

Dynamic stochastic models for indexes and thesauri, identification clouds, and information retrieval and storage

by

Michiel Hazewinkel
CWI
POBox 94079
1090GB Amsterdam
The Netherlands

Abstract. The first topic of this partial survey paper is that of the growth of adequate lists of key phrase terms for a given field of science or thesauri for such a field. A very rough ‘taking averages’ deterministic analysis predicts monotonic growth with saturation effects. A much more sophisticated realistic stochastic model confirms that.

The second, and possibly more important, concept in this paper is that of an identification cloud of a keyphrase (or of other things such as formulas or classification numbers). Very roughly this is (textual) context information that indicates whether a standard keyphrase is present, or, better, should be present, whether it is linguistically recognizable or not (or even totally absent). Identification clouds capture a certain amount of expert information for a given field. Applications include automatic keyphrase assignment and dialogue mediated information retrieval (as discussed in this paper). The problem arises how to generate (semi-)automatically identification clouds and a corresponding enriched weak thesaurus for a given field. A possible (updatable and adaptive) solution is described.

MSC2000: 68T35, 68U35, 91F20

Key words and key phrases. thesaurus, enriched weak thesaurus, growth of thesauri, identification cloud, information retrieval, information space, disambiguation, automatic indexing, thesaurus, standard keyphrase, dialogue search, neighborhood search, stochastic growth, dialogue mediated search, information storage, key phrase, automatic classification.

1. Introduction.

The first topic of this paper is concerned among others with the following question. Suppose one has made an index or thesaurus for a given (super)specialism like for instance discrete mathematics (understood as combinatorics) on the basis of a given corpus, like the two (leading?) journals ‘Discrete Mathematics’ and ‘Applied Discrete Mathematics’. How does one tell that the index made is more or less complete, i.e more or less good enough to describe the field in question. And, arising from that, are we really dealing with leading journals (as the publisher, in this case Elsevier, believes). As a matter of fact, indexes for the two journals named have been made, [21, 22] and a very preliminary analysis, [31], indicates that they go some way towards completeness.

One way to tackle this is to test the collection obtained against another corpus. However, such a second corpus may not be available. And if it were available one would like to use it also for key phrase extraction in order to obtain an index/thesaurus that is as complete as possible and the same problem comes back for the new index/thesaurus based on all material available.

Another way to try to deal with the question is to watch how the index/thesaurus grows as more and more material is processed. If, as one would intuitively expect, eventually saturation phenomena appear, that is a good indicator, that some sort of completeness has been reached. To deal with this not only qualitatively but also quantitatively, a dynamic stochastic model is needed, together with appropriate estimators. This is the first topic addressed in this paper.

The second topic deals with information retrieval and automatic indexing. These matters seem to have reached a certain plateau. As I have argued at some length elsewhere, see e.g. [14, 15, 19, 20] there is only so much that can be done with linguistic and statistical means only. To go beyond, it could be necessary to build in some expert knowledge into search engines and the like. This has led to the idea of identification clouds, which is one of the topics of this paper.

The same idea grew out of a rather different (though related) concern. It is known and widely acknowledged, that a thesaurus for a given field of inquiry is a very valuable something to have. However, a classical thesaurus according to ISO standard 2788, see [1], and various national and international multilingual standards, is not an easily incrementally updatable structure. Indeed, keeping up to date the well known thesaurus EMBASE, [6, 7], which is at the basis of Excerpta Medica, takes the full time efforts of four people.

This problem of semi-automatic incremental updating of a thesaurus has lead to the idea of an enriched weak thesaurus, [15, 20], and identification clouds are a central part of that kind of structure.

In the second part of this paper I try to give some idea of what ID clouds are and how they can be used. More applications can be found in the papers quoted. The idea has meanwhile evolved, largely because of the use of ID clouds in the EC project TRIAL SOLUTION, [39], and in this paper I also sketch the refinements that have emerged, and indicate some open problems that need to be solved if this approach is to be really useful.

This paper is an outgrowth of the lecture I gave on (some of) these matters at the IWAP 2002 meeting in Caracas, Venezuela, January 2002. I thank the organizers of that meeting for that opportunity.

2. A first preliminary model for the growth of indexes.

The problem considered in this section is how a global index, a list of terms supposed to describe a given field of enquiry, evolves as indexing proceeds and, simultaneously, the field develops (at a far from trivial pace). The questions arises how does such an index evolve chronologically (assuming, for simplicity, that the indexing is also done chronologically), and, most important, how does one judge on the basis of these data whether the index generated is adequate for the field in question or not.

Here is a very simple (and naive) stochastic model for this situation and a preliminary (deterministic) analysis of it. At starting time (time zero) there is an (unknown) collection, $K(0)$, of key phrases that is adequate for the field in question. In addition there is an infinite universe of potential terms that can be dreamed up by authors and others of new (important) key phrases. Thus, from the point of view of indexing and thesauri the field grows as:

$$K(t+1) = K(t) \cup B(t)$$

where the union is disjoint and $B(t)$ is the collection of new terms generated in period t . These are not yet known (i.e. identified/recognized), but they do exist in one form or another in the corpus as it exists at time t .

Now let indexing start. At time zero no terms have been identified. Let $X(t)$ stand for the set of terms recognized (found) at time t , $X(t) \subset K(t)$. Hence $X(0) = \emptyset$. A generalization would be that one starts with an existing thesaurus and tries to bring it up-to-date; then $X(0)$ is a

known subset of $K(0)$.

The indexing proceeds as follows. At time t a set of terms $S(t)$ is selected (found, recognized) and added to $X(t)$. This set $S(t)$ consists of two parts, $S(t) = A(t) \cup C(t)$, $A(t) \subset K(t)$, $C(t) \subset B(t)$, $A(t) \cap C(t) = \emptyset$. Thus

$$X(t+1) = X(t) \cup S(t) \subset K(t+1)$$

As a rule, of course, part of $A(t)$ is already in $X(t)$. The main problem is to have criteria or estimates to decide whether eventually $X(t)$ exhausts $K(t)$ or, $K(t - \tau)$ for a suitable delay τ , or not. For instance in the form

$$y(t) = \frac{x(t)}{k(t)} \rightarrow 1, \text{ as } t \rightarrow \infty$$

where $x(t)$ is the cardinality of $X(t)$ and similarly for $k(t)$. The (only) basic observable is $S(t)$ and deriving from that $X(t)$.

Let us do some rather crude average reasoning. First, let us assume linear growth of the field of science in question:

$$k(t) = k(0) + tv$$

for some constant v . Also on average u terms are selected (per period) with a fraction $\frac{x(t)}{k(t)}$ coming from known stuff, and a fraction $\frac{k(t) - x(t)}{k(t)}$ new terms. There results a recursion equation for $x(t)$:

$$x(t+1) = x(t) + u\left(1 - \frac{x(t)}{k(t)}\right)$$

Let $y(t) = x(t)/k(t)$ be the fraction of terms covered by the thesaurus at this time. Then

$$y(t+1) - y(t) = \frac{u}{k(t+1)} - \frac{(u+v)y(t)}{k(t+1)}$$

Assume that the differential equation

$$y' = \frac{u}{k(t+1)} - \frac{(u+v)y(t)}{k(t+1)}$$

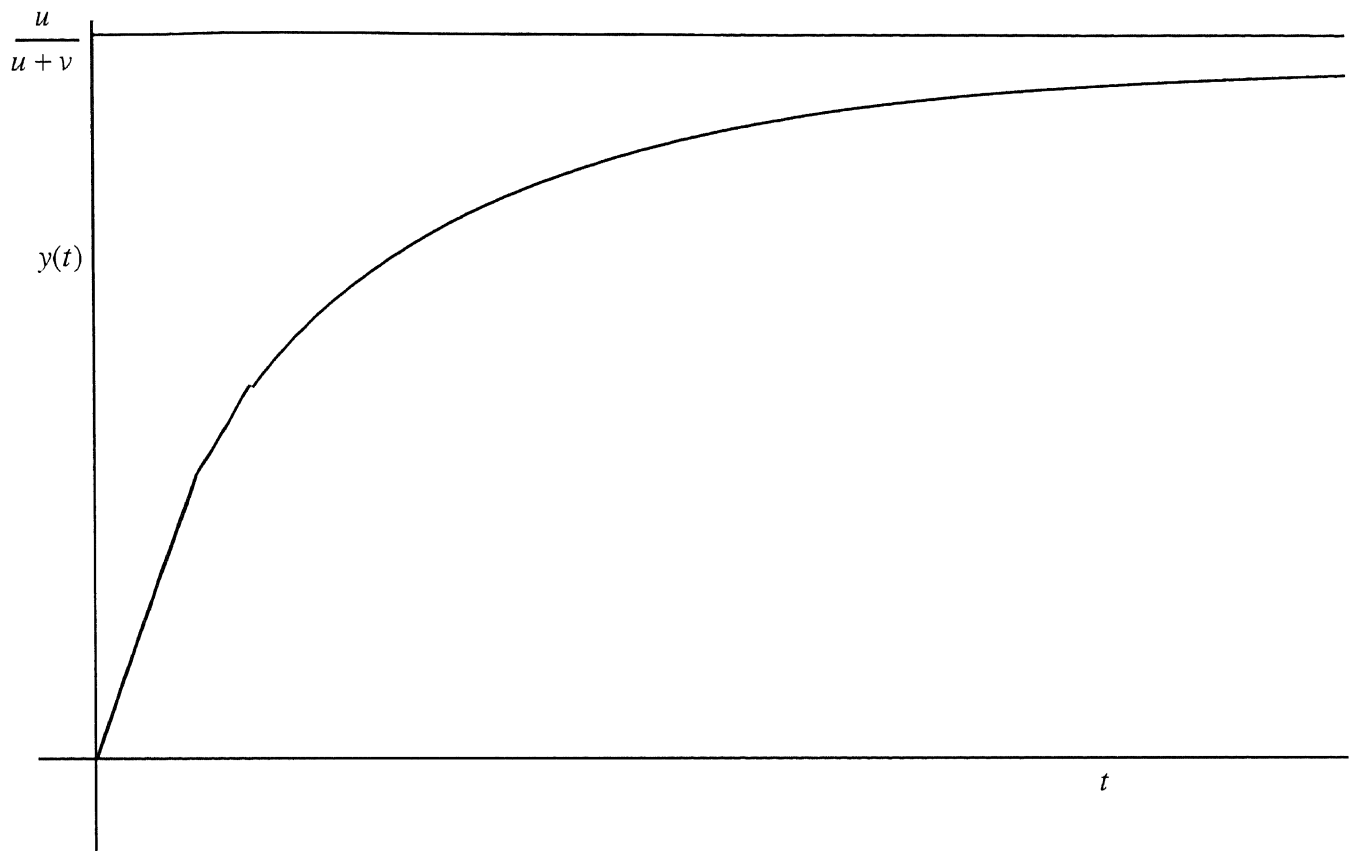
approximates the difference equation above well enough (which is certainly the case). This differential equation is actually explicitly solvable and the solution is:

$$y(t) = \frac{u}{u+v} - \frac{u(k+v)^{1+(u/v)}}{(u+v)(k+(t+1)v)^{1+(u/v)}}$$

where $k = k(0)$. So

$$\lim_{t \rightarrow \infty} y(t) = \frac{u}{u + v} \quad (2.1)$$

and $y(t)$ grows monotonically from 0 to the asymptotic limit value $u/u + v$.



In particular the recognized fraction of relevant (latent) index terms does not approach one as long as the field keeps growing, and it grows slowly (compared to the indexing rate) once one gets very close to the asymptotic limit. Note also that the saturation phenomenon alluded to in the introduction does indeed occur.

Of course this is quite primitive. Frequently, replacing stochastic phenomena with averages (in a nonlinear case) does not work. So a more sophisticated analysis of this kind of stochastic processes — apparently a new kind — is needed. This is described in the next section.

3. A dynamic stochastic model for the growth of indexes.

Using the same notations as above the basic assumptions of the model are as follows.

- The $x(t)$, the cardinalities of the sets of key phrases identified up to and including time t , form a random Poisson process. That is, the increments $\Delta x(t) = x(t) - x(t-1)$ are independent random variables with a Poisson distribution P_{λ_t} . For simplicity $x(0)$ is assumed to be a deterministic quantity. Let $n(t) = \mathbf{E}x(t)$, then $\lambda_t = \Delta n(t)$.

- The key phrases are numbered consecutively as they appear in time. A key phrase $w_k \in K(t)$ at the time of its emergence has attached to it a random weight W_k that reflects its relevance (= importance) at that time. The W_k are supposed to be iid positive random variables

with a distribution function F independent of the sequence $x(t)$, and $\mathbf{E}W_k = 1$.

• As before let $S(t)$ be the set of key phrases that were observed at time t and let $A_{k,t} = \{w_k \in S(t)\}$. The probabilities of the random events $A_{k,t}$ depend on the random weights W_k and the history so far, I_t , of the system considered. Assume that for fixed W_k and I_t , the events $A_{k,t}$, $k = 1, \dots, x(t)$ are conditionally independent and that the following equalities hold

$$\mathbf{P}\{A_{k,t} \mid I_t, W_{(\cdot)}\} = \min\left\{\frac{u_t W_k}{x(t)}\right\} \stackrel{\text{def}}{=} \pi_{k,t} \quad (3.1)$$

Here $u_t = \mathbf{E}S(t)$ is a deterministic function that reflects the importance of the corpus used. This (3.1) is quite a weak assumption, practically dictated by the way indexes and thesauri grow in practice.

The results to be quoted below are some of the ones in [25] and concentrate on the case that $W_k \equiv 1$. Obviously, much more general models should be examined. For one thing the importance of a key phrases is certainly not a constant and, moreover, is likely to change in time.

Set

$$h(t) = \mathbf{E}x(t), \quad a = \mathbf{E}\frac{Wu\lambda}{Wu + \lambda},$$

then, besides other asymptotic results, assuming $\lambda_t \equiv \lambda$, $u_t = u$

$$\lim_{t \rightarrow \infty} \mathbf{E}\left|\frac{x(t)}{k(t)} - \frac{a}{\lambda}\right| = 0$$

which in the case that $W_{(\cdot)} \equiv 1$ is precisely the result (2.1) of the crude “taking averages” analysis of section 2 above. It remains to be sorted out what happens in more general circumstances.

There is also an exhaustion result:

$$\lim_{t \rightarrow \infty} \mathbf{P}\{K(0) \subset S(t)\} = 1 \quad \Leftrightarrow \quad \sum_t \frac{u_t}{n(t)} = \infty$$

which means that if the observation rate is not too small compared to the growth rate of the field then, eventually, the (latent) key phrases at time zero will all be found.

Shifting time this means that for any time t a certain amount of time later all potential key phrases $K(t)$ will have been recognized with probability 1. What is still needed is an estimate of how much time that will take (depending of course on growth and observation rates).

For a number of statistical estimators of the parameters of the model see loc. cit.

4. Identification clouds.

Now suppose that we have a near perfect list of key phrases for, say, mathematics. That is not the case, but adequate lists do exist for certain subfields, [12, 16, 18, 21, 22, 23, 24].

Even then there remain most serious open problems of information storage and retrieval. To start lets look at an example. Here is a phrase that occurred in an abstract that came my way for indexing purposes some 6 years ago:

“ ... using the Darboux process the complete structure of the solutions of the equation can be obtained.”

At first sight, speaking linguistically, it looks like there is here a perfect natural key phrase to be assigned, viz. “Darboux process”. Presumably, some sort of stochastic process like “Cox process”, “Galton-Watson process”, “Dirichlet process”, or “Poisson process”.

However, there is no concept, or result, or anything else in mathematics that goes by the name “Darboux process”. Also the context did not look like having anything to do with stochastics and/or statistics. Had the abstract been classified — it wasn't — using the MSCS (Mathematics Subject Classification Scheme) it would have carried a number like 58F07 (1991 version) or 37J35 (2000 version), neither of which have anything to do with stochastics.

The proper name “Darboux” is also not sufficient to identify what is meant; there are too many terms with “Darboux” in them: “Darboux surface”, “Darboux Baire 1 function”, “Darboux property”, “Darboux function”, “Darboux transformation”, “Darboux theorem”, “Darboux equation”,....(these all come from the indexes of [13]).

Or take the following example from [33]. Suppose a querier is interested in “prenatal ultrasonic diagnosis”. Then texts containing phrases like “in utero sonographic diagnosis”, “sonographic detection of fetal ureteral obstruction”, “obstretic ultrasound”, “ultrasonics in pregnancy”, “midwife’s experience with ultrasound screening” should also be picked up. Or, inversely, when assigning key-phrase metadata to documents, the documents containing these phrases should also receive the standard controlled key phrase “prenatal ultrasonic diagnosis”.

One way to handle such problems (and a number of other problems, see below) is by means of the idea of identification clouds.

Basically the “*identification cloud*” of an item from a controlled list of standardized key phrases is a list of words and possibly other (very short) phrases that are more or less likely to be found near that key phrase in a scientific text treating of the topic described by the key phrase under consideration.

For instance the key phrase

Darboux transformation

could have as (part of its) identification cloud the list

soliton
 dressing transformation
 Liouville integrable
 completely integrable
 Hamiltonian system
 inverse spectral transform
 Bäcklund transformation
 KdV equation

KP equation
 Toda lattice
 conservation law
 inverse spectral method
 exactly solvable
 ...
 (37J35, 37K (the two MSC2000 classification codes for this area of mathematics))
 ...

And in fact this particular identification cloud solves the “Darboux process” problem above. The surrounding text contained such words as ‘soliton’, ‘completely integrable’, and others from the list above. The appropriate index phrase to be attached was “Darboux transformation”.

What the authors of the abstract meant was something like “repeated use of the process ‘apply a Darboux transformation’ will give all solutions”.

A human mathematician, more or less expert in the area of completely integrable systems of differential equations, would have no difficulty in recognizing the phrase “Darboux process” in this sense. Thus what identification clouds do is to add some human expertise to the thesaurus (list of key phrases) used by an automatic system.

The idea of an identification cloud is part of the concept of an enriched weak thesaurus as defined and discussed in [15, 17, 20]

5. Application 1: automatic key phrase assignment.

A first application of the idea of identification clouds is the automatic assignment of key phrases to scientific documents or suitable chunks of scientific texts.

It is simply a fact that it often happens that in an abstract or chunk of text a perfectly good key phrase for the matter being discussed is simply not present or so well hidden that linguistic and/or statistical techniques do not suffice to recognize it automatically.

The idea here is simple. If enough of the identification cloud of a term (= standard keyphrase) is present than that key phrase is a good candidate at least for being assigned to the document under consideration.

Here are two examples.

5.1. Example.

Two-dimensional iterative arrays: characterizations and applications.

We analyse some properties of two-dimensional iterative and **cellular arrays**. For example, we show that **arrays** operating in $T(n)$ time can be sped up to operate in time $n + (T(n) - n)/k$.

.....

computation. Unlike previous approaches, we carry out our analyses using *sequential machine characterizations of the iterative and cellular arrays*. Consequently, we are able to prove our results on the much simpler **sequential machine models**.

iterative array
 sequential characterization of cellular arrays
 sequential characterization of iterative arrays
 characterization of cellular arrays
 characterization of iterative arrays

arrays of processors

speed-up theorem

Here the available data consisted of an abstract (which is only partially reproduced here). In bold, in the abstract itself, are indicated the index (thesaurus) phrases which can be picked-out directly from the text. Below the original text are five more phrases, that can be obtained from the available data by relatively simple linguistic means, assuming that one has an adequate list of standard key phrases available. For instance “sequential characterization of cellular arrays” and sequential characterization of of iterative arrays” result from the phrase in italics in the abstract fragment above. Note that instead of doing (more or less complicated) linguistic transformations, these could also have been obtained by means of identification clouds. There are advantages in this because there are so very many possible linguistic transformations.

Then, in shadow, there is the term “array of processors”. This one is more complicated to find. But, given an adequate standard list, and with “array”, “processors” and “machine” all in the available text, it is recognizable, using identification clouds, as a term that belongs to to this document.

Finally, in bold-shadow, there is the key phrase “speed-up theorem” a well known type of result in complexity theory. In the text there just occurs “sped up”. Certainly, unless one has a good list of (standard) key phrases available, this would be missed. Also purely linguistic means plus such a very good list are clearly still not sufficient; there is no way that one can have a key phrase extraction rule like ‘if “sped up” occurs “speed-up theorem” is a likely key phrase’. However, “sped up” plus supporting evidence from the context in the form of a sufficient number of terms from the identification cloud of “speed-up theorem” being present, would do the job.

Example 5.2.

Sequential and **concurrent behaviour in Petri net theory**.

Two ways of describing the **behaviour of concurrent systems** have widely been suggested: arbitrary **interleaving** and **partial orders**. Sometimes the latter has been claimed superior because **concurrency** is represented in a ‘true’ way; on the other hand, some authors have claimed that the former is sufficient for all practical purposes. **Petri net** theory offers a framework in which both kinds of **semantics** can be defined formally and hence compared with each other. Occurrence sequences correspond to **interleaved behaviour** while the notion of a process is used to capture **partial-order semantics**. This paper aims at obtaining formal results about the

...

more powerful than **inductive semantics** using

...

of **nets** which are of **finite synchronization** and **1-safe**.

sequential behaviour in Petri net theory

Petri net theory

axiomatic definition of processes

interleaving semantics

1-safe nets

The style coding is the same as in the previous example. Here, the constituents “1-safe” and “nets” of “1-safe nets” actually occur in the text. But they are so far apart that without standard

lists and identification clouds the phrase would probably not be picked up. The same holds for the key phrase “interleaving semantics”.

Afterwards, I checked against the full text whether these extra key phrases were indeed appropriate. They were. Two more examples can be found in [19] or [20]. These are all actual examples which occurred in the corpora used to produce the indexes [16, 18].

A C-program that takes as input a keyphrase list with identification clouds and a suitably prepared corpus of documents (chunks of text or abstracts) and that gives as output the same corpus with each item enriched with automatically assigned keyphrases has been written in the context of the EC project “TRIAL SOLUTION” (Febr. 2000 - Febr. 2003), [39]. It also outputs an html file for human use which can be used to check how well the program worked. This validation test is currently (2002) under way.

It is already clear, that the idea of identification clouds needs refinements; certainly when used on rather elementary material (as in TRIAL SOLUTION). Two of these will be briefly touched on below.

6. Application 2: dialogue mediated information retrieval

Given a keyphrase list with identification clouds, or, better, an enriched weak thesaurus, it is possible to use a dialogue with the machine to refine and sharpen queries. Here is an example of how part of such a dialogue could look:

Query: I am interested in spectral analysis of transformations?

Answer: I have:

- spectral decompositions of operators in Hilbert space (in domain 47, operator theory, 201 hits)
- spectral analysis (in domain 46, functional analysis, 26 hits)
- spectrum of a map (in domain 28, measure theory, 62 hits)
- spectral transform (in domain 58, global analysis, 42 hits)
- inverse spectral transform (in domain 58, global analysis, 405 hits)

Please indicate which are of interest to you by selecting up to five of the above and indicating, if desired, other additional words or key phrases.

The way this works is that the machine scans the query against the available identification clouds (using some (approximate) string matching algorithm, e.g. Boyer-Moore) and returns those keyphrases whose ID clouds match best, together with some additional information to help the querier make up his mind.

7. Application 3: distances in information spaces.

As it is, the collection of standard keyphrases is just a set. It is a good idea to have a notion of distance on this set: are two selected standard key phrases near, i.e. closely related, or are they quite far from each other. Identification clouds provide one way to get at this idea: two phrases which have large overlap in their identification clouds are near to each other.

A use of this, again dialogue mediated, is as follows.

Query: I am interested in something related to <StandardKeyPhrase 1>. Please give me all standard keyphrases that are within distance x of this one.

For other ways to define distances on information spaces (such as the space of standard key phrases) and other potential uses of distance, see [20].

A distance on the space of key phrases is related to a distance on the space of documents, see loc. cit. This is also most useful in dialogue mediated querying. Suppose a really good document for a given query has been found. Then a very useful option is

Query: I am interested in documents close to <Document 1>. Please give me all standard documents that are within distance x of this one and which have two or more of the following key phrases in their key phrase metadata field.

Some search engines have a facility like this in the form of a button like ‘similar results’ in SCIRUS of Elsevier. But not based on distances in information spaces.

7. Application 4: disambiguation.

Ambiguous terms are a perennial problem in (automatic) indexing and thesaurus building.

Identification clouds can serve to distinguish linguistically identical terms from very different areas of the field of inquiry in question. E.g. “regular ring” in mathematics, or the technical term “net” which has at least five completely different meanings in various parts of mathematics and theoretical computer science. For instance ‘transportation net’ in optimization and operations research, ‘net of lines’ in differential geometry, ‘net’ in topology (which replaces the concept of a sequence in topological spaces where the notion of sequence is not good enough), ‘communication net’, ‘net(work) of automata’,

Identification clouds also serve to distinguish rather different instances of the same basic idea in different specializations. E.g. *spectrum* of a commutative algebra in mathematics, *spectrum* of an operator in a different part of mathematics, and *spectrum* (of a substance) in physics or chemistry are distantly related and ultimately based on the same idea but are in practice completely different terms.

Possibly an even worse problem is caused by phrases and words which have very specific technical meanings but also occur in scientific texts in everyday language meanings. A nice example is the technical concept “end” as it occurs in group theory, topology and complex function theory (three technically different though related concepts). Searching for “end” in a large database such as MATH of FIZ/STN (Berlin, Karlsruhe) is completely hopeless. Searching for “end” together with its ID cloud for its technical meaning in group theory would be a completely different matter. Note that specifying group theory as well in the query would not help much; there are simply too many ways in which the word ‘end’ occurs (end of a section, to this end, end of the argument, end of proof, ..). There are many more words like this; also phrases. For instance ‘sort’ (as in many sorted languages or sorting theory) and ‘bar’ (as in bar construction). For more about the ‘story of ends’, see [19].

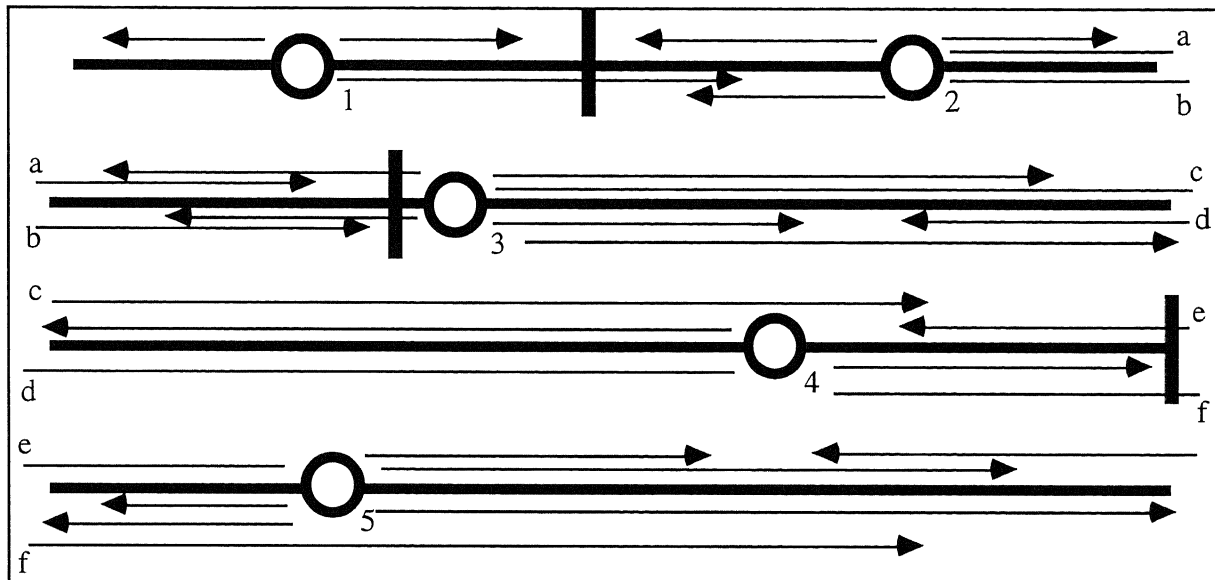
8. Application 5. Slicing texts

One important thing made possible by modern electronic technology, i.e. computers and the internet, is the systematic reuse of (educational) material and the composing of books and documents exactly tailored to the needs of an individual user. For instance a teacher may like the introduction to the idea of a topological space from book1, consider the formal definition of book2 better and may want to use some examples from book3, some exercises from book4, and some historical comments from book5.

The question arises how to chop up a longer text into chunks (slices) that can be efficiently recombined to form such individually tailored texts. This is the subject of the EC Framework 5 project TRIAL SOLUTION (Febr. 2000 - Febr. 2003), [39]. If the to be sliced document is well structured, for instance composed using LaTeX2e, the structure imposed by the author is a good guide where to slice and this is what TRIAL has so far concentrated on.

Now suppose we have a long section (slices should be relatively short; certainly not more

than one computer screen) or an unstructured text, i.e. no clear markings indicating sections, subsections etc., the exact opposite of a good LaTeX2e document. Suppose also that key phrases have been found and marked in the text and that for each key phrase the evidence for including that key phrase has also been marked; i.e. for each key phrase the corresponding items from its identification cloud have been marked. Treating the text as a long linear string we get a picture like the following.



The numbered fat hollow circles are key phrases in the text which is depicted as a fat horizontal line running over four lines; the arrows connect a key phrase to a member of its identification cloud. If the key phrase is not actually present, the fat circle is the centre of mass of the terms indicating its virtual presence. An arrow can run over more than one line; then labels are used to indicate how it continues.

It is now natural to cut the text at those spots where the number of arrow lines is smallest. For instance, at the three points indicated by fat vertical lines. This can be done at several levels to get a hierarchical slicing. To be able to do this optimally one needs a good stochastic model for the distribution of key phrases through a text and also for the distribution of identification cloud items for a key phrase.

The problem of slicing a text into suitable chunks also comes up in other contexts. For instance in the matter of automatic generation of indexes and identification clouds, see section 16 below, and in the topic of text mining, see [36], p. 7.

9. Weights.

One thing that emerged out of the use of identification clouds in the project TRIAL SOLUTION was that it is wise to give weights (numbers between 0 and 1 adding up to 1) to the elements making up an identification cloud.

Here is an example:

```
<KEYPHRASE NAME=<Burgers-Gleichung> THRESHOLD=<0.67>>
  <WORD VALUE=<Burgers-Gleichung> WEIGHT=<0.7>>
  <WORD VALUE=<Burgers> WEIGHT=<0.4>>
  <WORD VALUE=<Gleichung> WEIGHT=<0.2>>
```

```

<WORD VALUE=<Boussinesq> WEIGHT=<0.025>>
<WORD VALUE=<nichtlinear> WEIGHT=<0.025>>
<WORD VALUE=<Evolutionsgleichung> WEIGHT=<0.025>>
<WORD VALUE=<Solitonlösung> WEIGHT=<0.025>>
<WORD VALUE=<Transformation> WEIGHT=<0.025>>
<WORD VALUE=<KdV> WEIGHT=<0.025>>
<WORD VALUE=<sinh> WEIGHT=<0.025>>
<WORD VALUE=<Gordon> WEIGHT=<0.025>>
<WORD VALUE=<Hirota> WEIGHT=<0.025>>
<WORD VALUE=<Kadomzev> WEIGHT=<0.025>>
<WORD VALUE=<Pedviashwili> WEIGHT=<0.025>>
<WORD VALUE=<Soliton> WEIGHT=<0.025>>
<WORD VALUE=<Bäcklund> WEIGHT=<0.025>>
<WORD VALUE=<inverse spektral> WEIGHT=<0.025>>
<WORD VALUE=<HOPF> WEIGHT=<0.025>>
<WORD VALUE=<COLE> WEIGHT=<0.025>>
<\KEYPHRASE>

```

This particular identification cloud is designed to find occurrences of the Burgers equation as it occurs in the area of completely integrable dynamical systems (soliton equations, Liouville integrable systems). There are other areas where it occurs; a matter which is further discussed in section 17 below.

Of course if the phrase itself occurs that is enough as reflected by the first item in the ‘WORD VALUE list’. Note further that the occurrence of “Burgers” and of “equation” is not quite enough. There is a good reason for that. For one thing there is also a concept called “Burgers vector” (in connection with torsion in differential geometry); also “Burgers” is a fairly common surname. Further “equation” is of such frequent occurrence (in mathematics) that it can turn up just about anywhere. Thus the occurrence of both “Burgers” and “equation” in a chunk of text is not enough to decide that “Burgers equation” is a suitable key phrase for that chunk. But if three or more of the sort of words that belong to completely integrable dynamical systems are also present one can be quite sure that it is indeed a suitable key phrase.

Of course if formula recognition, see section 13 below, were available one would add to the list above

```

<WORD VALUE= < $u_t - u_{xx} - uu_x = 0$ > WEIGHT=0.7>

```

(which is the Burgers equation in formula form.)

How to assign weights optimally is a large problem. Obviously this cannot be done by hand: a more or less adequate list of standard key phrases for mathematics needs at least 150 000 terms. I propose to use, among other things, something like the following adaptive procedure.

Suppose one has an identification cloud of a term consisting of items $1, \dots, n$ with weights p_1, p_2, \dots, p_n adding up to 1. Let a subset $S \subset \{1, 2, \dots, n\}$ be successful in identifying the phrase involved. Then the new weights are:

$$\text{For } i \in S, \quad p'_i = p_i \left(\frac{\sum_{i \in S} p_i + r(1 - \sum_{i \in S} p_i)}{\sum_{i \in S} p_i} \right)$$

$$\text{For } i \notin S, \quad p'_i = p_i - rp_i$$

where r is a fixed number to be chosen, $0 < r < 1$. (Note that the new weights again add up to 1; note also that the $i \in S$ increase in relative importance and the $i \notin S$ decrease in relative importance; if $S = \{1, \dots, n\}$ nothing happens.) This is an adaptation of a reasonably well known algorithm for communication (telephone call) routing that works well in practice but is otherwise still quite fairly mysterious, [4, 34].

10. Application 6. Synonyms

There are a variety of things one can do with identification clouds to handle the well known problem of synonyms.

Suppose there are two synonymous key phrases. Then providing both of them with the same identification clouds (including both phrases themselves also as items) will cause both of them to be assigned to those documents where that is appropriate. This would probably be the best way to handle this in most circumstances.

Should, however, one prefer to have just one standardized key phrase this can be handled by having the alternative key phrases in the identification cloud of the standardized one with a weight equal or higher than the threshold value of the selected standardized key phrase; see section 9 above for how these weights would work.

11. Application 7. Crosslingual IR

There are a variety of applications of the idea of identification clouds when dealing with multilingual situations in information retrieval and storage. Suppose for instance one has English language key phrases supplied with German language identification cloud items. One bit of use one can make of this is to attach English language key phrases to German language papers and chunks of text.

Another one is as follows. Suppose we have a German speaking querier who is looking for English language documents as in dialogue mediated search (section 6 above). Then the same German identification clouds attached to English key phrases permit the machine to handle a German language query.

12. Application 8. Automatic classification

Here “automatic classification” means assigning to a document one or more classification numbers from the MSC2000 (Mathematics Subject Classification Scheme, [30]), or its precursor MSC1991. For instance

14M06: linkage

54B35: spectra

55M10: dimension theory

In this setting, instead of key phrases, it is the classification numbers from MSC2000 which are provided with information clouds. This also give these classification numbers substance and meaning. The terse descriptions like the three above are far from sufficient to indicate adequately what is meant (even to experts on occasion).

Certainly the mere occurrence of the word “linkage” should not be considered sufficient to

assign a paper or chunk of text the classification number 14M06. First of all one would like to be sure that the document in question is about algebraic geometry. this can be done by referring to the identification cloud of the parent node 14 (Algebraic geometry), and second one would like additional evidence like the presence of such supporting phrases as “complete intersection”, “determinantal variety”, “determinantal ideal”,

Inversely, a paper may very well be about the rather technical group of ideas “linkage” without ever mentioning that particular word.

The other two examples just given also need more complete descriptions as to what is really meant (disambiguation and more). For instance there are notions of spectrum in many different parts of mathematics: combinatorics, number theory (two different ones at least), homological algebra, ordinary and partial differential equations, dynamical system theory, harmonic analysis, operator theory, general topology, algebraic topology, global analysis, statistics, mechanics, quantum theory, Most are somehow related to the original idea of the spectrum of a substance as in physics/chemistry; but some others are completely different.

The exact phrase “dimension theory” occurs four times in MSC2000 while the stem “dimension” occurs no less than 94 times.

13. Application 9. Formula recognition

Recognizing (or finding) formulas in scientific texts is (in any case at first sight) a completely different matter from recognising or finding key phrases. First because formulas are two dimensional and second because the symbols occurring in formulas are not standardized (except a few like the integral sign and the summation sign). Even a standard symbol like π for the number 3.1415... that gives the radius of the circumference of a circle to its diameter, is not a reliable guide. The Greek letter π is also often used for, for instance, all kinds of mappings in various kinds of geometry, for partitions in combinatorics, and for permutations in group theory.

For instance the two expressions

$$\int_0^1 \frac{\sin x}{x} dx \quad \text{and} \quad \int_0^1 \frac{\sin t}{t} dt$$

mean exactly the same thing. It is the pattern rather than the actual glyphs that occur which determine what a formula means.

And even the patterns are not all that fixed. For instance here are a few versions of that very well known concept in mathematics and engineering, the (one dimensional) Fourier transform (there quite a few more):

$$\hat{f}(\xi) = \int f(x)e^{-i\xi x} dx \quad \text{see [28], p. 120}$$

$$\tilde{f}(p) = (2\pi)^{-1/2} \int_{-\infty}^{\infty} dq f(q) \exp(-ipq) \quad \text{see [38], p. 134}$$

$$g(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)e^{-iux} dx \quad \text{see [37], p. 34}$$

$$C(\lambda) = \int_{-\infty}^{+\infty} e^{-2i\pi\lambda x} f(x) dx \quad \text{see [32], p. 176}$$

$$F(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad \text{see [29], p. 376}$$

$$F(\omega) = \mathcal{F}[f(t)] = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad \text{see [27], p. 103}$$

$$F(y) = \int_{-\infty}^{\infty} f(x)\exp(-2\pi ixy)dx \quad \text{see [5], p. 45}$$

$$\hat{f}(\chi) = \int_G \bar{\chi} f d\lambda \quad \text{see [26], p. 359}$$

Most of the variations come from different notations for the exponential, the insertion or deletion of normalizing factors involving π , the engineering tradition of writing $\sqrt{-1}$ as j instead of i (as in most of mathematics and physics), different notations for integrands, and putting in or leaving out the integration limits.

Still, it is not easy to define formally what kind of transformations are allowed. On the other hand, trained mathematicians have no difficulty in recognizing any of the above (except possibly the last) as instances of a Fourier transform. Quite generally trained mathematicians can look at a text in their fields of expertise in a language totally unknown to them and still decide what topics the text deals with and at what level things are treated just by looking at the formulas. Whether that sort of expertise can be taught to machines is an open question. The field of formula recognition is still in its infancy — I would say it is still in a foetal stage.

Identification clouds can help. The idea is the same as before. But instead of a key phrase it is now a (standardized) formula which has an identification cloud attached to it. In the present case one can imagine that the (obligatory) presence of an integral sign, the (also obligatory) presence of the function symbols ‘exp(.)’ or ‘ $e^{(\cdot)}$ ’ and an integration variable ‘d’ in the formula, plus supporting evidence in the form of the occurrence of (some of the) words like “transform”, “Fourier”, “spectral analysis”, “harmonic”, ... in the surrounding text would do not a bad job in identifying Fourier transform formulas.

Some preliminary work on formula recognition using identification clouds is planned in the EC project [8].

14. Context sensitive IR

In a very real sense the idea of identification clouds is that of context sensitive approximate string recognition. Even if the string itself, that is the key phrase in question, is not recognized the context may provide sufficient supporting evidence to conclude that that string should be there as a key phrase. But the way the context is used is very much nonsophisticated. There is no (complicated) grammatical analysis or anything like that. I believe that this is how trained scientists function. They just look casually at the surrounding text of, say, a formula, and on the basis of what they see there decide what it is all about. I do not believe they really do any kind of grammatical analysis or transformations. Indeed, many of us are incapable of doing anything like that, for very often we have to work in foreign languages which are far from perfectly known to us.

15. Models for ID clouds

So far there has been no worry about just how the supporting evidence coming from identification clouds is distributed. This does not matter too much if one is dealing with the problem of assigning key phrases to short chunks of text or to abstracts. Say, to documents of the size of one computer screen or one A4 page maximal.

Things change drastically if one has to deal with longer chunks of text and especially if one has to assign key phrases, classifications, and other metadata to complete, full text documents.

Obviously if the items of an identification cloud for some key phrase of classification of formula or ... are spread around very far, are very diffuse, or if they are concentrated in just a few lines of text, makes an enormous difference.

Thus what is needed for many applications touched upon in this paper is an experimentally justified stochastic model on how the items of an identification cloud are distributed. And for that matter, how key phrases, whether actually present or not, are distributed over a document. This is of particular importance for the application “slicing of documents” discussed in section 8 above.

16. Automatic generation of identification clouds.

Take a large enough, well indexed corpus, and divide it into suitable chunks called documents. For instance take the 700 000 abstracts of articles in the STN/FIZ database Math (ZMG data)¹, or take as documents the sections or pages of a large handbook or encyclopaedia such as the Handbook of Theoretical Computer Science, [35] or the Encyclopaedia of Mathematics, [13], or an index like [18, 22] Now use a parser for prepositional noun phrases (PNP’s) (or an automaton recognizing PNP’s) or a software indexing program like TExTract or CLARIT, [2, 3, 9, 10, 11], to generate from these documents a list of key phrases, keeping track of what phrases come from what document. Now assign, as ID clouds, to the items of the list of keyphrases, those words and phrases found by, say, the software indexing program, which occur in the same document as the key phrase under consideration.

17. Multiple identification clouds.

Picture the set of all documents (chunks of text) in mathematics as a space. For instance a discrete metric space as in [20]. There may then very well be several distinct regions in this space where a given key phrase, like “Burgers equation” occurs with some frequency. In this case one may well need several different identification clouds for the same key phrase, even though there is no ambiguity involved. This happens in fact in the case at hand. The Burgers equation has relations with the field of completely integrable systems: it itself has soliton solutions and it is also related to what is probably the most famous soliton equation, the KdV equation (Korteweg de Vries equation). The identification cloud above in section 9 was designed to catch this type of occurrence of the concept. On the other hand it is the simplest nonlinear diffusion equation and plays a role as such and in discussions of turbulence. To catch those occurrences a rather different set of supporting words and phrases is needed (like diffusion, turbulence, eddy, nonlinearity,...). Just combining the two identification clouds is dangerous because then, by accident, the various different collections of supporting evidence phrases together may combine to give a spurious assignment. One can also not concentrate too much on the proper name “Burgers” for the reasons mentioned in section 9 above.

18. More about weights. Negative weights.

Another refinement that came out of the experiences with the TRIAL SOLUTION project is that it could be a very good idea to allow negative weights. Let’s look at an example.

“The next topic to be discussed is that of the Fibonacci *numbers*. The generating formula is very simple. But all in all these numbers and their surprisingly many applications are sufficiently *complex* to make the topic very interesting. Similar things happen in the study of fractals.”

¹ Though this one is not really well indexed in the sense that the key phrases assigned are not from a controlled list. However, if the intention would be to generate the controlled list at the same time as the corresponding ID clouds, this material would be most suitable.

Or even worse:

“These mixed spectrum solutions must be *numbered* among the more *complex* ones of the KdV equation. Still they can be not neglected.”

Both ‘complex’ and ‘numbers’ occur in the first fragment of text above (italized). But, obviously it would be totally inappropriate to assign the technical keyphrase ‘complex numbers’ to this fragment. A negative weight on ‘Fibonacci’ in the ID cloud of ‘complex numbers’ will prevent that.

For the second text fragment the technique of stemming, which needs to be used, will give “number”, and “complex” also occurs. But here also it would be totally inappropriate to assign the key phrase “complex numbers”. It is not so easy to see how to avoid this.

There are still other possible sources of difficulties because “complex” is also a technical term in algebraic topology and homological algebra so one can have a fragment like

“The Betti numbers of this cell complex are...”

or still worse:

“The idea of a simplicial complex numbers among the most versatile notions that ...”

Here even the exact phrase “complex numbers” occurs and negative weights are a must to avoid a spurious assignment.

Quite generally it seems fairly clear that the presence of the constituents of a standard key phrase in a given chunk of text is by no means sufficient to be sure that that key phrase is indeed appropriate. This is especially the case for concepts that are made up out of frequently occurring words like “complex numbers” or “boundary value formula”. But we have also seen this in the case of the “Burgers equation” above in section 9. For the case of the phrase “complex numbers” one needs an identification cloud like

```
<KEYPHRASE NAME=<complex numbers> THRESHOLD=<0.47>>
  <WORD VALUE=<complex numbers> WEIGHT=<0.5>>
  <WORD VALUE=<complex> WEIGHT=<0.2>>
  <WORD VALUE=<numbers> WEIGHT=<0.2>>
  <WORD VALUE=<field> WEIGHT=<0.06>>
  <WORD VALUE=<imaginary part> WEIGHT=<0.06>>
  <WORD VALUE=<real part> WEIGHT=<0.06>>
  <WORD VALUE=<absolute value> WEIGHT=<0.06>>
  <WORD VALUE=<Gauss> WEIGHT=<0.06>>
  <WORD VALUE=<argument> WEIGHT=<0.06>>
  <WORD VALUE=<principal value> WEIGHT=<0.06>>
  <WORD VALUE=<vector representation> WEIGHT=<0.06>>
  <WORD VALUE=<addition> WEIGHT=<0.06>>
  <WORD VALUE=<multiplication> WEIGHT=<0.06>>
  <WORD VALUE=<Fibonacci> WEIGHT=<-0.5>>
  <WORD VALUE=<Betti> WEIGHT=<-0.5>>
</KEYPHRASE>
```

So that besides “complex” and “number” one needs at least 2 more bits of supporting evidence to have a reasonable chance that the fragment in question is indeed has to do with the field of complex numbers. On the other hand if at least 8 of the last ten positive weight terms of the identification cloud above are present one is also rather sure that the fragment in question has to do with the field of complex numbers. The tentative identification cloud given above reflects this. But it is clear that assigning weights properly is a delicate matter; it is also clear that much can be done with weights.

Thus also in the case of occurrences of the same concept in the same part of mathematics, more than one identification cloud may be a good idea, reflecting different styles of presentation and different terminological traditions.

The concrete examples of section 2 above also illustrates the possible value of negative information.

19. Further refinements and issues.

There are a good many other issues to be addressed. Here is one. It is more or less obvious that making one keyphrase list with ID clouds for all of science and technology is a hopeless task. What one aims at is instead an Atlas of Science and Technology consisting of many weak thesauri that partially overlap, may have different levels of detail, and may focus on different kinds of interest. Much like a geographical atlas which has charts of many different levels of detail and many different kinds (mineralogical, roads and train lines, soil types, height, type of terrain, demographical, climatological, ...). Here the problem arises of how to match the different ‘charts’.

Another one is how to adapt the adaptive scheme of section 9 to a situation with negative weights and how to handle insertion and deletion of ID cloud members.

In an enriched weak thesaurus a key phrase has not only words in its identification cloud but also one or more classification numbers from MSC2000. In turn these classification numbers have identification clouds. The idea is that once a candidate key phrase has been found these are used to check that indeed the paper is related to the topics described by those classification numbers. This idea of referring to other (secondary) identification clouds can be used in all of the various applications described above. For instance it is needed of one uses a formula to identify a key phrase as suggested at the end of section 9. Such referring to other identification clouds was also briefly mentioned in section 12 above. Just how this should be implemented stil needs to be worked out.

Probably the most crucial issue to be addressed at this stage is the formulation of a good probabilistic model of ID clouds complete with statistical estimators, see section 15. A project in this direction has been started by the CWI, Amsterdam together with the IMI, Lithuanian Acad. of Sciences, Vilnius.

References.

1. Jean Aitchison, Alan Gilchrist, *Thesaurus construction*, Aslib, 2-nd Edition, 1990.
2. H Bego, *TExtract: snelle en eenvoudige 'back of the book index' generatie*. In: L G M Noordman, W A M de Vroomen (ed.) *Derde STINFON conferentie*, 1993, 214.
3. H Bego, *TExtraxt. Back-of-the-book index creation system*, TXYZ, Utrecht, 1997.

4. G Bel, P Chemouil, J M Garsia, F Le Gall, J Bernusso, *Adaptive traffic routing in telephone networks*, Large Scale Systems **8**(1985), 267-282.
5. D C Champeney, *A handbook of Fourier theorems*, Cambridge Univ. Press, 1987.
6. Ian Crowlesmith, *Creating a treasure trove of words*, Elsevier Science World. 14-15, 1993.
7. Ian Crowlesmith, *The development of a biomedical thesaurus*, NBBI Thesaurus Seminar. 1993.
8. J Davenport, a.o., *MKMNET. Mathematical knowledge management network*, Project IST-2001-37057. September 2002 - December 2003. 2001.
9. David A Evans, *Snapshots of the Clarit text retrieval*, Preprint, copies of slides, Carnegie Mellon university, 1994.
10. D A Evans, K Ginther-Webster, M Hart, R G Lefferts, I A Monarch, *Automatic indexing using selective NLP and first-order thesauri*. In: A Lichnérowicz (ed.), *Intelligent text and image handling*, Elsevier, 1991, 524-643.
11. David M Evans, Robert C Lefferts, *Clarit-Trec experiments*, Preprint, Carnegie Mellon, 1994.
12. Revaz V Gamkrelidze, Franz Guentner, Michiel Hazewinkel, Arkady I Onishchik, *ERETIMA: English Russian bilingual thesaurus for Invariant theory, Lie groups, Algebraic geometry, Dynamical systems, Optimal control, Commutative algebra*. INTAS project 96-0741, 2001.
13. Michiel Hazewinkel (ed.), *Encyclopaedia of mathematics; 13 volumes including three supplements*, KAP, 1988-2002.
14. Michiel Hazewinkel, *Classification in mathematics, discrete metric spaces, and approximation by trees*, Nieuw Archief voor Wiskunde **13** (1995), 325-361.
15. Michiel Hazewinkel, *Enriched thesauri and their uses in information storage and retrieval*. In: C Thanos (ed.), *Proceedings of the first DELOS workshop*, Sophia Antipolis, March 1996, INRIA, 1997, 27-32.
16. Michiel Hazewinkel, *Index "Artificial Intelligence", Volumes 1-89*, Elsevier, 1997. Large size
17. Michiel Hazewinkel, *Topologies and metrics on information spaces*. In: J Plümer R Schwänzl (ed.), *Proceedings of the workshop: "Metadata: qualifying web objects"*, <http://www.mathematik.uni-osnabrueck.de/projects/workshop97/proc.html>, 1997,
18. Michiel Hazewinkel, *Index "Theoretical Computer Science", Volumes 1-200*, Theoretical Computer Science **213/214** (1999), 1-699.

19. Michiel Hazewinkel, *Key words and key phrases in scientific databases. Aspects of guaranteeing output quality for databases of information*. In: Proceedings of the ISI conference on Statistical Publishing, Warsaw, August 1999, ISI, 1999, 44-48.
20. Michiel Hazewinkel, *Topologies and metrics on information spaces*, CWI Quarterly **12:2**(1999), 93-110. Preliminary version: <http://www.mathematik.uni-osnabrueck.de/projects/workshop97/proc.html>
21. Michiel Hazewinkel, *Index Discrete Applied Mathematics Vols 1-95*, Discrete Applied Mathematics **106** (2000), 1-261.
22. Michiel Hazewinkel, *Index Discrete Mathematics Vols 1-200*, Discrete Mathematics **227/228** (2001), 1-648.
23. Michiel Hazewinkel, *Index Information processing letters Vols 1-75*, Information processing Letters, **78:1-6** (2001), 1-448.
24. Michiel Hazewinkel, *Index journal of logic and algebraic programming volumes 1-45* 68, J. Logic and Algebraic Programming **50:1-2** (2002), 1-103.
25. Michiel Hazewinkel, R Rudzkis, *A probabilistic model for the growth of thesauri*, Acta Appl. Math. **67** (2001), 237-252.
26. Edwin Hewitt, Kenneth A Ross, *Abstract harmonic analysis. Volume 1*, Springer, 1963.
27. Hwei P Hsu, *Outline of Fourier analysis*, Unitech, 1967.
28. Yitzak Katznelson, *An introduction to harmonic analysis*, Dover reprint, 1976. Original edition: Wiley, 1968.
29. Benjamin G Levich, *Theoretical physics. Volume 1*, North Holland, 1970.
30. Editors of Mathematical Reviews and Zentralblat für Mathematik, *MSC2000 classification scheme*, 1998.
31. R Rudzkis, *Letter to M Hazewinkel*, 2002.
32. Laurent Schwartz, *Mathematics for the physical sciences*, Hermann, 1966.
33. Alan F Smeaton, *Progress in the application of natural language processing to information retrieval tasks*, The Computer Journal **35:3** (1992), 268-278.
34. P R Srikantakumar, K S Narendra, *A learning model for routing in telephone networks*, SIAM J. Control and Optimization **20:1** (1982), 34-57.
35. Jan van Leeuwen (ed.), *Handbook of theoretical computer science*, Elsevier, 1990,

36. Ari Visa, *Technology of text mining*. In: Petra Perner (ed.), *Machine learning and data mining in pattern recognition*. Second international workshop, Leipzig, 2001, Springer, 2001, 1-11.
37. Norbert Wiener, *The Fourier integral and certain of its applications*, Cambridge Univ. Press, 1933.
38. Kurt Bernardo Wolf, *Integral transforms in science and engineering*, Plenum, 1979.
39. B Ingo Dahn, TRIAL SOLUTION. Tools for reusable integrated adaptable learning systems; standards for open learning using tested interoperable objects and networking, Project IST-1999-11397: Febr. 2000-May 2003, 1999.