The Application of Linguistic Processing to Automatic Abstract Generation

F.C Johnson, C.D Paice, W.J Black and A.P Neal

Dr. F.C. Johnson: Centre for Computational Linguistics, UMIST (currently at the Department of Library and Information Studies, The Manchester Metropolitan University, All Saints, Manchester M15 6BH)

Dr. C.D Paice: Department of Computing, University of Lancaster, Bailrigg, Lancaster LA1 4YR.

Mr. W.J. Black: Centre for Computational Linguistics, UMIST, PO Box 88, Manchester M60 1QD.

Dr. A.P Neal: Department of Computing, University of Lancaster, Bailrigg, Lancaster LA1 4YR.

Abstract

One approach to the problem of generating abstracts by computer is to extract from a source text those sentences which give a strong indication of the central subject matter and findings of the paper. Not surprisingly, concatenations of extracted sentences show a lack of cohesion, due partly to the frequent occurrence of anaphoric references. This paper describes the text processing which was necessary to identify these anaphors so that they may be utilised in the enhancement of the sentence selection criteria. It is assumed that sentences which contain non-anaphoric nounphrases and introduce key concepts into the text are worthy of inclusion in an abstract. The results suggest that the key concepts are indeed identified but the abstracts are too long. Further recommendations are made to continue this work in abstracting which makes use of text structure.

1. Introduction

This paper describes a project which was funded by the British Library Research and Development Department to develop techniques for generating abstracts of technical papers by computer. The approach taken was to select from source text sentences which give a strong

indication of the central subject matter and findings of the paper. In general, sentences may be selected on the basis of various statistical, grammatical, positional and presentational clues (Paice 1). Not surprisingly, concatenations of extracted sentences show a lack of cohesion, due partly to the frequent occurrence of anaphoric references. This paper describes the text processing which was necessary to identify these anaphors so that they may be utilised or their effects neutralised in the sentence selection criteria.

This work brings together established automatic abstracting techniques with newly developed sentence selection and rejection rules. Not only are there traditional reasons for pursuing this work (i.e., to reduce human costs and to speed up information dissemination), but there are also new developments which could benefit. The use of networks for electronic journals and for knowledge dissemination is possibly the key issue for the future. The electronic medium offers sophisticated searching, with browsing and navigation at the full-text level, and with the ability to move within and between articles via hypertext links. The use of automatic abstracting techniques to identify key points and passages in a text may offer a way further to enhance these facilities.

2. Background Research in Automatic Abstracting

Interest in the problem of how to identify 'topic' sentences for abstracting dates from Luhn's (2) influential paper in 1958. Luhn's approach was to score each sentence in a text according to the weights, based on frequency of occurrence, of all the keywords in a sentence. The highest scoring sentences were extracted to produce an abstract. At about the same time, Baxendale (3) drew attention to the strong tendency of topic sentences to appear first, or sometimes last, in a paragraph. These ideas were subsequently taken up by other workers, in particular Edmundson (4), who in 1969 published the results of an experiment to compare the effectiveness of four extracting methods: the keyword, the title, the location and the cue method. The last, which

scores sentences according to the presence of bonus words and stigma words, was found to produce the best result.

Paice (5) later proposed the use of 'indicator constructs' such as "in this paper we show that ...", which introduce statements about the topic, aim or findings of an article. More recently an experiment was conducted to test the effectiveness of abstracts produced using the keyword and indicator phrase methods with respect to the function an abstract purports to serve (Black and Johnson 6). The results highlighted the problem of cohesion in the abstracts. In particular, the presence of dangling anaphoric references resulted in a disjointed, and at worst unintelligible abstract. This is not surprising, seeing that these techniques take no account of the structure of text in the task of identifying sentences for abstracting.

The aim of our research was to obtain a fuller understanding of the problem of cohesion in automatic abstracts. Research at Lancaster during the late 1980s (Paice and Husk 7, Paice 1) focused on the recognition of anaphors (pronouns and demonstratives) using local, i.e., within sentence, contextual information to decide whether potentially anaphoric words were actually being used anaphorically and to resolve or neutralise them in constructing a passage for extraction. 'Anaphora' is often used only to designate pronouns as they operate within the sentence (Allen 8). Our project addressed the problem posed by discourse phenomena in text. Coherent texts comprise sequences of sentences or other linguistic units each with a discernable relation of meaning to its predecessors. In other words, successive sentences either discuss further properties of a real or abstract object, related objects, or events instigated or affected by the objects. Although texts can be quite long, they have a 'cast' of relatively few objects and events. A consequence of this characteristic of text is the use of definite noun phrases (DNP). These are phrases like "the motor" which can refer over long distances. DNPs may involve reference to objects introduced into the discourse by quite different noun phrases ("a Ford car",

"the vehicle" or "the engine" etc). DNPs can also refer back to events, "X bought the purchase".

The outcome of our work to address the problems caused by DNPs in automatic abstracting was the development of grammatical criteria used to identify points in the text where new concepts are introduced. Those sentences which introduce important concepts and do not refer to discourse entities previously mentioned in the text are surely candidates for extraction. Thus we had thrown light on a new criterion for selecting isolated sentences for abstracting.

The principles behind this approach are described in detail in Neal (9). The motivation was to analyse texts to find chains of DNPs and to ascertain how far back in the text one should be expected to look to resolve each DNP. A sentence containing such referring expressions may refer to discourse entities in a previous sentence. Likewise a sentence containing connectives or comparatives may only be interpreted with reference to some previous sentence(s). If such sentences are selected for an abstract they presuppose something that was said in another sentence which may not have been selected. Neal, using the terminology of logic, states that these sentences fail to be propositions[1]. Using this perspective, it may be assumed that the anaphors must be resolved within the boundaries of a proposition: thus the aim was to identify the points in the text where new propositions begin.

For most referring expressions unsatisfied within the extract, the discourse entity referred to (which may itself be an anaphor) lies in the preceding sentence. With a DNP this entity may be a long way off, requiring a special strategy. Neal proposed that if all propositional sentences, which contain no unresolved connectives, anaphors of comparatives, and selected for inclusion

[1] That is, a translation to a classical logical form would include free variables.

in an abstract, then it may be assumed that any DNP in later selected sentences will be resolved. Taking this approach eliminates the need to search backwards for the entity referred to. The outcome was a set of heuristics to identify non-anaphoric noun phrases and to select sentences containing these key concepts for abstracting. A summary of those which form part of the sentence selection or rejection criteria are presented here. Following this, we describe in some detail the text processing which is necessary to exploit the grammatical clues and text structure in abstracting.

3. Sentence Selection Rules

The methodology of the project represents an extension of the **extract and rearrange** methods described above. The system is constructed out of two rule sets. The first of which is a selective tagger and parser derived from a similar approach (O'Shaugnessy (10)). The tagger assigns grammatical 'tags' to each word in the text according its morphological structure using criteria on the kinds of ending (or suffixes) words will take. Since this does not result in an unique interpretation for each word, the parser is used to disambiguate the tags and in the process structures the sequence of these word categories according to a grammar. The second rule set identifies two classes of sentence in the source text for inclusion in the abstract. The sentence selection/rejection rules are devised to make use of and develop techniques which deserve further attention in abstracting, the use of indicator phrases (Paice 5) and clue words (Edmundson 4). Some of the rules specify rhetorical constructs indicating the relative salience of sections of text (conclusions have high salience, references to previous work have low salience and so on). These are mostly concerned with sentence rejection. Other rules rely on logical and linguistic hypothesis about text structure, and exploit more narrowly grammatical criteria to identify points in the text where new concepts are introduced. From an analysis of ten papers from the journal *Nature* Vol 340 comprising of approximately 30,000 words, the authors found that sentences lacking anaphors and not introduced by rhetorical connectives frequently

introduce key information into a discourse. The development of the rules to identify indicator phrases is outlined in Paice (5). These two rule sets, to identify non-anaphoric sentences and to identify sentences containing an indicator phrase are the only sentence *selection* rules used in the system. Further rules, as stated above, are concerned with the *elimination* or *rejection* of sentences.

The sequence of the sentence selection rules is shown below with corresponding lists.

> **CASE 1**. Select a sentence if it contains an indicator phrase. **List 1** presents a sample of phrases recognised by the system. These are defined by structural patterns rather than enumerated as a list of cases. The representation and implementation, based on an adaptation of Definite Clause Grammar (Pereira and Warren 11) rules are described in Black and Johnson (6).

List 1. Indicator phrases

The |_| *objective* | *of* | *this* | *study* | *is ...*
The | *primary* | *aim* | *of* | *the present* | *investigation* | *was ...*
The | *main* | *hypothesis* | *of* | *the* | *research* | *was ...*

The | *procedures* | *introduced* | *in* | *the following* | *study ...*
The | *problem* | *considered* | *by* | *our* | *research ...*
The | *subject field* | *examined* | *in* | *this* | *project ...*
The | *ideas* | *presented* | *here ...*
The | *model* | *outlined* | *below ...*

The | *results* | *of* | *this* | *analysis* | *confirm ...*
The | *findings* | *from* | *our* | *research* | *show ...*

We | *have* | *proved* | *that ...*
We | *may* | *conclude* | *that ...*
We | *have tried to* | *demonstrate* | *that ...*

CASE 2. Reject a sentence if it is introduced by a connective or by an anaphoric prepositional phrase (**List 2**). These sentences are dependent on others in the text and should not be included. This also applies to a connective which occurs before or just after the main verb. For example, the following sentence would be rejected because the connective "however" appears just after the verb indicating that the statement relies on some previous sentence for its full interpretation:

Enhanced activities are, however, most apparent at very low ionic strengths.

List 2. Connectives
also, then, therefore, firstly, secondly, thirdly, even, although, while, first, second, third, finally, consequently, similarly, since, hence, perhaps, even if, however, for example, in all, in contrast, as a result, in conclusion.

CASE 3. Reject a sentence if the subject is an anaphoric pronoun (**List 3**). The following sentence would be rejected because "they" refers to some group of people or a set of results discussed in a previous sentence(s).

They appear to support our hypothesis.

List 3. Anaphoric subject pronouns
he, she, it, they, that, this, those, all, his, her, their.

CASE 4. Reject a sentence if the first conjunct contains an incomplete comparative construction (i.e., missing the comparand which follows "than") (**List 4**). The first sentence given below is rejected because the comparative "greater" suggests a comparand in some earlier sentence; but the second is not rejected since the comparand of generated enzymes, "wild types", is given following "than" :

The yield loss was considerably greater in 1986
Enzymes generated were far more active than wild types under certain conditions

List 4. Comparatives

larger, smaller, shorter, higher, greater, other, another, more, less, further, since.

CASE 5. If a sentence begins with any of the following phrases (**List 5**), then the remainder of the sentence following the phrase must be tested against all the rules for rejection or selection. These phrases cannot be used to resolve anaphors in later sentences and so they are in a sense ignored in the rules. For example, the following sentence starts with a "it ... that" phrase. The remainder of the sentence, "the incremental change of adoption ..." would eventually be selected as non-anaphoric using rule no **9** given below:

It may be remarked that the incremental change of adoption rates were more pronounced in other provinces than in the punjab in the case of almost all the new technologies.

List 5. Non-antecedents

I, we, the author, my, our, it...that, it...to

CASE 6. Reject a sentence if the subject noun phrase begins with an anaphoric quantifier (**List 6**). The following sentence would be rejected because "each modal peak" refers to some previously introduced entity for its full description:

Each modal peak corresponds to a larval instar.

List 6. Anaphoric quantifiers

each, all, no, total

CASE 7. Reject a sentence if it contains the demonstratives "this" or "these" and others (**List 7**) anywhere in the sentence. For example, the following sentence would be rejected because "this" refers to some previously observed event:

> This could be due to inadequate sampling methods.

List 7. Demonstratives etc.

this, these, the same, the above, the following, the former, the latter

CASE 8. Reject a sentence if the subject noun phrase before the main verb is anaphoric. These generally begin with a quantifier or determiner (**some, every, the**) and are anaphoric (e.g.,The pupae gave rise to adults at the end of the 6 month period) **unless** every occurrence of the determiners or demonstratives (**the, that, those**) is justified by a following preposition, **of** (e.g., the rotation of crops).

CASE 9. Otherwise accept sentence. These sentences are those which are non-anaphoric and should introduce key concepts into the text. Thus the idea of chains of anaphoric reference, whereby subsequent sentences rejected by the above rules, refer to these concepts. The following sentence would be considered to be non-anaphoric since the subject nounphrase cannot be rejected by any of the above rules. Subsequent sentences in the text will be expected to refer to these soil samples:

> Soil samples were taken at approximately monthly intervals during November 1985 to June 1987 in an established lucerne field at the Upington Agricultural Research station.

The preliminary analysis of the sentences selected using these definitive rules has begun to suggest how they might be augmented. Tentative statements "perhaps" or "might" may indicate deselection of a sentence. Likewise, verb tense may also indicate deselection. Furthermore, contextual rules, such as those used in GARP (Paice and Husk 7, Paice 1), to reduce the number of false identifications may be introduced. It is expected that future work to extend and improve the rule set will require the use of a success rate analysis to measure the performance of each rule and the expected enhancement to the system from the addition of further rules.

4. The Architecture of the Abstracting System

This approach to sentence selection depends on the ability to recognise anaphoric noun-phrases in a sentence and also any rhetorical structures. Most of the rules can be implemented without recourse to real parsing (The Garp rules (7) to recognise anaphors and connectives clearly show this). However, parsing is necessary for the implementation of rule 8 which requires that DNPs can be recognised. As such, it requires that text is unambiguously tagged to permit noun-phrase parsing.

The architecture defines an implementation of the sentence selection and rejection rules as a series of text filters, using the tagger and parser developed for this purpose. The first filter subjects a text to morphological and lexical analysis, assigning grammatical tags to words. This is referred to as initial tagging. Multiple tag assignments are then disambiguated by partial parsing to identify the noun-phrases required by the abstracting rules. This filter works selectively, only assigning tags where they are required by the sentence selection rules. The important feature of the system is that it is designed to be reasonably fast in operation. The use of a parser to disambiguate tags means that a corpus for statistical analysis is not necessary, as in the stochastic methods (Church 12, DeRose 13, Marken 14). Also, the parser segments the

sentence into phrasal units (in line with O'Shaughnessy 10) rather than relying on a full linguistic analysis with an extensive grammar. This ensures that there is no restriction on the type of sentence structure which the system will attempt to parse, thus for example it will not 'fail' when faced with a 'garden path' sentence, e.g., "The largest rocks during the experiment", where local ambiguity force a parser to backtrack to arrive at a single correct interpretation.

The only manual intervention required is the initial pre-editing of the texts to separate out headings, captions, figures and formulae, and to mark up the start of each new paragraph. This is, in principle, automatable, particularly assuming access to marked up (e.g., SGML) versions of the text. The information is used at a later automated stage to record structural information which may be used in abstracting.

4.1 Initial Word Tagging

4.1.1 The Dictionary

The construction of a dictionary plays an important part in tagging, especially since the closed class words in the dictionary carry a great deal of information about the syntactic structure of a sentence. The initial tag assignment is performed on the basis of a limited dictionary (ca. 300 words) consisting of most function words and some content words (such as all adverbs not ending in "ly" and common verbs "do" "be" and "have"). Exceptions to the morphology rules are included, e.g.,the irregular forms of the nouns "women", "men". This allows for the assumption that all plural nouns and s-forms of verbs can be identified. The dictionary lists all the possible parts-of-speech for each word. For instance, the word "after" has the possible tags preposition, adverb, or adjective.

An extract of the dictionary with its information in the format **word &tag(features,** is shown in **Figure 1**. The features associated with determiners ("ana","non") state whether they form

anaphoric noun-phrases and the second feature ("s","p") state whether the determiner when combined with a noun will form a singular or plural noun-phrase. The features of verbs and auxiliaries ("pres","past","ing") state the tense.

Comparison of text words against the dictionary is performed, a sentence at a time, by a sequential merging process coded in the 'C' language. The words of the sentence are first sorted alphabetically in order to facilitate the look-up process. Afterwards, any word found in the dictionary will have received one or more tags.

```
a &det(non,s
about &adv(_,_
about &prep(_,_
again &adv(_,_
against &prep(_,_
alive &adv(_,_
all &predet(_,_
almost &adv(_,_
did &aux(pres,_
did &v(past,_
do &aux(pres,_
do &v(pres,_
doing &v(ing,_
done &v(past,_
during &prep(_,_
each &det(ana,s
```

Figure 1 (**page 10**). A Dictionary Extract.

4.1.2 The Morphology Analyzer

The majority of content words not listed in the dictionary can be tagged using morphological information about suffixes (usually, **-ment, -ity, -ness** indicate nouns, **-ous, -cal** indicate adjectives and **-ly** adverbs). These, with the associated part-of-speech, are listed in **Figure 2**. Various checks are used to avoid incorrect assignments. In general, the stem must contain at least three letters. For example, only words with more than three letters ending in -s are assigned the associated tag of plural noun or s-form verb. This excludes "bus" and "gas". A

check to ensure that the penultimate letter is not "s" "u" or "i" rules out s-form tagging of "discuss", "surplus" and "analysis". In addition to these rules, a word containing a capital letter is tagged as a likely proper noun.

The program for the recognition of word endings was written in 'C' using the UNIX **LEX** utility for pattern matching.

The default categories of single noun or baseform verb are assigned to any word which does not comply with the morphology rules. Research into lexicon construction has shown that the majority of new words will be nouns, abbreviations or proper names (Amsler (15)). An unknown word may also be an adjective, but since adjectives and nouns occur interchangeably in similar positions in our grammar the information lost by treating adjectives as nouns is not considered to be important in this application.

NOUN(-ness, -ics, -ster, -eer, -izer, -grapher, - loger,
-er*, -al*, -ty, -ory*, -ry, -cy, -ectomy, -fy, -y*, -on,
-ment, -ance, -art, -ic*, -ick*, -igue*, -ism, -hood, -et,
-ship, -age, -encence, -ful*, -ive*, -ard, -or, -ar*,
-tude, -um, -ice, -eme, -ean*, -arian*, -ician, -gram,
-ete, -ia, -ock, -ode, -ome, -ile*, -ot, -ote, -cule,
-cle, -ist, -ade, -ad, -il*, -ese*, -form*, -ine*,
-id*, -nd*, -oid*, -gen, -cide, -th, -ule, -ure, -stat,
-phil*, -phile*, -phobe*, som*, -some*)

ADJECTIVE(-cal, -ble, -lytic, -logic, -genic, -like, -ward,
-lent, -ior, -ular, -an, -ose, -ac, -ant, -esque, -excent,
-ern)

ADVERB(-wards, -ively, -ibly, -fully, -ily, -ically,
-edly, -itive, -ative, -fuge, -wise)

ADJECTIVE & ADVERB(-less, -ways, -way, -ly, -st, -fold)

POSS(-'s)

NOUN(plural) & VERB(sform) (-s)

VERB(edform) (-ed)

VERB(ingform) & NOUN (-ing)

ADJECTIVE & VERB (-ish)

VERB(-ize, -esce)

*may also indicate an adjective

Figure 2. Morphology Information.

The output from this stage is a set of Prolog clauses describing the text in the following form, **con(SN,SP,EP,Word,Category,Feature1,Feature2)** where,

[SN] is the Sentence Number in which the word occurs.

[SP] is the Start Position of the word in the sentence.

[EP] is the End Position of the word in the sentence.

[Word] is the word in question.

[Category] is the assigned category as indicated by the recognised ending of the word, or by the dictionary.

[Feature1] is the tense of a verb or the anaphoric indicator of a determiner.

[Feature2] is the number feature of singular or plural

Example predicates for a sentence are shown below in **Figure 3**.

```
con(8,0,1,developing,v,ing,_).con(8,0,1,developing,n,_,s).
con(8,1,2,countries,n,_,p).con(8,1,2,countries,v,pres,_).
con(8,2,3,today,n,_,s).
con(8,3,4,do,aux,pres,_).con(8,3,4,do,v,pres,_).
con(8,4,5,not,adv,_,_).
con(8,5,6,have,aux,past,_).con(8,5,6,have,v,pres,s).
con(8,6,7,a,det,non,s).
con(8,7,8,world,n,_,s).con(8,7,8,world,v,pres,s).
con(8,8,9,of,prep,_,_).
con(8,9,10,resources,n,_,p).con(8,9,10,resources,v,pres,_).
con(8,10,11,to,adv,_,_).con(8,10,11,to,aux,_,_).con(8,10,11,to,prep,_,_)
con(8,11,12,freely,adj,_,_).con(8,11,12,freely,adv,_,_).
```

```
con(8,12,13,exploit,n,_,s).con(8,12,13,exploit,v,pres,s).
con(8,13,14,and,coord,_,_).
con(8,14,15,a,det,non,s).
con(8,15,16,few,det,non,p).
con(8,16,17,are,aux,ing,_).con(8,16,17,are,v,pres,_).
con(8,17,18,now,adv,_,_).con(8,17,18,now,subord,_,_).
con(8,18,19,beginning,v,ing,_).con(8,18,19,beginning,n,_,s).
con(8,19,20,',',punct,_,_).
con(8,20,21,out,adv,_,_).con(8,20,21,out,prep,_,_).
con(8,21,22,of,prep,_,_).
con(8,22,23,necessity,n,_,s).
con(8,23,24,',',punct,_,_).
con(8,24,25,to,adv,_,_).con(8,24,25,to,aux,_,_).con(8,24,25,to,prep,_,_)
con(8,25,26,look,n,_,s).con(8,25,26,look,v,pres,s).
con(8,26,27,towards,prep,_,_).
con(8,27,28,a,det,non,s).
con(8,28,29,more,adj,_,_).con(8,28,29,more,adv,_,_).con(8,28,29,more,n,_,s)
con(8,29,30,self,n,_,s).con(8,29,30,self,v,pres,s).
con(8,30,31,reliant,adj,_,_).
con(8,31,32,road,n,_,s).
con(8,32,33,to,adv,_,_).con(8,32,33,to,aux,_,_).con(8,32,33,to,prep,_,_)
con(8,33,34,development,n,_,s).
```

Figure 3. Prolog predicates containing tag information.

4.2 Disambiguation by Local Syntactic Context

Clearly this process of initial tagging creates a number of tags which are extremely unlikely in the immediate context. We experimented with the possibility of using a set of heuristic constraint rules to eliminate some of these. These rules comprise a trigger and a consequence. The trigger is the presence of a certain assigned tag and the consequence is the selection from a choice of tags following the trigger by the removal of the unlikely tags. These rules are presented in **Table 1** with an example of the original set of predicates resulting from the morphology and lexicon analysis. These rules state: if a noun-or-verb follows a determiner, retract the verb; if an auxiliary follows an auxiliary-or-verb, retract the verb; if an adjective-or-adverb follows a verb, retract the adjective; if a noun-or-present verb follows a verb-or-nonpresent auxiliary, retract verb and the auxiliary; and finally, if a modal-or-noun follows a

determiner, retract the modal. The italics in the examples indicate the removal of a predicate from the database as a result of the rule.

RULE	EXAMPLE
con(_n,_p1,_p2,_,det,_,_), con(_n,_p2,_p3,_,n,_,_), con(_n,_p2,_p3,_,v,_,_), retract(con(_n,_p2,_p3,_,v,_,_)	con(4,0,1,the,det,ana,_) con(4,1,2,detectors,n,_,p) *con(4,1,2,detectors,v,pres,_)*
con(_n,_p1,_p2,_,aux,_,_) con(_n,_p1,_p2,_,v,_,_) con(_n,_p2,_,_,aux,_,_) retract(con(_n,_p1,_p2,_,v,_,_)	con(1,0,1,it,pron,_,s) con(1,1,2,has,aux,past,_) *con(1,1,2,has,v,pres,s)* con(1,2,3,been,aux,_,_)
con(_n,_p1,_p2,_,v,_,_) con(_n,_p2,_,_,adj,_,_) con(_n,_p2,_,_,adv,_,_) retract(con(_n,_p2,_,_,adj,_,_)	con(1,3,4,suggested,v,past,_) *con(1,4,5,recently,adj,_,_)* con(1,4,5,recently,adv,_,_)
con(_n,_p1,_p2,_,aux,_t,_), not(_t=pres), con(_n,_p1,_p2,_,v,pres,_), con(_n,_p2,_,_,n,_,_), con(_n,_p2,_,_,v,pres,_), retract(con(_n,_p1,_p2,_,aux,_,_)), retract(con(_n,_p2,_,_,v,pres,_))	*con(5,6,7,are,aux,ing,_)* con(5,6,7,are,v,pres,_) con(5,7,8,20cm,n,_,p) *con(5,7,8,20cm,v,pres,s)* con(5,8,9,apart,adv,_,_)
con(_n,_p1,_p2,_,det,_,_) con(_n,_p2,_,_,modal,_,_) con(_n,_p2,_,_,n,_,_) retract(con(_n,_p2,_,_,modal,_,_)	con(2,0,1,the,det,ana,_) *con(2,1,2,will,modal,_,_)* con(2,1,2,will,n,_,s)

Table 1 : Heuristic Constraint Rules

These rules were applied to a text of 470 words: 236 of these words were correctly and unambiguously tagged by the morphology and lexicon. A further 70 words had their tags correctly selected by these constraint rules. This gives a total success rate of 65% and leaves 164 words to be resolved. It is possible to continue developing the rules to deal with more cases. Hindle (16) developed a set of about 350 rules of this type using a corpus of texts and statistical analysis to determine the frequency with which certain categories are likely to occur together in a sentence. However, he reported a success rate of 81%, which meant that nearly 1

out 5 of the ambiguous words are incorrectly disambiguated in any given sentence. SIMPR, a knowledge-based text storage and retrieval system, (Gibb (17)), pre-processes text for automatic indexing using morphological analysis to identify the word tokens, or tags, in text. Using a lexicon considerably larger than ours, of approximately 57000 entries, and approximately 400 rules for context-dependent disambiguation according to the particular location in which each word occurs, it was able to resolve about 95% of the morphological ambiguities. In addition, the rules expressed as a constraint grammar eliminate around 90% of syntactic ambiguities and produces a syntactic representation giving a structure name (such as noun-phrase) for each major groupings of words.

The use of existing tagging and parsing software, such as CLAWS (18), was considered. However, the output of CLAWS, an unstructured sequence of tags, did not appear to suit our requirements for later processing. We only became aware of the contraint grammar parser used in SIMPR, (Karlsson 19-21) once the work reported here had got under way. Our approach was, then, to adapt in-house componenets using the fragments of grammar rules to capture much of what is stated in the heuristic constraint rules described above. In this way, further ambiguity following initial tagging will be resolved during the parsing process. Since the aim was to parse the sentences to identify noun-phrases it was decided to continue the tag disambiguation process using grammar rules, the *local parser*, with an added mechanism to deal with the problems of partial parsing, the *global parser*. The five heuristic constraint rules are retained since they will make subsequent parsing significantly faster.

4.2.1 The Tag Disambiguator

Locally, a bottom-up chart parser is used with a grammar to group words together that are likely to form noun groups or verb groups by exploiting the word order in these groups. Thus boundaries may be identified; for example, a quantifier generally starts a noun group and an

auxiliary initiates a verb group. In this way, unrestricted text can be partially analyzed using the fixed lower level structure of some constituents to disambiguate tags. At a global level, the parser attempts to link a phrasal unit found to earlier units so that clauses can be identified.

A major problem in locating phrase boundaries is encountered when they are not marked by function words. For example, consider the sentence, "*The blue book defines file transfer*" where all the words apart from "the" are possible verbs or nouns. Faced with this sequence of unidentified words, number agreement may be used to decide that "defines" is the verb following a singular np. However, there are always some difficult cases, consider "*the boy adores fish*" and "*the boy scouts fish*". Based on number agreement alone, it is not possible to state when the verb is in the s-form. Likewise ed-forms of verbs may also present problems. Consider, "*the machines scattered papers*" and *"the machine disentangles scattered papers"*. In such cases, it is hoped that the remaining words in the sentence will force the decision. Thus, at present, these undecided cases are dealt with in the global parser.

4.2.2 The Parser

Definite Clause Grammar rules are adapted for use with a bottom-up parser by storing the results on the arcs of a chart. The basic principle of bottom-up parsing is to reduce the words whose categories match the right hand side of a grammar rule to a phrase of the type on the left hand side of the rule. There are several rule invocation strategies for chart parsing. A left corner parsing strategy (Gazdar and Mellish 22) was used which is based on an interaction of data-driven analysis and prediction based on grammar rules. Some state-of-the-art heuristics (cf. Wiren 23) were used to cut the parser's search space roughly by a third. Details of the implementation are recorded in Johnson, Black, Neal and Paice (24).

4.2.3 The Grammar

The left corner chart parsing strategy is used with a predominantly noun-phrase (np) grammar to return a partial analysis of the text. The np grammar can correctly identify nps, especially when they are separated by an auxiliary verb, a common verb (shown in **sentence 1** below) or a determiner which signals the end of a vp (as shown in **sentence 2** below). The nps selected for these sentences are given from their start to end position.

Sentence 1.

0 another 1 important 2 feature 3 of 4 expert 5
systems 6 is 7 their 8 mode 9 of 10 operation 11.

0 6 np(nom(nom(prmod(adj(another,adj(important))),n(feature)),
pmod(pp(of,np(n(expert,n(systems)))))))
7 11 np(poss(poss(ppron(their))),nom(n(mode),
pmod(pp(of,np(n(operation))))))

Sentence 2.

0 this 1 paper 2 considers 3 the 4 need 5 to 6
provide 7 some 8 form 9 of 10 local 11 area 12 network
13 management 14 .

0 2 np(det(this),n(paper))
3 5 np(art(the),n(need))
7 14 np(quant(some),nom(n(form),pmod(pp(of,np(nom(prmod(
adj(local)),n(area,n(network,n(management)))))))))

4.2.4 The Global Parser

The determining of higher-level syntactic structures that link these groups together is difficult, especially when dealing with unrestricted text. The approach taken is to recover the units that occur inbetween the nps initially selected. In **sentence 2** above from positions 2 to 3 there is a verb and from positions 5 to 7 a verbphrase (vp). In the global parser these are acceptable units to occur between a np and so the nps are accepted as correct. Further illustration of the global parsing is shown below to indicate the categories which may appear between two nps. Square brackets are used to indicate the optional presence of a category, e.g., [,]. Notice that the parser is fairly rudimentary. For example, it is not necessary to identify whether a preposition occurs

in or between nps. The parser only does what is necessary in this application and in doing so reduces the search space and thus the time taken.

{np} [,] prep {np} {a primary factor} in {public health}

{np} [,] conj {np} {large numbers of people in the rural areas} and {old quarters of cities}

{np} conj prep {np} {the areas in the rural quarter of the city} and in {the poorer quarters}

{np} relative clause {technologies} which are efficient in the use of local materials

prep {np} [,] {np} By {cosmic ray events}, {the distribution}

{np} vp {np} {the west's technological development} was founded on {the cheap raw materials}

In addition to the global parsing rules, a set of recovery procedures are needed when the group appearing between two nps is not accepted. These are given below and are all performed on the arcs built up during the chart parsing.

{np1} relative {np2}
& **np1** ends with a
past particle --> {the results suggested} that {the larvae}
reduce np1 to {the results} suggested that {the larvae}
recover vp "suggested"

{np1} aux {np2} --> {each packet} may {travel by the same route}
reduce np2 to may travel by {the same route}
recover vp "may travel"

{np1} vp conj {np2} {we} must research and {develop}
-->
recover np2 as **vp** {we} must research and develop

{np1} adverb {the rate of n release depends} essentially
prep {np2} on {the soil temperature}
-->

reduce np1 to {the rate of n release} depends essentially
recover vp on {the soil temperature}

{np1} {np2} {industries depend on selling}{their wares}
-->
reduce np1 to {industries} depend on selling {their wares}
recover vp

pron {np2} it {depends on the rules}
-->
reduce np2 to it depends on {the rules}
recover vp

5. Evaluation of the Parser

The results in **Table 2** were obtained for 310 sentences parsed from test texts, **test A**, which were not used in the development of the parser. Similar results were obtained during earlier experiments, **test B**, over a total of 1200 sentences.

TYPE	NO.OF SENTENCES Test A Test B	PERCENTAGE Test A Test B
ALL CORRECT	135 516	43.3% 43%
CORRECT 1ST NP&VP	124 504	40.2% 42%
INCORRECT	51 180	16.5% 15%

Table 2: Evaluation of the parser

The types of analysis used to obtain these statistics are described below.

The following sentence is an example which was considered to be correctly parsed: *Seeds of both species were germinated on moist filter papers which were soaked in deionized water in a constant temperature box.* The parse results are as follows:

np(0,4,np(ana,p):(np(n(seeds)),pp(of),np(art(both),n(species))))
vp(4,7,vp(_,_) : vp(aux(were),vp(vp(v(germinated)),on))).
np(7,21,np(non,s):((np(nom(prmod(adj(moist,np(n(filter)))),
n(papers))),relnp(rel(which,seq(vp(aux(were),vp(vp(v(soaked)),in)),
np(nom(prmod(part(deionized)),n(water)))))),pp(in),
np(art(a),n(constant,n(temperature,n(box))))).

The following sentence has only its first np and vp correctly parsed (this being adequate for our purposes): *Sprinkler irrigation was provided with the rows configured in such a way that runoff was prevented from contaminating adjacent treatment areas.* Although all the nps and vps were found, the word "such" was tagged as an adjective and not as a predeterminer. The expression "such a way" could not be recognised. This meant that it was unable to find a permissible construction between the nps "the rows configured" and "a way". As a consequence, the relative clause starting "that runoff" could not be joined to the np.

np(0,2,np(_,s):np(n(sprinkler,n(irrigation)))).
vp(2,5,vp(_,_):(vp(aux(was),vp(v(provided))),pp(with))).
np(5,8,np(ana,_):np(art(the),n(rows,pmod(part(configured))))).
vp(15,17,vp(past,_):vp(aux(was),vp(v(prevented)))).
np(18,21,np(_,p):np(n(adjacent,n(treatment,n(areas))))).

unselected(10,12,np(non,s):np(art(a),n(way))).
unselected(13,14,np(_,s):np(n(runoff))).

However, this does demonstrate an advantage of this approach. There are many expressions which may occur in sentences but which may cause difficulties when trying to write a grammar for unrestricted text. For example, along with the example *in such a way* we might also find the expressions, *is some what surprising* or *greater than that of.* The partial parser is able to ignore these expressions, which means that the delimitation of nps would rely on other clues such as a noun-phrase begins with a determiner.

Finally, in some sentences the first noun-phrase or verb-phrase is not correctly identified owing

to restrictions in coverage of the grammar. More compendious grammars exist but the project lacked the resources to assimilate them to its software environment.

Although there is much scope for improvement it was decided that the tag disambiguation method by partial parsing was adequate for this application. Such improvements may be obtained by simply extending the grammar rules. For example, the errors outlined above may be dealt with by including idiomatic phrases in the dictionary (e.g., "more than ever"), and by assigning more tags in the dictionary ("such" tagged as a predeterminer). However, at present the tagger and noun-phrase parser has allowed us to produce abstracts using the sentence selection rules outlined at the start of this paper.

6. Evaluation of the Extracts

This system should, according to the principle, produce abstracts which are cohesive pieces of English and reproduce the sense of the original text. An example abstract produced is given in **Appendix 1,** abstract 1. Alongside this are abstracts produced using a technique which relies on keywords, using Earl's (25) algorithm, (abstract 2) and one which relies on the identification of indicator constructs outlined in Paice (5), (abstract 3), for comparison. The methods for producing these additional abstracts are outlined in Black and Johnson (6).

At a glance, it may be said that our objectives have been met. None of the selected sentences in any of the three abstracts is obviously inappropriate. However, whilst both abstracts 1 and 2 are more informative than abstract 3, abstract 1 is more cohesive than 2. However, it could be argued that abstract 1 is too long, which raises the question, is there a 'correct length' for an abstract? Clearly, there are limits: the abstract should convey more information than the title alone, and it should be shorter than the full text. As a rule of thumb, the length of an abstract of 250-500 words is often stated (Rowley 26). Biological Abstracts, on the other hand, advised its

abstractors to aim at 3-5% of the length of the original text (Batten 27). Thus, although we can give the actual length of the abstracts in the appendix in terms of a percentage of the full text, it is generally accepted amongst abstractors that an abstract does not need to be a specified length but should be long enough to convey the information to allow the abstract to fulfill its function.

There are a number of problems to address when seeking an objective framework for the evaluation of these abstracts. In particular, it is not realistic to base the evaluation on a target set of extracted key sentences from the source text: a given idea might be expressed in two or three different alternative sentences (cf. Edmundson 4) and the whole abstract by many valid alternative subsets of the sentences in the text. We instead propose evaluation in terms of the information conveyed in the selected sentences. A template is created, before looking at the abstracts obtained, which sets out the information found in the text under certain headings: for an example, see **Figure 4**. The scores in Figure 4 are arbitrary values, assigned by the authors, intended to indicate the relative importance of the various ideas. A score of 5 is used for a concept which is assumed to be central to the paper, and which must be mentioned in the abstract. A score of 0 is used for a concept which, although is not necessary, would not appear out of place in the abstract. The assignment of intermediate weights is a rather subjective activity. However, what is important is not the actual scores but the ranking they imply. The plausibility of this scheme used to score the abstracts was tested by composing an abstract by hand, based on the tabulation. This is shown in **Figure 5** with the automatic abstract and an abstract produced by CAB for comparison.

It is important to note that evaluation is not only a matter of information selection. We also need to find some means of evaluating the abstracts in terms of their cohesiveness. In addition, we also found that we had to evaluate the success of the tagger and parser in its use in the sentence selection rules. As stated above, with limited resources this system is obviously rather

rudimentary.

Despite these problems, a preliminary analysis of one abstract is demonstrated below. To indicate the success of our prime objective of producing a coherent piece of text, anaphoric references are categorised in the abstract as follows. If apparently anaphoric expressions occur which are considered to be acceptable, they are marked by italics. If the anaphoric expression is resolved by other sentences included in the abstract, they are marked by bold type and a subscript marks the sentence number in which the reference is found. Finally, as in sentence 35, any unresolved anaphor would be marked in capitals. Sentence 35 was selected for its indicator phrase: but the expression referred to, "*the results*", is provided in the previous sentence, "*Since recovery of first-instar larvae in field collected samples was unsatisfactory, the head-capsules of 20 first-instar larvae, hatched in the laboratory, were measured*". Unfortunately this sentence was rejected. Our rules include "since" as a connective, although it is used here as an **intra** sentence connective. This highlights our need to develop and refine our rules based on the results.

The abstract is then scored against the template, as shown in the column headed 'extract' in **Figure 4**. The square brackets show the sentence numbers from the abstract given below. This abstract only gets a score of 16, including the title which covers idea 9. This low score could be shown in a better light if it is considered that an abstract for this text from CAB Abstracts, shown in **Figure 5**, scores 23 out of a possible 30. Due to the arbitrary nature of the scores, the exact numerical totals are not intended to be taken too seriously. For example, idea 4, 'pest of Lucerne', is not explicitly stated in the abstract, only implied and therefore is assigned a score of 2 instead of 4. The main point is that where the highly scored ideas are not included then the abstract is penalised accordingly. In this example abstract, findings are almost unrepresented. In the template, it was considered that idea 17 referring to the findings was especially worth

reporting. This does not appear in the abstract, but is included in the CAB abstract which again scores higher.

Text: "Some aspects of the biology of the white-fringed beetle, in the Lower Orange River irrigation area of South Africa."

NO	Topic:-		score	extract	CAB
1.	Subject sp. is:	White-fringed Beetle	5	5 [1]	5
		(or 'beetle' 1)	–	–	–
2.	Origin of sp. is:	S.America	0	0 [1]	–
3.	Incidence of sp. is:	E.US, SE.Australia etc	0	0 [1]	–
4.	Role of sp. is:	pest of Lucerne	4	–	4
		(or 'pest'or 'lucerne' 2)	–	2 [1]	–
5.	Parts damaged are:	roots	1	–	1
		underground stems	0	–	–
6.	Stage of 1. causing 5:	larva	1	–	1
	Aim:-				
7.	Purpose of study is:	biology of 1.	1	1 [20]	1
8.	Stage of focus is:	larva	0	–	0
	Setting:-				
9.	Locality of study:	LOR irrign. area of S.Africa	3	[3]title	3
		(or Lower Orange River 1)			
		(or 'S.Africa 1)			
	Methods:-				
10.	General method is:	survey	0	–	0
11.	number of localities:	several	0	–	0
12.	specific methods:	soil sampling in lucerne fields	2	2 [25]	–
		(or just 'soil sampling' 1)			
		count & sort larvae	0	0 [26]	–
13.	Measurements:	head capsule widths of larvae	2	2 [41]	1
14.	Analysis method:	probit analysis	0	0 [41]	–
	Findings:-				
15.	Geographical distribn.	recent eastward spread	0	–	–
16.	Infestation rates	highest in central & east			
		parts of region	0	–	–
17.	No. of larval instars:	7	3	–	3
18.	Life cycle period	12-15 months	2	–	2
19.	Larval period	9-12 months	2	–	2
20.	Maturation of larvae:	faster at higher temperature	2		
		(or affected by temp. 1)	–	1[53]	–
21.	Peak populations:	February	0		
22.	Distribn. in soil:	mostly in top 300 mm.	0		
		depths down to 750 mm.	1	–	–
		disagrees with earlier report	1	–	–
	TOTALS		30	16	23

Figure 4. Evaluation of the abstracts

7. Conclusion

This paper has described an enhanced sentence selection method for automatic abstracting. These rules rely on grammatical criteria to identify desirable isolated sentences to include in an abstract. A simple system, based on the limited resources of a dictionary, morphological analyser and noun-phrase parser, is used to satisfy this requirement. The advantage of using a partial grammar and a chart parser for simple recovery procedures means that no restrictions are placed on the text handled.

The results suggest that this work may be a step in the right direction for automatic abstracting. However, much remains to be done. The output from our program is far from perfect and our sentence selection rules need to be refined to produce shorter, more acceptable abstracts. We have identified the need to extend the dictionary, particularly to recognise idiomatic phrases, and the need to refine the parsing rules. At present, the system is rather rudimentary, designed to be fast in its operation while allowing us to explore various automatic abstracting techniques. We have been encouraged by the results of the sentence selection rules outlined in this paper. The main drawback is that the abstracts produced are too long, although this could be helped by use of alternative sentence selection criteria. Positional criteria may be employed to eliminate sentences which occur in the middle of the text or paragraphs (Baxendale 3; Edmundson 4).

It is not generally sufficient to concatenate a set of isolated key sentences from a text. An understanding of the structure of texts and how they are organised beyond the level of the sentence must be utilised in the process. After all, the author of the text will have endeavoured

to use the structure to help convey meaning and to ensure that key concepts are introduced at appropriate places.

8. Future Work.

Further understanding of rhetorical structure theory and text grammars (e.g., Mann & Thompson 28, Sillince 29) may provide a way of analysing text according to the way in which the meaning is organised to convey some kind of message. Ideally, it may provide a means to allow us to keep track of the relationships between a text's propositions **and** to determine the relative importance of the sentences concerned (ideas along these lines have been expressed by Paice (30)). Integrating this work into that of automatic abstracting may enable us to further our ultimate goal of producing coherent and useful abstracts.

MODEL Abstract.
This paper concerns the White-fringed Beetle, G.leucoloma, a pest whose larvae cause damage to the roots of lucerne. A study of the biology of this insect was carried out in the Lower Orange River irrigation area of South Africa. Soil samples were taken, and head-capsule widths of larvae were measured. Seven larval instars were found to occur. The total life cycle took 12-15 months. The larvae matured in 9-12 months: the period was shorter at warmer seasons of the year. Larvae occurred down to 750mm below the soil surface, in disagreement with an earlier report.
(**11%** of the full text)

CAB Abstract.
The biology of Graphognathus leucoloma was studied in the Lower Orange River irrigation area of South Africa in 1985. Information is presented on its geographic distribution within the region, number and size of larval instars, and phenology. Larvae caused severe damage to the roots of lucerne throughout the region. During its life cycle of 12 to 15 months, 7 larval instars were present over a period of 9-12 months.
(**6.3%** of the full text)

AUTOMATIC Abstract
Note/ Sentences marked with '?' could be excluded using further criteria which have been considered for the development of the system. These are sentences which begin with a verbal noun (e.g., "readings") or a relational noun (e.g., "yields", "measurements") which assume a relation with some previously mentioned entity (e.g., "measurements **of** larvae size"). Sentences marked with 'i' are selected on the basis that they contain an indicator construct. The only occurrence of unresolved anaphora is in sentence 35. In the text "the results" refers to the measurement of the head-capsule

width of first-instar larvae hatched in the laboratory.

1 the white fringed beetle, graphognathus leucoloma, a south american insect, is an established pest of pastures and crops in *the* eastern united states, south eastern australia, new zealand and south africa.
10? reproduction is parthenogenetic and only females are known.
15? pupation takes place in **the**$_{1:crops}$ upper soil layers from where **the**$_{15:pupation}$ adults make their way to the soil surface.
20i in *this* paper results of *our* investigations on *the* biology of **this**$_{1:beetle}$ insect are reported.
24 a single survey was conducted during september 1985 in established lucerne at seven localities in *the* lower orange river irrigation area.
25 five soil samples were taken at random in lucerne fields at **each**$_{24}$ locality.
26? larvae were sorted from **the**$_{25}$ soil samples by hand and stored in 70 percent ethyl alcohol.
28 soil samples were taken at approximately monthly intervals during November 1985 to June 1987 in an established lucerne field at *the* Upington Agricultural Research Station.
30? larvae and pupae were removed from **the**$_{28}$ soil samples by hand and stored in 70 % ethyl alcohol.
31? adults were sampled in 50 pit traps in **the**$_{28}$ soil which were placed at random in **this**$_{28}$ lucerne field.
35i **THE** results were subjected to probit analysis.
39 the highest percentage rate of infestation occurred in the central parts of **the**$_{24}$ region.
41 as instar sizes frequently overlap, probit analysis was used to calculate instar head capsule size using the method of frampton.
53 temperature appears to play an important role in the duration of especially **the**$_{20:biology}$ pupal and adult stages.
(**22%** of the full text)

Figure 5 (page 23): Abstracts.

Citations

(1) PAICE, C.D. Constructing literature abstracts by computer: techniques and prospects. *Information Processing and Management*, 26(1) 1990, 171-186.

(2) LUHN, H.P. The automatic creation of literature abstracts. *I.B.M. Journal of Research and Development*, 2(2) 1958, 159-165.

(3) BAXENDALE, P.B. Man-made index for technical literature - an experiment. *I.B.M. Journal of Research and Development*, 2(4) 1958, 354-361.

(4) EDMUNDSON, H.P. New methods in automatic abstracting. *Journal of the Association of Computing Machinery*, 16(2) 1969, 264-285.

(5) PAICE, C.D. The automatic generation of literature abstracts: an approach based on the identification of self-indicating phrases. *In*: R.N.Oddy, S.E Robertson, C,J van Rijsbergen and P.W Williams, (*eds). Information Retrieval Research*. Butterworths, 1981, 172-191.

(6) BLACK, W.J AND F.C JOHNSON. A practical evaluation of two rule-based automatic

abstracting techniques. *Expert Systems for Information Management*, 1(3) 1992, 159-177.

(7) PAICE, C.D AND G.D HUSK. Towards the automatic recognition of anaphoric features in English text: the impersonal pronoun it. *Computer Speech and Language*, 1(3) 1987, 109-132.

(8) ALLEN, J. *Natural language understanding*. Menlo Park: Benjamin/Cummings Publishing, 1987.

(9) NEAL, A.P. A tool for the syntactic resolution of anaphora. *In:* K. Jones *(ed). The Structuring of Information.* Proceedings of Informatics 11. London: Aslib, 1991, 27-36.

(10) O'SHAUGHNESSY, D.O. Parsing with a small dictionary for applications such as text to speech. *Computational Linguistics*, 15(2) 1989, 97-109.

(11) PEREIRA, F.C.N AND D.H.D WARREN. Definite clause grammars for language analysis - a survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence*, 13 1980, 231-278.

(12) CHURCH, K. A stochastic parts program and NP parser for unrestricted text *Proceedings of the second Association of Computational Linguistics conference on Applied Natural Language Processing*. 1988, 136-144.

(13) DEROSE, S.J. Grammatical category disambiguation by statistical optimization. *Computational Linguistics*, 14(1) 1988, 31-39.

(14) MARKEN, C.G. Parsing the LOB corpus. *Association of Computational Linguistics Annual Meeting*, 1990, 243-251.

(15) AMSLER, R.A. Research toward the development of a lexical knowledge base for natural language processing. *In*: N.J. Belkin and C.J. Van Rijsbergen, (*eds*). *Proceedings of the 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Cambridge MA, New York: ACM, 1989, 242-249.

(16) HINDLE, D. Acquiring disambiguation rules from text. *Association of Computational Linguistics Annual Meeting,* 1989, 118-125.

(17) GIBB, F ********** *Journal of Document and Text Management* 1(2), 1993, **

(18) GARSIDE, R. The CLAWS word tagging system. *In*: R. Garside, G. Leech and G. Sampson (*eds*). *A computational analysis of English: A corpus-based approach*. London: Longman, 1987.

(19) KARLSSON, F. Constraint Grammar as a Framework for Parsing Running Text. *In*: Hans Karlgren (*ed*). *Proceedings of the XIIIth International Conference on Computational Linguistics*, Vol 3, Helsinki 1990, 168-173.

(20) KARLSSON, F., VOUTILAINEN, A., HEIKILLA, J. and ANTTIL, A. *Natural Language Processing for Information Retrieval Purposes*. Helsinki: Research Unit for Computational Linguistics, 1990, (SIMPR-RUCL-1990-13.4e).

(21) KARLSSON, F., VOUTILAINEN, A., ANTTILA, A. and HEIKILLA, J. Constraint Grammar: a Language-Independent System for Parsing Unrestricted Text, with an Application to English. *In*: *Workshop Notes from the Ninth National Conference on Artificial Intelligence (AAAI-91)*, California: American Association for Artificial Intelligence, July 15, 1991.

(22) GAZDAR, G. AND C. MELLISH. *Natural language processing in Prolog: An introduction to computational linguistics*. Wokingham: Addison-Wesley, 1989.

(23) WIREN, M. A comparison of rule-invocation strategies in context-free chart parsing. *Association of Computational Linguistics Proceedings, Third European Conference*, 1987, 226-235.

(24) JOHNSON, F.C., BLACK, W.J., NEAL, A.P. AND C.D. PAICE. Development and evaluation of a parser for use in an automatic abstracting system. *British Library Abstracting Report*, no. 3, May 1992.

(25) EARL, L.L. Experiments in automatic abstracting and indexing. *Information Storage and Retrieval*, 6(4) 1970, 313-334

(26) ROWLEY, J. *Abstracting and indexing*. London:Clive Bingley, 1992, p.14.

(27) BATTEN, W.E (ed). *Handbook of special libraries and information work*. London:ASLIB, 1975, p.131.

(28) MANN, W.C AND S.A THOMPSON. Rhetorical structure theory: A theory of text organisation. *ISI Reprint Series ISI/RS-87-190*, Information Science Institute, 1987.

(29) SILLINCE, J.A.A. Argumentation-based indexing for information retrieval from learned articles. *Journal of Documentation*, 48(4) 1992, 387-406.

(30) PAICE, C.D. The rhetorical structure of expository texts. *In:* K. Jones *ed*. *The Structuring of Information.* Proceedings of Informatics 11. London: Aslib, 1991, 1-25.

Appendix 1

These abstracts are taken from the same paper to illustrate the results of using three different techniques for sentence selection. The first is produced using the sentence selection rules described in this paper; the second is produced using Earl's (20) keyword technique; and the third is produced using the identification of Paice's (5) indicator constructs. The original article contained 107 sentences and each abstract is expressed as a percentage of this length.

1. British Library project for Abstracting technique.

Developing countries today do not have a world of resources to freely exploit and a few are now beginning, out of necessity, to look towards a more self reliant road to development. This article deals particularly with the indigenous technologies of cooling, using largely natural sources of energy and techniques which have been developed by people locally. The supply of safe drinking water is a primary factor in the maintenance of public health in developing countries. Consideration must be given not only to the water source and its quality but also to the distribution and storage systems. Nile water and water from irrigation channels is unfit for drinking and often carries dangerous pathogens such as bilharzia larvae. Drinking water is usually scooped out of the pot with a dipper, though it was discovered that water collected at the base after it had been filtered through the pot is much cleaner. An experiment was set up using portable meteorological testing equipment in order to evaluate the cooling action of the maziara. Water samples were taken at various stages in the system, to be measured later in the laboratory for purity. Over a 16 hour test period a single jar produced 1700 k cal of cooling. Samples were taken from the river source and from the effluent runoff after water had been allowed to filter through the maziara system. Samples were tested in the government laboratories in the luxor hospital and it was found that the filtered outflow water was pure to the government's drinking water standards, even though the original nile water that was put into the jar was contaminated. The result of the purification tests illustrates that chances of drinking water contamination can be reduced if the maziara's filtering action is used. Technological sophistication is usually measured in terms of the number of transistors or moving parts. If

we evaluate sophistication in terms of efficiency we find the opposite. The hazards of modern air conditioning systems are rarely advertised in the glossy brochures distributed by companies dealers in the third world. Mild shocks sometimes occurs at the entry of an excessively cooled building, if the temperature differences between the inside and outside are too great. Comparative experiments are currently being planned by the authors in Iran, in the use of water jars for air cooling within buildings as against mechanical cooling. In Iran, wind shafts often lead to basement water cisterns. A domestic cooler was developed using a porous compartment to hold the food. This article has dealt with some of the technological innovations that have grown out of an indigenous scientific approach to a basic problem cooling in many third world countries. (**18.7%**)

2. Keyword Method

The maziara is a traditional water cooling and purification system used in rural areas of upper Egypt. As the air becomes drier more water evaporates from the water jar's surface and the cooling rate increases. The hazards of modern air conditioning systems are rarely advertised in the glossy brochures distributed by companies' dealers in the third world. Comparative experiments are currently being planned by the authors in Iran, in the use of water jars for air cooling within buildings as against mechanical cooling. This article has dealt with some of the technological innovations that have grown out of an indigenous scientific approach to a basic problem cooling in many third world countries. (**4.7%**)

3. Indicator Method

This article deals particularly with the indigenous technologies of cooling, using largely natural sources of energy and techniques which have been developed by people locally. The result of the purification tests illustrates that chances of drinking water contamination can be reduced if the maziara's filtering action is used. This article has dealt with some of the technological innovations that have grown out of an indigenous scientific approach to a basic problem cooling in many third world countries. (**2.8%**)