Information Processing & Management Vol. 26, No. 1, pp. 171-186, 1990
Printed in Great Britain.

0306-4573/90 $3.00 + .00

CONSTRUCTING LITERATURE ABSTRACTS BY COMPUTER: TECHNIQUES AND PROSPECTS

CHRIS D. PAICE
Department of Computing, University of Lancaster,
Bailrigg, Lancaster LA1 4YR, United Kingdom

(*Received* 28 *February* 1989; *accepted in final form* 20 *July* 1989)

Abstract—The automatic generation of abstracts has always been a rather neglected problem. Text summarisation methods based on artificial intelligence approaches can only be made to work in very restricted domains; consequently, practical abstracting has been attempted by simple *ad hoc* methods. The early workers concentrated on techniques for extracting informative sentences from documents. Inevitably, this resulted in 'abstracts' which were highly disjointed, due especially to the presence of dangling anaphoric references. Subsequently, work has been carried out on the twin problems of (a) recognising anaphoric references, and (b) constructing passages in which no anaphoric references are left dangling. The correct handling of definite noun phrases however represents an unsolved problem. The question of how to achieve proper balance in auto-abstracts is a problem which has hardly been addressed hitherto; it appears that assignment of selected textual material into 'abstract-frames' may offer a solution. However, radical progress on the construction of satisfactory abstracts now seems to require an adequate theory of text structure.

1. INTRODUCTION

The automatic construction of abstracts from the texts of documents has always been a relatively neglected corner of information science. Despite the increasing availability of documents in electronic form, and the spread of desk-top publishing systems, abstracts continue to be produced manually. It therefore seems timely to review the state-of-affairs in this field, and to suggest the direction in which progress may lie.

We may define an abstract as a concise summary of the central subject matter of a document. The minimal function for any useful abstract is *indicative*—helping a reader to decide whether it will be worthwhile to look at the full document. Many abstracts also contain *informative* or *substantive* material, such as main results and conclusions; in this case the reader may be able to glean useful information without needing to refer to the full document at all. Some workers in this field state that their aim is to produce indicative abstracts, whereas, others refer to informative abstracts. In terms of the techniques used, the distinction does not seem to mean very much; in practice, we can regard the indicative function as essential and the informative function as desirable. Obviously, in attempting to evaluate auto-abstracts, it would be appropriate to separate the two functions. The inclusion of *critical* and *comparative* material does not appear to be a feasible proposition at present, and will be ignored in this review.

We could, of course, envisage systems which involve cooperation between human being and computer—whether in the form of 'intelligent' workstations for abstractors, or interactive systems for marking up texts to facilitate subsequent processing [1]. Such systems are ignored here.

Obviously, we might consider constructing abstracts from a wide variety of text types. In practice, much of the reported work has been concerned with abstracting papers from learned journals and conference proceedings. In this review, it may be assumed that documents are of this type unless otherwise indicated. There is some evidence that even within this narrow genre, variations in text structure and content can have a radical effect on the success of abstracting programs [2].

The aim of automatic abstract construction is pragmatic, and in consequence the methods used have been fairly superficial. The usual approach is to select various portions from the complete text, and then fiddle around in an attempt to make them look as though they belong together. Radical analysis of the meaning of the text is not attempted. This may be contrasted with artificial intelligence approaches to text summarisation, in which detailed semantic information is handled explicitly. For instance, DeJong's FRUMP system analyses news articles by instantiating slots in one of a set of predefined frames [3]. When the analysis is complete, a script is used to generate a summary of the information held in the relevant frame. In Rau's SCISOR system, detailed linguistic analysis of a text (or indeed of several interrelated texts) results in construction of a semantic graph, which is convenient for intermediate storage [4]. A natural language generator may then produce summaries from the stored material. Other comparable systems have been described by Rumelhart [5], Lehnert [6], Tait [7], and Hahn and Reimer [8].

It must be understood that these summarisation systems are only capable of processing texts within a narrow domain, whose characteristics are predictable and well understood. Thus, FRUMP operates on news stories, and SCISOR is restricted to reports on corporate mergers and acquisitions. Hahn and Reimer's TOPIC system [8] is designed to summarise texts about micro-processor systems. There is little prospect of such approaches succeeding, at least for the foreseeable future, with the much more variable material found in typical journals.

Section 2 of this review will deal with methods which have been used or proposed for selecting 'extract-worthy' sentences from documents. Section 3 will discuss the problem of producing coherent, stylistically acceptable output; the discussion will centre on work which has been carried out over the past few years at Lancaster University. Section 4 will then discuss the possible use of 'abstract-frames' for ensuring proper coverage and balance. I do not propose to address the thorny question of how abstracts ought to be evaluated.

2. AUTOMATIC SENTENCE EXTRACTION

The first experiment on automatic abstracting was reported in a paper by H. P. Luhn published in 1958 [9]. This and all subsequent work reported up to about 1970 was concerned with methods for producing *extracts*, that is, sets of sentences selected to provide a good indication of the subject matter of the document. The general approach is to examine each sentence, looking for clues to its importance; to compute a score for the sentence based on the clues found; and then either to select all sentences whose score exceeds some threshold, or to select the highest scoring sentences, up to a certain total. The sentences are then printed in their order of occurrence in the original text. At least seven distinct types of clue to sentence significance have been tried, and these are now briefly summarised.

2.1 *The frequency-keyword approach*

This approach involves looking for mentions of important concepts within each sentence, that is, occurrences of words and phrases which refer to central concerns of the document. This means that before the sentence can be scored there must exist a list of candidate index terms for the document. In all of the experiments so far reported, the index terms are individual words (sometimes conflated), and are commonly called *keywords*.

Following Luhn, the usual method for identifying keywords is to take the complete text of the document, to delete the common function words by means of a stoplist, and then to sort all the remaining distinct words (sometimes conflated, sometimes not) into a frequency-ordered list. Any words which occur below a certain frequency are regarded as likely to be chance occurrences, and are therefore deleted from the list.

For each of the sentences in the document, keyword occurrences are noted, and various methods are used for computing the sentence score. Luhn [9] and Oswald [10], noting the importance of phrases for denoting concepts, looked for groups or clusters of keywords, and based the overall sentence score on these. Edmundson [11] weighted keywords according to their frequency in the document, and summed the keyword weights for each sentence. Rath *et al.* [12] and Earl [13] used simple keyword counts as the basis for computing sentence scores.

2.2 *The title-keyword method*

This method resembles the last in that a list of keywords is compiled for a document before starting the sentence-scoring process. In this case, the keywords are selected from the title, subtitle and headings of the document, on the basis that the main concepts of a document are likely to be mentioned there. Obviously, a stoplist is used to eliminate non-significant words.

The only trial of this method was by Edmundson [11], who assigned a higher weight to keywords from the main title than to those found only in lesser headings. Considering the simplicity of this method, it is interesting to note that in a comparative experiment (see Section 2.8) it did somewhat better than the frequency-keyword method.

2.3 *The location method*

It was Baxendale in 1958 who first noted that, within a paragraph, the first sentence is usually the most central to the theme of the text, while in many other cases it is the last [14]. Building on this observation, Edmundson defined the location method, in which sentences received various scores according to whether they occurred at the beginning or end of a paragraph, near the beginning or end of the whole document, or below a heading [11]. More recently, the importance of the position of a mention within a text has been confirmed in psychological studies by Kieras [15].

2.4 *Syntactic criteria*

In an experiment reported in 1970, Earl [13] examined the hypothesis that the extract-worthiness of sentences might be correlated with their syntactic structure. She used a program to assign one of '13 traditional parts of speech' to each word of a text, and then represented each sentence by its parts-of-speech pattern. She then used a further program to convert the above patterns into phrase structure representations. After processing more than 3,000 sentences in this way, she found, for both forms of representation, that around [illegible]9% of the patterns occurred once only, so that no correlations could be obtained.

Earl was understandably disheartened ("these are disappointing results"), and apparently did not pursue the idea further. It does seem at least possible that judicious simplification of the grammar and pruning of extraneous material from sentences might have [illegible]ed some progress to be made. My own view, however, is that there is little prospect [illegible] turning this into a useful extraction method.

[illegible] *[illegible]e cue method*

[illegible]is method is based on the notion that certain of the words or phrases occurring in [illegible]nce, though not in themselves keywords, nonetheless provide an indication of [illegible] the sentence deals with important material. Edmundson [11] used a list contain-[illegible] 'bonus words,' whose occurrence would increase the sentence score, and 73 'stigma [illegible] which would decrease the score. Bonus words included superlatives and value [illegible] as "greatest" and "significant," whilst the stigma words included anaphors and [illegible]pressions (e.g., "hardly," "impossible").

[illegible] *et al.* [16] modified the cue method by constructing a 'word control list' which [illegible] short phrases as well as individual words. Each entry was accompanied by a [illegible], which indicated whether the word or phrase was a strong/weak posi-[illegible] indicator, or whether it had some other significance. The list contained a pre-[illegible] negative indicators, so that the extraction process relied more on rejection [illegible]on of sentences. Some of the entries also had a syntactic code, but this was [illegible]oses other than simple sentence extraction (see section 3).

[illegible] *[illegible]or-phrase method*

[illegible]e strong positive entries in Rush's word control list were expressions which [illegible]mpany explicit statements about the topic of the text; examples are "pur-[illegible]rk." Indicator phrases are more elaborate constructs of the same kind, "The main aim of the present paper is to describe. . . ."

"The purpose of this article is to review. . . ."

"In this report, we outline an investigation into. . . ."

"Our investigation has shown that. . . ."

The usefulness of these constructs for sentence extraction was first discussed by Paice in 1981 [17]; their use has recently been re-examined by Black and Johnson [18]. The first three of the examples are likely to introduce indicative material, whilst the fourth may be followed by an informative statement (see Section 1 above).

Clearly, very many different indicator phrases are possible. However, it turns out that there are only seven or eight really distinct types, each of which can be identified with reference to a 'template' allowing substitution of alternative words or phrases—compare the first two examples above. Cumulative weights may be attached at various points in a template, thus allowing a score to be returned even if the template is not matched right to the end. Similarly, the most reliable template types can have larger total weights than poorer ones.

Some documents, especially discursive ones, may contain several good indicator phrases, whereas others contain none at all. Clearly, it would not be safe to rely on indicators alone in an extracting system, but where they *do* occur they appear to be very useful.

2.7 *Relational criteria*

Skorokhod'ko [19] outlines an extracting method which involves building a 'semantic structure' for the document. This is a kind of graph structure in which the sentences are vertices, and significant inter-sentence links are represented by edges. The idea is that the most important sentences in a document are those which are related to the largest number of other sentences: such sentences are prime candidates for extraction. Potential links are considered to exist wherever distinct sentences contain references to the same concept.

Words in the text are assigned weights on the basis of their 'statistical and formal-semantical characteristics,' and word-word relations are detected on the basis of lexical similarity or by using a thesaurus. On the basis of such information, significant inter-sentence relations are identified. When this stage is complete, each sentence is scored as a function of the number of sentences to which it is significantly related, and the degree of change in the network structure which would result from its deletion.

2.8 *Performance of extracting procedures*

Several of the reports of extracting experiments are accompanied by some samples of output. These generally look reasonable, but it is hard to form any real impression of how useful the extracts would be in practice, still less to judge between the different approaches. Moreover, there must obviously be a question whether the selection of examples for publication was entirely impartial.

Unfortunately, the only substantial comparative evaluation of extracting methods is that carried out over 20 years ago by Edmundson, working at Thompson-Ramo-Wooldridge, Inc. [11]. Edmundson prepared a 'target extract' for each of his documents, and then scored each auto-extract according to how well its sentences agreed with the target sentences. He tried four distinct extracting methods, optimised them to some extent by adjusting parameters, and then combined them in all possible combinations, so that 15 different auto-extracts were produced and scored for each document.

The four methods he tried, listed here in order of decreasing performance, were the location method, the cue method, the title-keyword method and the frequency-keyword method. The best results of all were obtained by combining the first three of these methods, using a linear function with appropriate coefficients to obtain the overall scores. Edmundson drew the general conclusion that:

"future automatic abstracting methods must take into account syntactic and semantic characteristics of the language and the text: they cannot rely simply on gross statistical evidence."

Since no other detailed comparative evaluation has been reported, there is a tendency to afford Edmundson's results more significance than perhaps they warrant. His experiment used 200 documents in a single field (Chemistry), and investigated just four distinct methods, most of them capable of much adjustment of detail. Moreover, he used only one evaluation criterion which is in fact open to serious criticism. The use of target extracts is unsatisfactory because it assumes that there is just one correct set of sentences; the low level of agreement between human extractors [20] illustrates the weakness of this assumption.

2.9 *Comments*

The keyword methods (sections 2.1 and 2.2) are based on the very reasonable notion that any terms which are germane to the topic of a document ought to be included in an extract. However, due to the methods used for identifying keywords and for scoring sentences, the inclusion of particular terms is in fact a highly chancy affair. Now suppose it is actually *certain* that a particular term is central to the topic of the document; in that case, rather than depending on the contingencies of a scoring system, it should be mandatory to include that term in the abstract.

Of course, in natural language processing, certainty is hard to come by. Nonetheless, there must be certain contexts and constructions which at least make it highly probable that the term is central. Examples are the placing of a phrase in italics in a leading position in the document, or the use of an indicator construct such as, "In this paper, we describe the theory of . . . . ." It need not necessarily be the introductory sentence itself which is extracted, since that may sometimes be difficult for other reasons; but the phrase must be flagged as 'essential,' and a sentence containing it found from somewhere.

More generally, the whole idea of scoring needs to be critically examined especially for cases where various different kinds of evidence—perhaps conflicting—appertain to the same sentence. Edmundson [11] combined different kinds of score using a linear combination, but this in effect implies that the clues provide independent items of evidence, which merely require to be cumulated together. In fact, it seems at least possible that different clues may sometimes interact in special ways. It may be that ultimately some kind of knowledge-based approach will be required to weigh the various kinds of evidence which may co-occur.

3. TEXTUAL COHESION

[illegible] is easy to appreciate that computer-produced extracts suffer from two main problems: lack of cohesion and lack of balance. The former problem is discussed in the present [illegible]tion while the problem of achieving balance in the content of an abstract will be [illegible] with in Section 4.

[illegible] our discussion of cohesion, we will be mainly concerned with the occurrence of [illegible] references within a sentence which can only be understood by reference to material [illegible]where in the text. This covers anaphoric reference, lexical or definite reference, and rhetorical connectives. Certainly, there are other levels of coherence, such as [illegible] whether the 'role' of a particular sentence in an extract is consistent with the roles [illegible] sentences, but discussion of these will be left until later.

[illegible] problem of the lack of cohesion in abstracts seems to have been raised first by [illegible] pp. 17–18], and was discussed again by Edmundson [11], who explicitly [illegible] the problem posed by anaphors in extracts. The problem was first tackled on [illegible] level by Rush and his co-workers [16], whose abstracting program ADAM (see [illegible] 2.5) attempted to deal with the problem by *either* adding preceding sentences [illegible] any sentence whose first clause contained an anaphor, *or* if more than three [illegible] sentences would need to be added, by deleting the original sentence. The pro[illegible]ded mechanisms for deleting unimportant clauses and parenthetical material [illegible] It is probably fair to regard Rush's as the first computer program to pro[illegible] rather than extracts. The idea of adding adjacent sentences was studied fur[illegible] who coined the term *aggregation* for it. Relevant work has also been [illegible]en [21].

Quite recently, Black and Johnson [18] have performed experiments which seem to confirm that cohesiveness is a desirable property. They compared extracts produced by an Earl-type method [13] with aggregated passages based on indicator phrases and with manually-produced abstracts. In terms of informativeness, ease of reading and ease/speed for locating desired information, the order of quality was abstract > aggregated passage > extract. However, due perhaps to the small scale of the experiment (five documents, 35 Ss), most of the results were not statistically significant. In Black and Johnson's work, the aggregation was performed manually. To do it by computer raises a host of problems, some of which are outlined in the following sections.

3.1 *The recognition of anaphors*

Rush *et al.* acknowledged that certain potentially anaphoric words have 'multiple uses,' and that contextual criteria would be required for deciding the matter. They gave the example of "it," which is not anaphoric if followed shortly afterwards by "that." They seemed to believe that this problem of *anaphor recognition* could be solved by means of a few simple rules. This might well be true if we only needed a success rate of 75% or so, but that would hardly suffice for the production of good abstracts. If we want to get it right nearly every time, the whole thing becomes much more difficult.

The problem of recognising anaphors has recently been studied in some detail by Katzer, Bonzi and Liddy at Syracuse University, N.Y. [22, 23], and by Paice and Husk at Lancaster University, U.K. The work at Syracuse led to a set of rules which specified the grammatical characteristics of the various usages of 95 potentially anaphoric words. Many of the rules are not in a form which can be implemented by computer at the present time; nonetheless, they provide a useful basis for future development.

The work carried out at Lancaster involved the implementation of a computer program known as GARP* which scans a text and applies appropriate contextual rules whenever it encounters a potentially anaphoric word [24]. These rules (in contrast to those developed by the Syracuse group) are entirely superficial in nature. Their purpose is not only to decide whether the current word or expression is anaphoric or not, but also, if it *is* anaphoric, to estimate whether the antecedent lies within the current sentence or elsewhere. This is often done by means of a simple positional test; for instance, if the word "these" is preceded by less than ten other words, its antecedent is assumed to lie in the previous sentence.

It should be mentioned here that Craven also has devised a set of contextual rules for recognising referential expressions [1]; these are used in an interactive system for marking up the structure of a text.

Most of the anaphor recognition rules developed at Lancaster have not been described in the literature, and it therefore seems appropriate to provide a brief overview here. The reader who is not interested in the details may skip on to Section 3.4.

Note first that for present purposes we are only interested in words and expressions which may refer outside the current sentence; thus we disregard words like reflexive pronouns which are always *structural*, i.e., syntactically bound to their antecedents or postcedents. By the same token, rhetorical connectives like "moreover" and "on the other hand" *are* considered as anaphors, even though they may lack an antecedent in the usual sense. Cataphors, which refer *forward* in the text, are infrequent in the kinds of text that concern us, and have been largely ignored.

3.1.1 *True anaphors.* These include third person pronouns, nominal demonstratives, indefinites, quantifiers and nominal substitutes. Some typical rules are:

- "he," "him" and "his" are:—
  internal if "he," "him" or "his" occurs earlier in the same sentence;
  external if not later than 10th word of sentence;
  internal otherwise.

*('Gareth's Anaphor Recognition Program', after Gareth Husk who wrote the code)

- "more" is: –
  nonanaphoric in the phrases "more or less," "more and more";
  nonanaphoric if followed by "than" with no more than four words between;
  external if not later than 10th word of sentence;
  internal otherwise.

Rules can also make reference to predefined word lists, as in the following:

- "that" is: –
  nonanaphoric if preceded by a research-verb (e.g., "assum – ", "show – ", "demonstrat – ", "suggest – ") with no more than one word between;
  nonanaphoric if followed immediately by a pronoun, article, quantifier, demonstrative, preposition, common-verb or comma;
  external if not later than 10th word of sentence;
  internal otherwise.

It may be instructive to compare the above with the corresponding rule (wording modified) from the Syracuse rule set:

- "that" is: –
  nonanaphoric {subordinating conjunction} if preceded by a cognitive verb ("know," "think," "fear," "say" etc.) or by a corresponding nominalisation ("assume," "suggest," "explain" etc.), *and* if the following clause contains no empty syntactic slot;
  nonanaphoric {subordinating conjunction} in the expressions "but that," "in that," "such that," "so that," "in order that," *and* if the following clause contains no empty syntactic slot;
  nonanaphoric {subordinating conjunction} in the expression "that is";
  anaphoric {relative pronoun} if the following clause has a syntactically empty slot (e.g., ". . . . a system that diagnosed human illness");
  anaphoric {demonstrative adjective} if followed by a noun or adjective;
  anaphoric {demonstrative pronoun} otherwise.

Note that the Syracuse researchers classify relative pronouns as anaphoric, whereas from our viewpoint they are structural.

The pronoun "it" requires an exceptionally complex set of rules to deal with such constructs as "it . . that," "it . . to," "it . . whether," "it . . which" and parenthetical "it." These are discussed in detail in a separate paper [25].

3.1.2 *Logical and rhetorical connectives.* These are cohesive features which indicate the nature of the relationship between a sentence or clause and its predecessor or successor. Halliday and Hasan [26] have provided an elaborate classification for these connectives, whilst their functional properties have been discussed by van Dijk [27]. Many of them are unambiguous, requiring a simple positional test to decide whether the reference is internal or external (e.g., "but," "in fact," "nonetheless," "however," "on the other hand," "indeed"); these also have the useful property that, if they are a nuisance, they can be safely deleted. Some of the others require additional tests, e.g.,

- "then" is: –
  nonanaphoric if "if" occurs earlier in the same sentence;
  external if no later than fifth word of sentence;
  internal otherwise.
- "so" is: –
  nonanaphoric if immediately followed by "to say" or "to speak";
  nonanaphoric if immediately followed by "far" or "many";
  nonanaphoric if immediately followed by "much"; {test simplified}
  nonanaphoric if immediately preceded by "and";

nonanaphoric if followed by "as" or "that" within ten words;
external if no later than sixth word of sentence;
internal otherwise.

Polonskaya has also published a set of rules for identifying these connectives [28], but they are less developed than the Lancaster rules.

3.1.3 *Long-range references.* These are metatextual references to distant parts of the text. They are not properly dealt with in the existing rule-set, apart from one or two obvious examples such as "the above." It should however be reasonably easy to devise rules for one well-defined group of these, in which a long-range anaphor or cataphor is combined with the past participle of an 'explain' verb—for instance

"…. the method summarised earlier…."

"…. is discussed more fully in the next section…."

"…. in the following chapter…. will be discussed in detail…."

3.2 *Define noun phrases*

Noun phrases introduced by the definite article "the" are regarded as typically referential. In fact, it seems that in technical English somewhat more than half the cases are nonanaphoric (mostly structural). Easily the commonest of these is the construct "the … of", with one or two words between, which accounts for nearly a third of all instances of "the"; this rule has an exception in the case where "of" is preceded by a 'container' word, as in "the bucket of water" and "the list of addresses". Other similar rules have been defined for constructs such as "the … which", "the … between" and "the … in". Further nonanaphoric cases are exemplified by "the formula {xyz}," "the case," "the term," "the well-known," "the … range" (with zero or one word between) and "the … century" (with exactly one word between).

There are two other groups of words starting with "the" which, though they have a referent, do not need to be linked to an antecedent in the text. The first consists of words which have permanently available referents, which therefore do not need to be introduced explicitly. They are called *homophors*, and include expressions like "the sun," "the weather," "the future" and "the human race." Some other expressions, moreover, are homophoric within a particular text genre or domain; examples are "the literature" and "the periodic table."

The second group contains what might be referred to as 'social homophors,' since they mostly refer to well-known organisations or persons—for example "the United States" or "the President." In English, these words and expressions are normally capitalised, and this feature may be used to recognise them. Such a rule may appear laughably simple, but in practice it performs rather well.

For all instances of "the" which do not fit any of the foregoing rules, an antecedent must be located. Unlike typical anaphors, however, these *definite references* may link to items a long way off in the text, so that a simple internal/external distinction is not adequate.

The Lancaster GARP program includes a mechanism for searching for the most recent previous mention of the concept concerned [24]. Since a noun phrase may contain several substantive words, the program attempts to select up to four words as 'link arguments' from immediately after "the." Fewer than four link arguments will be accepted if another article, or a preposition, demonstrative, known verb or punctuation symbol is encountered. Moreover, certain words, though they do not halt the forward search for link arguments, may not themselves be used as arguments. Thus, in the fragment "… the whole indexing procedure was … " the word "whole" is not allowed as an argument, whilst "was" terminates the search, leaving "indexing" and "procedure" as link arguments. Once the arguments have been selected, the linker searches backwards to find the most recent word whose stem matches the stem of *one* of the link arguments. Thus, for the above arguments, a link could be formed to a sentence containing "… the documents were *indexed* by means of ... ". The above linking mechanism is unsatisfactory in a number of ways; possible improvements to it will be discussed in Section 3.4.

3.3 *The aggregation process*

A sentence or passage of text is said to be *tidy* if it contains no anaphor or other referential device that is not resolved within the same sentence or passage. At least at a certain level, a tidy passage is intelligible on its own, and is thus a suitable unit for including in an extract. Hence, if a sentence selected for extraction is found not to be tidy, adjacent sentences may be added one-by-one until a tidy passage has been constructed. If however the incipient passage becomes too large, then either one of the offending anaphors must be 'neutralised', or (as in the ADAM system [16]) the whole passage must be rejected.

The aggregation program written at Lancaster searches a text for indicator phrases (Section 2.6) in order to identify candidate sentences for extraction. The GARP rules outlined above are applied to each sentence in order to find any external anaphors. If only short-range external anaphors are present, the previous sentence is added to the passage, and the program then looks there for further anaphors. If a definite reference is found (see above) then several previous sentences may be added in one operation. The process continues in this way until no further dangling anaphors are found, or the start of the current section is reached [24].

Test runs were performed on the texts of several papers in the field of Computer Science. To our surprise, hardly any of the passages constructed were more than eight sentences long, and most were of one or two sentences only. This contrasted sharply with the results of some earlier experiments which had used texts from a much wider range of 'technical' domains and types. We could see three possible reasons for the difference:

a). Indicator phrases (not used in previous experiments) tend to select sentences mostly from near the beginning and end of a text, where perhaps anaphors are used less densely;
b). maybe computer scientists write simpler, less 'literary' (or less literate?) English than many other authors;
c). the earlier experiments used texts published in the early 1960s, whereas the Computer Science texts were all recent.

The initial output from the aggregation program included a large number of patently unsuitable passages, due mainly to the lax nature of the rules used for identifying indicator phrases. Suitable tightening of the rules allowed most of the inappropriate passages to be rejected. Nonetheless, some obvious problems remained and it was therefore felt that a formal evaluation would not be worthwhile until significant changes had been made to the system.

Figure 1 shows selected passages from one of the Computer Science papers, with anaphors marked in italics and indicator phrases in bold. Note that in passage (c), the connective "Similarly" has been flagged to indicate that, if it were deleted, the sentence containing it would be tidy. Most of the rhetorical connectives can be deleted without much harm, offering possible break-points in long passages.

3.4 *Desirable improvements*

In this section I shall summarise the more substantial changes which would appear necessary if the present approach is to result in satisfactory abstracts. Though framed in terms of the inadequacies of the existing Lancaster programs, these comments are strongly relevant to the development of future systems.

3.4.1 *Selection and aggregation.* The existing aggregation program uses only one criterion for identifying kernel sentences: the presence of indicator phrases. Many texts however have very few indicators, or even none at all, so plainly this criterion is not always sufficient. (As a matter of fact, the sample output in Figure 1 was chosen simply because it was the only paper processed in which a good number of indicators were found.) Among the extraction methods reviewed in Part 2, the cue method, the location method and the

(a)
"It is fair to say that major technical difficulties have been encountered in research into distributed operating systems. **It is our premise that** many of *the problems encountered* in distributed computing are a consequence of basing designs on an inadequate model. **In this paper,** we attempt to substantiate *this* view."

Overall text length = 54 or 0.88%

(b)
"**We consider models for** distributed computing with respect to two orthogonal issues – control and data."

Overall text length = 18 or 0.29%

(c)
"It is the assignment statement which causes many of the problems in programming languages; it is difficult to reason about the value of a data item due to the time-dependency of its value. *Similarly*(1), **we argue that** it is 'update in place' which leads to many of the problems in distributed systems."

Overall text length = 58 or 0.94%
(1) maybe deletion of word *Similarly*
Changed length would be 21 or 0.34%

(d)
"**The conclusion of the study is that** 'multilateral resource management with each management entity having absolutely equal authority' provides the basis for highly available, extensible systems."

Overall text length = 30 or 0.48%

(e)
"**We propose a model** in which system management is performed by a set of cooperating managers (comanagers), one on each node."

Overall text length = 25 or 0.41%

(f)
"**The data model we propose** is influenced by the concepts of functional programming languages [Hend80]."

Overall text length = 18 or 0.29%

(g)
"**We shall demonstrate how** the notion of time invariance can substantially ease problems of consistency."

Overall text length = 16 or 0.26%

(h)
"**In this paper, we have argued that** many of the problems of distributed computing stem from inadequate models."

Overall text length = 20 or 0.33%

Fig. 1. Selected output from aggregation program for a single text.

title-keyword method all appear to be useful and quite simple. Salton [29] has recently expressed the view that, with the use of up-to-date term weighting strategies, the frequency-keyword method may also repay further attention. Following Edmundson's approach [11] it would be possible to score sentences using some or all of these criteria together with a score for any indicator phrase.

The use of negative cue-expressions to reduce the score is certainly important: sentences which match indicator templates often contain features which show that the sentence is concerned with some 'local' issue, rather than that overall purpose or strategy—e.g., the word "now" in

"We now show how our m.c.t. algorithm performs the task of transforming ..."

We have seen how certain connectives may be deleted to provide break-points within passages. It would be useful to be able to create break-points for a wider variety of anaphors—for instance, by replacing pronouns by their antecedent expressions. The automatic identification of antecedents is however a major problem [30–32], so this possibility should probably be left aside for the present.

It is obvious from a consideration of Fig. 1 that a sequence of passages, even if they are all tidy, does not constitute a real abstract. This particular selection consists predominately of single sentences, some of which are very bald, and surely require the addition of either preceding or following sentences. The question of forward aggregation has not been seriously examined up to now, but there are certainly some cases where it would be useful. For instance, sentence (h) in Fig. 1 looks much more useful when accompanied by its following sentence:

> "In this paper, we have argued that many of the problems of distributed computing stem from inadequate models. In particular, inherently centralised control models and time variant data models lead to problems in terms of consistency, availability and extensibility."

A number of rhetorical connectives (e.g., "in fact," "in particular" and "thus"), when used at the start of a sentence, commonly serve to focus attention on the statement which follows, and could well be used for forward aggregation.

3.4.2 *Noun phrases and definite reference.* Some proper procedure for identifying and parsing noun phrases is most desirable, since without it the linking mechanism for definite references cannot operate reliably. Deep parsing may not be necessary, but a grammatical tagger would certainly be valuable. There are also other areas where some grammatical analysis is required. For example, in Fig. 1(a) the middle sentence contains the phrase "the problems encountered in", which is wrongly marked as an anaphor. The present GARP rules contain a test for a nonanaphoric "the ... in" construct, but this only allows one intervening word (so as not to misclassify a case like "the investigation resulted in"). Piecemeal adjustments to the rule set to cover more and more such cases is ultimately unacceptable.

Another matter in need of investigation is how best to recognise the antecedent of a definite reference—which of the word-stems in two possibly matching noun phrases are really significant? Also a mechanism is needed for detecting a semantic link when there is no formal resemblance between the two noun phrases. Thus in the example,

> "As the ISO standards become more stable they will replace the Coloured Book protocols—the transition from one to the other is currently under discussion."

the notion of "transition" is semantically implied by "replace." Some such cases may be handled by thesaurus, but in general the problem is a difficult one.

Hitherto it has been assumed that the appropriate way of handling definite references is to identify the most recent mention of the concept concerned. It can be argued on the contrary that it is the *first* mention of a concept that is usually the most significant, and thus it may be appropriate to select the sentence where the concept is actually introduced into the discourse. Doubtless in reality things will not be that simple, and this issue is unlikely to be resolved without careful study.

3.4.3 *Composition.* It should I think be clear that the decision on what passages to include in the final abstract cannot really be made until the whole text has been analysed. The real function of the passage selection process is to identify *candidate passages* (some containing alternative break-points) from which a final abstract may be composed. The strategy, used in many of the early systems, of simply selecting the highest-scoring sentences hardly seems adequate.

In composing an abstract, it is obviously desirable to avoid repetition. In Fig. 1 it is noticeable that sentence (h) says in effect the same thing as the middle sentence of passage (a). Where cases like this are detected, it is necessary to decide which version should be retained. More generally, abstracts should be composed in such a way as to ensure proper coverage and balance. How this might be achieved is discussed in Section 4.

In order to produce acceptable finished abstracts, final stylistic and cosmetic adjustments will be necessary. Mathis, Rush and Young [33] defined some transformations which are able in some circumstances to combine a pair of short sentences into a single longer sentence. Indicator phrases present a problem, since of themselves they do not convey useful content information, and they often contain words like "we" and "our" which are inappropriate in abstracts. Certain parenthetical or subordinate material may need to be deleted. In addition, the tense and voice of verbs may need to be harmonised; for instance, in Fig. 1 sentences (g) and (h) are out of step with the others.

3.4.4 *Summary.* Of the above improvements, the ones which appear most practicable in the short term include:

- integration of several sentence selection criteria, using a combined scoring scheme;
- improved linguistic procedures, especially for identifying noun phrases;
- addition of neighbours (even in absence of explicit anaphors) to augment short tidy sentences;
- elimination of parenthetical material;
- elimination of certain cases of repetition.

Unfortunately, the existing Lancaster software, which was written in the Pascal language, is probably not capable of supporting all of these improvements; a programming tool with more linguistic capability is required. A strong candidate here would seem to be the Direct Clause Grammar (DCG) formalism, defined by Pereira and Warren within the Prolog language [34]. Black and Johnson [18] have found DCG very useful for identifying indicator phrases; the formalism also appears highly suitable for encoding the anaphor recognition rules, and for performing many of the other linguistic operations required.

4. BALANCE AND COVERAGE

The abstracting system discussed in Section 3 is in effect no more than a linear descendent of the ADAM system [2, 16], and as such its potential is limited. Ultimately, it seems that progress with automatic abstract generation must depend on the existence of a satisfactory theory of text structure. The relevant idea here is that each sentence (or other coherent segment of text) plays a specific role, both in relation to the overall purpose of the text and to other nearby sentences. If these roles and relationships could be properly characterised they should provide a much sounder basis for sentence selection than some vague and undifferentiated notion of 'sentence importance.'

Quite a lot of theoretical and empirical work has been done since the mid 1970s on the logical and rhetorical structuring of textual material [35–38]. McKeown [39] has actually made use of 'rhetorical predicates'—i.e., of a functional classification of the roles of sentences and statements—in her TEXT system, which uses a schema-based approach to generate informative paragraphs about the contents of a database. However, none of this work is concerned with the problem of how to determine textual structures automatically.

Certainly, expository texts seem to contain a lot of clues which might be useful for elucidating their structure, including both lexical clues (especially rhetorical and logical connectives) and presentational clues (e.g., use of italics and layout). Some preliminary work along these lines has been reported by Berson *et al.* [40] and by Craven [41]. However, the structures built by their systems appear to be too superficial to provide a basis for the generation of high quality abstracts.

The problem is that at present a great gap exists between the obvious, local structural features of a text, and the kind of sophisticated cognitive model of text structure and meaning exemplified by the work of Meyer [42]. There is no space here to speculate on how that gap might be bridged. I shall therefore just concentrate on one aspect of text structure which seems to offer the prospect of improving the general structure and balance of automatic abstracts.

4.1 *Textual superstructure*

In the discussion up until now it has been assumed, though barely stated, that abstracts do and should consist of continuous, freely-formatted text. In fact this is not the only possibility, as is made clear by Kent [43], who describes various 'stylized' arrangements. Among these is the *formatted abstract*, which is set out in a *pro forma* with headings such as "Purpose of study," "Procedures," "Findings." The main sections may be further subdivided—e.g., "Procedures" into "General methodology," "Characteristics of population," "Sampling procedures" etc. This is considered to be helpful both to the reader, who can readily identify relevant portions of the abstract, and to the abstract writer, who is helped to achieve proper coverage and balance.

This does not at all imply that typical free-text abstracts do not possess a structure, but simply that their structure is not made explicit. A reader is able readily to discern that different parts of an abstract provide different kinds of information, and that there is a certain commonality between the functional structures of different abstracts, particularly within the same domain. Instructional texts make it very clear to trainee abstractors what functional elements they should consider including in their abstracts [44, 45]. Recent empirical research by Liddy [46–48] has provided detailed information about how abstracts in two subject areas (educational research and psychology) are structured in practice.

Obviously, it is not only abstracts which possess a structure; almost any text can be analysed into a small number of major components, which tend to occur rather predictably (though not inflexibly) in texts of that type. Following van Dijk [49], I shall refer to this structure as the *superstructure* of the text. It appears that the superstructure of an abstract will normally correspond in a rather direct way to the superstructure of the full text which it summarises. This correspondence may be used to guide the composition of the abstract.

4.2 *Abstract-frames*

In the opening paragraphs of this review, mention was made of systems such as FRUMP [3] which use schema-based (often frame-based) formalisms for generating textual output. An important point to appreciate is that these systems use *semantic* schemas, which provide ready-made frameworks for representing objects, ideas, events or activities which are typical of the domain. Semantic analysis of texts is then greatly assisted because the 'semantic shape' of the concepts which will be encountered is largely predictable. Each domain is characterised by a special repertoire of carefully-constructed semantic schemas, one of which is selected to represent each discourse concept. Yet even with this approach, misleading interpretations can easily be produced, while outside the domain no meaningful processing is possible.

If we desire a system which can produce useful summaries of text in an extensive and perhaps ill-defined domain, then the use of semantic schemas seems not to be viable. Rather, it is necessary to consider approaches which, though they may be several steps beyond the 1970s-style abstracting systems, nonetheless rely on relatively undetailed features of text.

A promising possibility here, arising from the discussion by Liddy [47], is to leave aside semantic schemas in favour of 'superstructural' schemas. An important aim of Liddy's work was to investigate

> "the clues which indicate to humans how to instantiate their mental frame of a particular text type based on clues and frame expectations fulfilled in the text. If this can be determined it then becomes feasible to automatically instantiate a frame-like data structure." [47, pp. 29–30]

Liddy's work was concerned not with generating abstracts, but with analysing the structure of abstracts so that it might subsequently be possible to use them more effectively for information retrieval [47, 50]. Hence, she wished to assign all the pieces of text in an abstract to one or other of the available slots. For abstracts generation, the task is to take an original document and select representative portions of text for insertion into the slots.

The idea in effect is to start with a blank *pro forma* for a formatted abstract, and then look for appropriate portions of text to fill all—or at least the most important—of the blanks.

Although the use of *abstract-frames* (as I shall call them) may provide a way round the strict domain dependency of the semantic schema approach, it would be wrong to imagine that complete domain independence can be achieved. It is clear that, even if we restrict attention to typical technical papers, different subject areas tend to require different forms of abstract. Nonetheless, it seems possible that quite a small number of abstract-frames—perhaps a few dozen—would suffice for a very wide range of technical literature. Depending on a document's general subject area and particular source (e.g., specific journal) a handful of useful frames could be made available.

I commented above that 'at least the most important' slots would be instantiated. This raises the thought that slot importance might be not a fixed feature of an abstract-frame, but might be adjusted according to requirements. The notion of 'tailor-made abstracts' was indeed considered by some of the early researchers [16, 51], who seemed to think that adjustments to scores and thresholds might do the trick. The use of abstract-frames now seems to offer a much more soundly-based means for achieving this goal.

4.3 *Slot instantiation*

The question which now arises is how the slots of an abstract-frame may be filled. This actually involves two distinct stages: first, each portion of the original text must be *classified* according to the slot or superstructural component to which it belongs; subsequently, one or more passages must be *selected* to represent each relevant slot, using methods of the type discussed in Sections 2 and 3 above.

Most of the documents to be abstracted will already contain explicit section and subsection headings, and these may help in the classification process. But obviously the information thus provided will usually not be detailed enough nor reliable enough to be used on its own. Fortunately, most texts are replete with other clues which may be used during both the classification and selection stages. Indeed, many of the methods presented earlier in this review may be adapted for use in the slot-filling process. In particular, cue expressions and indicator phrases provide valuable clues to the superstructural class of a sentence. Thus, a sentence containing "we have shown that" plainly belongs to the 'Findings' component of the document. Less obviously, the 'Historical background' component would typically be characterised by use of the past tense and the presence of citations, and the 'Experimental methods' component by use of the past passive, often with verbs such as "made," "used," "measured" and "determined."

The incidence of various such clues in the texts of abstracts was investigated in an experiment by Maeda [52]. More recently, Liddy has compiled extensive concordances of expressions which characterised the various structural elements of her abstracts [47]. Whilst presumably the clues present in full documents will vary in many ways from those in abstracts, it nonetheless seems likely that these sources will prove useful in devising rules for sentence classification.

4.4 *Coverage*

The use of abstract-frames may improve the balance of automatic abstracts, by ensuring that all of the important superstructural elements are represented; it does not however ensure that the coverage of substantive material is adequate. Many papers do not just deal with one simple topic, but with a group of interrelated topics. It is all too easy, with the methods so far described, for one aspect of a document's content to be included whilst another, equally or perhaps more important, may be omitted. Of course, if all of the important topics can be identified, then we can ensure that are all included (see section 2.9); however, clear-cut evidence may not always be readily available. There are however at least two sources of information within documents which might prove helpful: sectional organisation and 'orientation' material.

The use of sectional organisation is obviously full of problems, but the basic notion at least is clear: separate sections deal with separate aspects of a document's message. Thus, if two or three distinct sections are assigned to the same superstructural element each of them should if possible be represented. In effect, the relevant slot in the abstract-frame would be divided into sub-slots representing the various sections. If possible, one or more prominent concepts should be identified for each section to help focus the selection process; the section heading itself might provide this information.

Orientation material typically comprises one or more sentences which inform the reader how the document is organised; for instance, Section 1 of this review concludes with an orientation paragraph. Such statements provide a convenient advance listing of the main topics which will be dealt with, and are thus most valuable for defining sub-slots. They are easy to recognise, being rather stereotyped in form and including distinctive phrases such as "section 3," "final part," "will deal with," "we will describe," etc. (see also section 3.1.3).

5. CONCLUSIONS

This review has dealt with what might be termed the 'old fashioned' approach to automatic abstract generation. Since a system along these lines has not yet been fully realised, it is impossible to know how successful it would be for practical purposes. If our aim is to produce abstracts which cannot be told from manual abstracts, then it is hard to believe that systems relying on selection and limited adjustment of textual material could ever succeed. Real progress in that direction would have to wait on the development of an adequate theory of text structure. However, for most purposes it may well be acceptable to provide a system which is able to produce flawed abstracts quickly and cheaply. It therefore seems entirely correct to continue work along the lines described here, with the aim of constructing a system which can be fully evaluated, both experimentally and in a practical situation.

Acknowledgments—I am grateful to Bill Black and A.W.S. Cater for helpful comments on drafts of this review. Some of the work described in Section 3 was carried out by Gareth Husk under grants from the University of Lancaster Research Grant Fund and from the British Library R & D Department (project number SI/G/734).

REFERENCES

1. Craven, T.C. An interactive aid for coding of sentence dependency structures. Can. J. Inf. Sci., 14(3): 32–41; 1989.
2. Pollock, J.J.; Zamora, A. Automatic abstracting research at the Chemical Abstracts Service. J. of Chem. Inf. and Comput. Sci., 15(4): 226–232; 1975.
3. DeJong, G. An overview of the FRUMP system. In: W.G. Lehnert and M.H. Ringle, eds. Strategies for Natural Language Processing. London: Lawrence Erlbaum, 1982; 149–172.
4. Rau, L.F. Conceptual information extraction and retrieval from natural language input. User-oriented Content-based Text and Image Handling, Proceedings of RIAO 88 conference, M.I.T., March 21–24; 1988; 424–437.
5. Rumelhart, D. Understanding and summarising brief stories. In: D. LaBerge and S. J. Samuels, eds. Basic Processes in Reading: Perception and Understanding. Hillsdale, NJ: Lawrence Erlbaum Associates; 1977; 265–303.
6. Lehnert, W.G. Plot units and narrative summarization. Cog. Sci., 5: 293–331; 1981.
7. Tait, J.I. Automatic Summarising of English Texts, Technical Report no. 47, University of Cambridge Computer Laboratory; 1982.
8. Hahn, U.; Reimer, U. The TOPIC Project: text-oriented procedures for information management and condensation of expository texts. Constance, W. Germany: University of Constance; 1985.
9. Luhn, H.P. The automatic creation of literature abstracts. I.B.M. J. of Res. Dev., 2(2): 159–165; 1958.
10. Oswald, V.A., Jr. *et al.* Automatic indexing and abstracting of the contents of documents, Los Angeles, Calif.: Planning Research Corp., 31 October 1959.
11. Edmundson, H.P. New methods in automatic abstracting. J. of ACM, 16(2): 264–285; 1969.
12. Rath, G.J.; Resnick, A.; Savage, R. The formation of abstracts by the selection of sentences: Part 1: sentence selection by men and machines. Am. Doc., 12(2): 139–141; 1961.
13. Earl, L.L. Experiments in automatic extracting and indexing. Inf. Stor. Ret., 6(6): 313–334; 1970.
14. Baxendale, P.B. Man-made index for technical literature—an experiment, I.B.M. J. of Res. Dev., 2(4): 354–361; 1958.
15. Kieras, D.E. Thematic processes in the comprehension of technical prose. In: Britton, B.K. and Black, J.B., eds. 1985: ref. [37], 89–108.
16. Rush, J.E.; Salvador, R.; Zamora, A. Automatic abstracting and indexing. II. Production of indicative abstracts by application of contextual inference and syntactic coherence criteria. J. of Amer. Soc. for Inf. Sci., 22(4): 260–274; 1971.
17. Paice, C.D. The automatic generation of literature abstracts: an approach based on the identification of self-indicating phrases. In: R.N. Oddy, Robertson, S.E., van Rijsbergen, C.J. and Williams, P.W., eds. Information Retrieval Research, Butterworths, 1981; 172–191.
18. Black, W.J.; Johnson, F.C. A practical evaluation of two rule-based automatic abstracting techniques. Exp. Sys. for Inf. Man., 1(3): 159–177, 1988.

19. Skorokhod'ko, E.F. Adaptive method of automatic abstracting and indexing. Information Processing 71, North Holland; 1972.
20. Rath, G.J.; Resnick, A.; Savage, R. The formation of abstracts by the selection of sentences: Part II: the reliablity of people in selecting sentences, Am. Doc., 12(2): 141–143; 1961.
21. Craven, T.C. Customized extracts based on Boolean queries and sentence dependency structures. Int. Classif., 16(1): 11–14; 1989.
22. Katzer, J.; Bonzi, S.; Liddy, E. Impact of Anaphora in Information Retrieval, Final report to National Science Foundation, School of Information Studies, Syracuse University, Syracuse, N.Y.; 1986.
23. Liddy, E.D.; Bonzi, S.; Katzer, J.; Oddy, E. A study of discourse anaphora in scientific abstracts. J. of Amer. Soc. for Inf. Sci., 38(4): 255–261; 1987.
24. Husk, G.D. Techniques for Automatic Abstraction of Technical Documents using Reference Resolution and Self-Indicating Phrases, M. Phil. dissertation, University of Lancaster, U.K.; 1988.
25. Paice, C.D.; Husk, G.D. Towards the automatic recognition of anaphoric features in English text: the impersonal pronoun it: Comp. Sp. Lang., 2: 109–132; 1987.
26. Halliday, M.A.K.; Hasan, R. Cohesion in English, Longman, London; 1976.
27. van Dijk, T.A. Connectives in Text Grammar and Text Logic. In van Dijk, T.A. and J.S. Petofi, eds. Grammars and Descriptions, Berlin & New York: W. de Gruyter; 1977: 11–63.
28. Polonskaya, O.R. Logical-semantic connectors in the English language as formal indicators of text cohesion. Aut. Doc. Math. Ling., 20(3): 91–99; 1986.
29. Salton, G., personal communication.
30. Hobbs, J.R. Resolving pronoun references. Lingua, 44(4): 311–338; 1978.
31. Hirst, G. Anaphora in Natural Language Understanding: A Survey, Springer-Verlag; 1984.
32. Rich, E.; Luper-Foy, S. An architecture for anaphora resolution. Proc. Second Applied Natural Language Processing Conf., Austin, Texas; 1988: 18–24.
33. Mathis, B.A.; Rush, J.E.; Young, C.E. Improvement of automatic abstracts by the use of structural analysis. J. of Amer. Soc. for Inf. Sci., 24(2): 101–109; 1973.
34. Pereira, F.C.N.; Warren, D.H.D. Definite clause grammars for language analysis—a survey of the formalism and a comparison with augmented transition networks. Art. Intell., 13: 231–278; 1980.
35. Grimes, J. The Thread of Discourse, The Hague: Mouton; 1975.
36. Janos, J. Theory of functional sentence perspective and its application for the purposes of automatic abstracting. Inf. Proc. Man., 15(1): 19–25; 1979.
37. Britton, B.K.; Black, J.B. Understanding expository text: A theoretical and practical handbook for analyzing explanatory text. Hillsdale, NJ: Lawrence Erlbaum Associates; 1985.
38. Fox, B.A. Discourse structure and anaphora: Written and conversational English, Cambridge Studies in Linguistics 48, Cambridge University Press; 1987.
39. McKeown, K. Discourse strategies for generating natural language text, Artificial Intelligence. 1–41; 1985. Also, McKeown, K. Text Generation, Cambridge University Press; 1985.
40. Berson, V.E.; Blekhman, M.S.; Zakharov, A.A.; Pevzner, P.P. Computer implementation of a system analyzing supersentence connections. Aut. Doc. Math. Ling., 18(5): 39–51; 1984.
41. Craven, T.C. Automatic recognition of sentence dependency structures. Proceedings of RIAO 88 conference, M.I.T., March 21–24, 1988; 911–924.
42. Meyer, B.J.F. Prose analysis: Purposes, procedures and problems. In: Britton, B.K. and Black, J.B., eds., 1985, ref. [37]: 11–64.
43. Kent, A. Textbook on mechanized information retrieval, Wiley-Interscience, Second Edition; 1966.
44. Maizell, R.E.; Smith, J.F.; Singer, T.E.R. Abstracting scientific and technical literature, Wiley-Interscience; 1971.
45. Cremmins, E.T. The art of abstracting, Philadelphia: ISI Press; 1982.
46. Liddy, E.D. Discourse-level structure in abstracts. Proceedings of the 50th annual meeting of A.S.I.S.; 138–147, Medford NJ: Learned Information; 1987.
47. Liddy, E.D. The Discourse-level Structure of Natural Language Texts: an Exploratory Study of Empirical Abstracts, Ph.D. dissertation, Syracuse University, N.Y.; May 1988.
48. Liddy, E.D. Structure of information in full-text abstracts. Proceedings of RIAO 88 conference, M.I.T., March 21–24, 1988; 183–195.
49. van Dijk, T.A. Macrostructures, Hillsdale, N.J.: Lawrence Erlbaum, 1980; chapter 3.
50. Bonzi, S.; Liddy, E.D. The use of anaphoric resolution for document description in information retrieval. Proceedings of the 11th SIGIR Conference on Research and Development in Information Retrieval, Grenoble, France, June 13–15, 1988.
51. Edmundson, H.P. Problems in automatic abstracting. Comm. of ACM, 7(4): 259–263; 1964.
52. Maeda, T. An approach toward functional text structure analysis of scientific and technical documents, Inf. Proc. Man., 17(6): 329–339; 1981.