from the respective variables (*F1, G1, H1* variables). Lastly, appends it with the isolate number of the super concept and uses '+' as a separator.

For Binomial term

alphanotspace(T1^T2, No, V) :-
name(No, Ascii),
name(T1, [F, G_]), name(T2, [H, I, _]),
F1 is F-32, G1 is G-32, H1 is H-32, I1 is I-32,
append(Ascii, [F1, G1, 43, H1, I1], V1),
name(V, V1).

In this case the '*alphanotspace*' predicate collects first two letters of each word (i.e. *F, G* and *H, I* variables), converts them to upper case by subtracting 32 (i.e. *F1, G1* and *H1, I1* variables), adds them with a separator '+' (i.e., ascii value is 43). Finally, this part is appended with the isolate number of the super concept. For example, *blood^dysentery*: *T1* is blood, *T2* is dysentery. Letter combination *bl, dy* (two first letters of words) are collected. They are transformed into capitals *BL* and *DY*, connected by '+' i.e., *BL+DY*, and appended to isolate number 474. The number for blood dysentery will be constructed as *474BL+DY*.

For Single worded term

alphanospace(T1, No, V):-

name(No, Ascii),
name(T1, [F, G, H | _]),
F1 is F-32, G1 is G-32, H1 is H-32,
append(Ascii, [F1, G1, H1], V1),
name(V, V1).

The system collects the first three letters of the term (i.e. *F, G, H* variables), transforms them into upper cases by subtracting 32 (i.e. *F1, G1, H1* variables), appends with the isolate number of the super concept. For example, encephalitis, *T1* is itself encephalitis. It collects first three letters i.e. enc, makes them upper case i.e. 'ENC'. Lastly appends it with the virus number i.e. 423. The number becomes 423ENC. Please refer the Appendix included at the end of this paper to verify with out put of the system with regard to the use of devices for the disease 'encephalitis'.

Alphabetical Device for Space Schedule

The individual geographical features may be connected using Alphabetical Device (AD) with a '0' (zero) preceding the AD number.

For example

The Thames river of UK 56=zf70T
 The Ganges or
 The Ganga river of India 44=zf70G

In case of using AD (Alphabetical Device), it is decided to pick up first three capital letters to make it more specific. Therefore,

The Thames river of UK 56=zf70THA
 The Ganges or
 The Ganga river of India 44=zf70GAN

In Prolog, the above rules are established using the following predicates.

endalphadevice (Object, V):-
cc7 (Object, number, No),
endalpha (Object, No, V).
endalphadevice(Object, V):-
cc7 (Object, part_of, Superconcept),

has_value(Superconcept, number, No),
endalpha(Object, No, V).

endalpha(Term^C, No, V):-
name(No, HH),
append(HH, [48], AS),
name(Term, [F, G, H | T]),
F1 is F-32,
G1 is G-32,
H1 is H-32,
append(AS, [F1, G1, H1], V1),
append(L1, V1, V2),
name(V, V2).

Geographical Device

The Geographical Device consists of using the appropriate Geographical Characteristics (i.e. geographical area, continent, country, etc) to (i) form an isolate number; or (ii) subdivide an isolate idea further.

The geographical characteristics, which warrant the use of GD, are place of origin, prevalence, habitation etc [24]. In the notational plane, the geographical numbers are used with an isolate number to sharpen an isolate idea. This device uses the Space schedule extensively. The Geographical Device is used in the different places of schedules of different basic subjects, ACI, PCI etc.

Geographical Device has not been used to a large extent in the medicine schedule. But to give an account of the overall view of using Geographical Device in future for other schedules (like language), its treatment by the inference engine is given. In the Space schedule, some special components may be used with the space constituents employing the indicator digit '=' to form compound space isolate. Different Empires of different countries have been agglomerated in one place with instruction to subdivide further using geographical device [25].

- A*Z By Empire (Divisions by Geographical Device)
- A52 Roman Empire (52 for Italy)
- A53 French Empires (53 for France)
- A56 British Empires (56 for United Kingdom)
- and so on

Though these special components are computed using geographical characteristics they cannot be used independently. They must be used with another space isolate to form compound space isolates.

1 = A56 The British Empire of the World

4 = A52 Roman empire in Asia and so on

In the Knowledge base, a fact *ifneeded* ($T^{\wedge}empire$, *geographicdevice*) is used along with two predicates *geographicdevice* and *geographic*.

ifneeded($T^{\wedge}empire$, *geographicdevice*).

cc7($T^{\wedge}empire$, *category*, *space*).

cc7(*roman* $^{\wedge}T$, *part_of*, *italy*).

cc7(*italy*, *part_of*, *europa*).

cc7(*italy*, *number*, '52').

cc7(*europa*, *category*, *space*).

cc7(*europa*, *number*, '5').

geographicdevice(*Object*, *V*):-

cc7(*Object*, *number*, *No*),

geographic(*Object*, *No*, *V*).

geographicdevice(*Object*, *V*):-

cc7(*Object*, *part_of*, *Superconcept*),

has_value(*Superconcept*, *number*, *No*),

geographic(*Object*, *No*, *V*).

geographic(*Term* $^{\wedge}C$, *No*, *V*):-

name(*No*, *N*),

append([61, 65], *N*, *V1*).

The system computes the isolate number by adding the isolate number of the country isolate to the symbol '=A'.

Similarly inference engine for other devices are also developed, but due to space limitations those are not discussed in detail.

CONCLUSION

Viswamitra, an attempt to design and develop an automatic classification system using Artificial Intelligence (AI) techniques, proves that Ranganathan's thinking was more computer compatible. Developing an automatic classification system stands as one of the major interesting research areas in Library and Information Science. The present study attempts to build an automatic classification system using AI tools and techniques. Basically, the approach is highly interdisciplinary one encompassing AI, NLP, ES using Logic Programming and Analytico Synthetic Classification [26]. Such study will not only build the classification numbers, but in addition, will help to extend its scope of its exhaustive use in different automated environment and even in the Web. The concept of

Analytico Synthetic Classification has already been applied to the Web. In an interesting article, Aimee Glassel [27] has shown how Ranganathan's idea of analysis and synthesis (Analytico Synthetic) of facets are used for information retrieval from the Internet. Also Steven Pollitt [28] studied possibilities of interactive information retrieval with the help of faceted classification concept of Ranganathan.

As the present work is based on the subject Medicine, devices with regard to this schedule are tested to generate classification number. In CC7, there are some other devices viz. Mnemonic Device, Speciator Device, Favored Category Device, Classic Device, Numerical Device. These have not been used to compute isolate number like other devices. Generally these devices are used to form an enumerated isolate number for the construction of the schedule. However, these can also be developed easily through the same line of thinking.

REFERENCES

- 1 BHATTACHARYYA (G). Information Science: a unified view through systems approach. 1978. Indian Association of Special Libraries and Information Centres; Calcutta (=Kolkata). p29.
- 2 RANGANATHAN (S R). Colon Classification Edition 7 (1971) : a preview. *Library Science with a Slant to Documentation*. 6, 3; 1969. Paper M. p193-242.
- 3 RANGANATHAN (S R). Colon Classification. 7th ed, (ed by M A Gopinath). 1989. Sarada Ranganathan Endowment for Library Science; Bangalore. 332p
- 4 *Ibid*
- 5 PANIGRAHI (Pijushkanti). An Artificial Intelligence approach towards Automatic Classification(Part 1). *IASLIC Bulletin*. 45, 3 ; 2000. p73-81.
- 6 MERRIT (Dennis). Building Expert Systems in Prolog. 1989. Springer-Verlag; New York. p2.
- 7 PATTERSON (D W). Introduction to Artificial Intelligence and Expert Systems. 1992. Prentice-Hall of India; New Delhi. 448p
- 8 RANGANATHAN (S R). Prolegomena to Library Classification. 3rd ed. (*Reprinted in 1989*). 1967. Sarada Ranganathan Endowment for Library Science; Bangalore. 640p.
- 9 PANIGRAHI (Pijushkanti). An Artificial Intelligence approach towards Automatic Classification(Part 2). *IASLIC Bulletin*. 45, 2; 2000. p105-118.
- 10 *Op. Cit* 3
- 11 CLOCKSIN (E J) and MELLISH (C S). Programming in Prolog. 3rd rev ed. 1993. Narosa Publishing House; New Delhi.281p.

- 12 *Op. Cit 3*
- 13 PRASAD (A R D) and PANIGRAHI (Pijushkanti). A Knowledge Representation Model for Time Isolates of Colon Classification (7th Edition). *SRELS Journal of Information Management*, 42; 3, 2005.p221-244.
- 14 *Ibid*
- 15 *Ibid*
- 16 SAINT-DIZIER (Patrick). An introduction to programming in PROLOG. 1990. Springer Verlag; New York. 184p
- 17 *Op. Cit 3*
- 18 *Op. Cit 11*
- 19 *Op. Cit 8*
- 20 *Ibid*
- 21 *Ibid*
- 22 *Op. Cit 11*
- 23 *Ibid*
- 24 SINGH (Sewa). A practical manual of colon classification edition 7. 1990. Jaico Publishing House; Bombay.
- 25 *Op. Cit 3*
- 26 WATERMAN (D A). A Guide to expert systems. 1986. Addison – Wesley; Massachusetts
- 27 GLASSEL (Aimee). Was Ranganathan a Yahoo?. *End User's Corner*. (<http://scout.cs.wisc.edu/toolkit/enduser/archive/1998/euc-9803.html> visited on Mar 17, 2002)
- 28 POLLITT (Steven). Interactive Information Retrieval Based on Faceted Classification using Views. (<http://www.hud.ac.uk/schools/clear/dorking.htm>, visited on March16, 2002).