The Automatic Generation of Templates for Automatic Abstracting

Michael P. Oakes

Computing Department, Lancaster University
Lancaster, England

Chris. D. Paice

Computing Department, Lancaster University
Lancaster, England

Abstract

Our goal is the automatic abstraction of journal articles, initially in the field of crop protection. We build a set of templates against which the original text is compared. The templates are designed so that they match the text at points of high information content, where inferences can be made about which expressions best reflect the content of the document. Strings found by matching templates are assigned roles specific to each template. These roles correspond to slots in a frame which is used to represent the document as a whole. An abstract is generated which contains the concept-strings selected from the text.

1 Background

An abstract may be defined as a concise expression of the central subject matter of a text, in particular of a research paper. Two classes of abstract are commonly recognised, namely *indicative* and *informative* abstracts. An indicative abstract is used to help a literature searcher to decide whether the full document may be worth reading, whereas an informative abstract attempts to substitute for the full document by including the main findings and conclusions.

Two traditional approaches to automatic abstracting are:

1. *Extraction*, whereby specific sentences are selected from the source text according to some assessment of their importance. Importance indicators include the concentration of topic-relevant terms (these are terms occurring at high frequency through the text, or occurring in titles or captions); the occurrence of focussing terms and expressions, such as “important”, “clearly”, “to sum up” etc.; and the position of the sentence within the text. This approach is exemplified by Pollock & Zamora's ADAM system [1], and was reviewed in [2].

   The problems with this approach are that importance clues are often not reliable, and that the extracted sentences do not always constitute a coherent text, since they often contain dangling anaphors and other cross-references.

2. *Summarisation*, whereby detailed semantic analysis is applied to the text, and a representation such as a conceptual dependency graph or a semantic net is produced, from which a summary is then generated. An example of this approach is Rau's SCISOR system [3]. This approach requires a very large knowledge base, is rather slow in operation, and tends to be domain specific.

The research described here relies on an alternative approach, known as *concept-based abstracting* (CBA), whose background is described in detail by Paice & Jones [4] and Jones [5]. With this approach, abstracts are produced in three stages: (i) selection from the text of a collection of strings which may contain key ideas; (ii) selection from among these candidate strings of specific

names which are associated with relevant semantic roles; and (iii) generation of an abstract containing all the selected concept names.

Stage 1: Selection of Candidate Strings

The selection of candidate strings takes advantage of the stylistic and semantic regularity found in typical research articles. The stylistic regularity is exemplified by the common use of expressions such as "the effect of X on Y", which suggests that X is an independent variable or influence, while Y is a dependent variable or property. Semantic regularity is seen where the main concepts of the paper fit into a largely predictable range of roles. In the domain of agriculture, these roles might be species (SPE), cultivar (CV), high level property (HLP), low level property (LLP), pest (PES), agent (AGEN), influence (INF), location (LOC), time (WHE) and soil (SOI). Similar roles are described in conjunction with the PLEXUS system of Vickery & Brooks, which was designed for answering questions in the domain of gardening [6].

Both stylistic and semantic regularity may be captured by contextual patterns or *templates*, which may be created by the manual analysis of a domain text corpus, or automatically as described in this paper. The templates we use are alternating sequences of literals (compulsory word sequences) and fillers (any sequence of words found before, between or after literals). The templates thus represent possible contexts of use of the semantically interesting fillers. They are compared with text sentences, and whenever a match is found, the matching sequences become candidates for roles described in the text.

For example, the template "effect on ? of ?" might have the interpretation "effect on HLP of PES". When this pattern is compared against the text, it will match such phrases as "effect on yield of potato leafroll virus". This will provide evidence that "yield" is the high level property (HLP) central to the paper, and that "potato leafroll virus" is the pest.

Some templates have more than one corresponding interpretation. For example, the phrase “effect on soybean of excessive rainfall” is correctly interpreted by “effect on SPE of INF”. Also, even single-meaning templates can often pick up useless strings. For instance, we may have a template “SPE plants”, which correctly interprets the first two words of “Soybean plants were harvested...”, but produces nonsense if applied to “The late-harvested plants were....”. These problems are dealt with at the second stage of the CBA process.

The selected strings may be weighted, since some templates provide stronger evidence for a certain role or slot filler than others. In this paper we will discuss empirical methods for weighting templates. Fillers, particularly those occurring at the left or right of a pattern, can be truncated by the use of stop words as cut-off points; for example the right filler "Spunta, which was planted in Cyprus", becomes simply "Spunta" (the name of a potato cultivar) if "which" is in the stoplist.

Stage 2: Selection of Concept Names

When the template matching stage is complete, for each conceptual role there will be a list of candidate strings. From these, we must select one or more appropriate names to denote each relevant concept. The method used in the present work is to look for recurring substrings of these candidates which have a high aggregate weight. Suppose, for example, that the candidate strings for the role "SPE" (crop species) are "potato tuber", with weight 0.4, and "seed potato", weight 0.3. These two candidate strings yield the following substrings:

potato tuber (0.4)

potato (0.4)

tuber (0.4)

seed potato (0.3)

seed (0.3)

potato (0.3)

The aggregate weight for "potato" is (0.3 + 0.4 = 0.7), so this is the most highly weighted substring, and hence the most likely candidate for the role "SPE".

Output of our template matching program is a list for each role of the most likely candidates for that role, arranged in decreasing order of a) weight, b) length in words and c) number of occurrences. A sample extract of the final output showing the three most highly weighted candidate strings for the role of influence (INF) is shown in Figure 1.

INF			
Weight	**Length**	**Occurrences**	**Candidate string**
11.81	1	18	Temperature
5.48	2	7	Leaf canopy
4.25	1	5	Photoperiod

Figure 1: Sample Extract of the Final Output

There are often two or more relevant fillers for a given role; for example, a paper entitled “The effect of late rainfall and early frost on barley and winter wheat” in concerned with two influences and two crop species. A decision must therefore be made as to how many candidate strings per role should be accepted.

The selection of concept names uses the following criteria, each with an adjustable threshold:

a) the candidate should be one of the top *n* weighted strings for that role.

b) the weight of the candidate must be a given ratio of the weight of the highest weighted candidate for that role.

c) The weight of the candidate should be above a certain threshold.

Stage 3: Generation of Abstracts

A program for the generation of abstracts given the set of instantiated role slots is described by Jones & Paice [7]; the precise details are not relevant here.

2 Method of Creating Templates Automatically

Paice & Jones [4] created templates by the manual analysis of a domain-specific corpus. In this paper, we describe a method for the automatic generation and weighting of the templates. This relies on the use of a training corpus and a small domain thesaurus. Our training corpus consisted of 50 journal articles in the domain of crop protection; an excerpt from one of the articles is shown in Figure 2. The thesaurus contains a list of names of domain concepts, each associated with the appropriate role indicator; some entries from this thesaurus are shown in Figure 3.

> Effects of potato leafroll virus on the crop processes leading to tuber yield in potato cultivars which differ in tolerance of infection. Production of crop dry matter can be analysed as the amount of light intercepted and the efficiency with which intercepted light is used. Yield of a particular organ or tissue is the result of a third process, partitioning of assimilates, and can be conveniently measured as the ratio of the dry weight of the harvestable component to the total plant dry weight. With the potato crop a fourth process, the change in dry matter content of the tubers is also important. Climate, pests and disease affect tuber yield by influencing one or more of these four crop processes, which have been described in a simple model of potato by MacKerron & Waister (1985).

Figure 2: Excerpt from the Original Training Corpus

Automatic thesaurus generation proceeds through a series of four stages, as follows:

1. The thesaurus is employed to replace all the specific terms in the collection with their role identifiers. For example, given that one entry in the thesaurus is “midge|PES|”, meaning that “midge” belongs to the concept-class “pest”, every occurrence of "midge" in the test collection is replaced by the role identifier "PES". This results in a ‘training corpus with roles’, as illustrated in Figure 4.
2. At this stage, frequently occurring sequences of words and roles are selected from the training corpus with roles. Only sequences which contain at least one role, and which do not begin or end with stopwords, are selected. These sequences define contexts in which the various role indicators are found often to occur.
3. The selected sequences are then rearranged to give a set of unweighted templates. For example, suppose that a common sequence is found, such as "effect of LLP on HLP", where LLP means low level property and HLP means high level property. The positions of the generic categories HLP and LLP are replaced by template fillers, denoted "?". The generic categories LLP and HLP then become the interpretations of these template fillers, so the template becomes: “effect of ? on ? /1: –,LLP,–,HLP /1” (the “/1” tags indicate that this is an unweighted template).
4. The unweighted templates are now applied again to the original test corpus to find all matches, including "incorrect" matches in which the filler is not a role indicator. The weight of each template element is then determined by the proportion of times the filler matches the interpretation in an automatic evaluation procedure.

The thesaurus employed in this study was hand crafted, and consists mainly of entries in a list of words and phrases which occur at least ten times in the original training corpus, and were felt to be possible candidates for one of the roles. The word and phrase frequency lists were obtained using Scott's WordSmith package [8]. It would also be possible to use an existing thesaurus converted to the format shown in Figure 3.

ethirimol|AGEN|	tridemorph|AGEN|
King Edward|CV|	Troy|CV|
growth|HLP|	yield|HLP|
drought|INF|	cold|INF|
greenhouse|LAB|	field|LAB|
leaf area|LLP|	weight|LLP|
Tasmania|LOC|	Saskatchewan|LOC|
larvae|PES|	midge|PES|
loam|SOI|	peat|SOI|
wheat|SPE|	peanut|SPE|
1997|WHE|	spring|WHE|

Figure 3: Excerpt from the Thesaurus

Effects of SPE leafroll PES on the crop processes leading to tuber HLP in SPE cultivars which differ in tolerance of infection. Production of crop dry matter can be analysed as the amount of INF intercepted and the efficiency with which intercepted INF is used. HLP of a particular organ or tissue is the result of a third process, partitioning of assimilates, and can be conveniently measured as the ratio of the dry LLP of the harvestable component to the total plant dry LLP. With the SPE crop a fourth process, the change in dry matter content of the tubers is also important. Climate, pests and disease affect tuber HLP by influencing one or more of these four crop processes, which have been described in a simple model of SPE by MacKerron & Waister (1985).

Figure 4: Excerpt from the Training Corpus with Roles

```
window length = 4

STOP 6 "and foliar treatment AGEN"
5 "foliar treatment AGEN +"
5 "treatment AGEN + AGEN"
4 "effect of mildew AGEN"
3 "AGEN gave a significant"
2 "AGEN was the most"
2 "AGEN at different sowing"
2 "AGEN increased fertile tillers"
LOW-FQ 1 "effect of AGEN sprays"
```

Figure 5: Repeated Sequences found by the Concordancing Program, with their Frequencies

Note that for our method to work, it is not necessary for the thesaurus to be comprehensive - the aim is simply that most occurrences of each role in the initial corpus should be replaced by the role indicator.

The output from our concordancing program is a list of unweighted templates. Each template is derived from phrases occurring more than once in the training corpus which contain a role indicator, are of length 2, 3 or 4 words, and do not start or end with a word in a stoplist. This stoplist consists of common function words and punctuation characters. Examples of common word sequences found by the concordancing program are shown in Figure 5.

In Figure 5, AGEN denotes a chemical agent. The sequence marked STOP will not be used to form a template since it begins or ends with the stopword "and"; the sequence marked LOW-FQ will not be used to form a template as it does not occur frequently enough. The retained sequences are in the form of alternating generic categories ("fillers") and "literals". A sequence such as "literal GENERIC literal" is converted to the form

```
literal ? literal /1:
        –,GENERIC,– /1.
```

where "/1" denotes an unweighted template. The two "–" notations on the second line means that no interpretation is given to the literal components of the template. The set of unweighted templates which are derived from the word sequences shown in Figure 5 is shown in Figure 6.

```
foliar treatment ? + /1:
        –,AGEN,–/1.
treatment ? + ? /1:
        –,AGEN,–,AGEN/1.
effect of mildew ? /1:
        –,AGEN/1.
? gave a significant /1:
        AGEN,–/1.
? was the most /1:
        AGEN,–/1.
? at different sowing /1:
        AGEN,–/1.
? increased fertile tillers /1:
        AGEN,–/1.
```

Figure 6: Excerpt from the File of Unweighted Templates

The templates are then ordered according to a) number of components (i.e., literals and fillers), b) number of characters, and c) alphabetical order. The purpose of this ordering is so that when template matching is performed against the corpus later, the longest templates are tested first. If a match is found, shorter templates which may be subsets of the longer template just matched are not tested.

3 Template Evaluation and Weighting

The template weighting program reads in the version of the training corpus with roles, and the templates are matched against this text. To illustrate the template evaluation procedure, consider the unweighted template

```
spring ? was sown/1:
        –,SPE,–/1.
```

This would match the following sections of text, with the indicated judgements on the quality of the match:

a) “spring SPE was sown” : CORRECT

b) “spring and summer SPE was sown” : PARTIAL

c) “spring was sown” : NULL

d) “spring CV was sown” : INCORRECT

e) “spring field was sown” : INCORRECT

In example a), the filler "SPE" matches the interpretation "SPE", so the match is judged correct. In b), the filler is "and summer SPE", resulting in a partial match. In c) the filler corresponds to no text at all, so the match is null. In d) and e) the fillers "CV" and "field" do not correspond at all to the interpretation "SPE".

If x is the total number of matches found for the template (correct, partial, null or incorrect), and y is the number of matches which were either correct or partial, then the weight for that template with the given interpretation is y / x. The weighted template is stored in the following form:

```
spring ? was sown /x:
        –,SPE,–/y.
```

This format allows several interpretations of the same template to be represented. For example, in the template

```
average ? /74:
        –,HLP/4;
        –,INF/7;
        –,LLP/15.
```

the word or words following "average" have three possible interpretations: HLP with a weight of 4/74, INF with weight 7/74 and LLP with weight 15/74. Further examples of weighted templates are shown in Figure 7.

4 Comparison of Manual and Automatic Templates

In this pilot study, a set of 11 role indicators were found for each of four different articles in the domain of crop protection, firstly using the set of templates manually produced by Paul Jones [5], then repeating the experiment using a set of templates generated automatically from a training set of about 50 articles, also in the domain of crop protection. The resulting sets of role fillers were compared against the manually adjudged content of the original article.

```
? and distribution /8:
        LLP,–/7.

? application /50:
        AGEN,–/13.

? are favourable /2:
        INF,–/2.

average seed ? /4:
        –,LLP/3.

foliar treatment ? + /6:
        –,AGEN,–/6.

seeding ? increased /5:
        –,LLP,–/3.

? to high ? /4:
        INF,–,–/4;
        –,–,INF/4.
```

Figure 7: Excerpt from the File of Weighted Templates

Number of articles = 4	
Manually-generated templates	**Automatically-generated templates**
Number of template matches 27	Number of template matches 30
Fully correct 5 (18.5 %)	Fully correct 9 (30.0 %)
Partially correct 8 (26.7 %)	Partially correct 8 (26.7 %)
Wrong 13 (59.3 %)	Wrong 13 (43.3 %)
Null matches 1	Null matches 11

Table 1: Results comparing Manual and Automatic Templates

In this evaluation, only the highest weighted candidate for each role filler was considered. An assessment of "fully correct" was given if the program-generated role filler matched the humanly judged role filler exactly, while "partially correct" was assigned to cases where the program generated role filler was a substring of the humanly assigned one. For example, a program generated role filler of "infection" was considered "partially correct" if the human judge had decided that the role filler for INF should be "viral infection". This simple evaluation ignored the possibility that there might be more than one valid identifier for a particular role in a given document, such as both "Marnoo" and "Wesbell" being valid for "CV". *Either* "Marnoo" or "Wesbell" would be deemed correct.

In some articles, certain aspects of the domain are not discussed at all. For example, the article entitled "The effect of tiller removal on ... spring barley" did not mention any chemical agents, so there could not be any acceptable candidate for the role filler "AGEN". These cases were ignored from the evaluation of the role indicator concerned.

At a later stage we would like to be able to distinguish between cases where there is or is not any true role filler by such methods as determining an acceptability threshold for the weight of the highest weighted candidate. Cases where the program did not propose any identifier for a given role because no templates associated with role were found to match were also not considered in this evaluation.

The results of our experiment are shown in Table 1. Thus, initial findings with a small training sample and a very small test collection suggest that the automatically-produced templates with no refinements were slightly more effective than the set of manually-produced templates for the domain of crop protection. We are now in the process of conducting a more extensive study with a training set of 200 articles taken from three different journals (650,000 words) and a test set of 100 articles, each indexed by two postgraduate students of agriculture.

5 Conclusion: Potential Advantages of Automatic Template Creation

We have described a method for the automatic creation of templates for the concept-based method of automatic abstracting. The production of abstracts is an integral part of the information retrieval process, enabling the user to judge quickly the relevance or otherwise of retrieved documents. In addition, the automatic abstracting process has a number of potential applications:

1. Automatic abstracting by means of automatic templates is applicable to machine translation, as may be seen by comparing the methods described here with RECIT, a multilingual analyser of medical texts, produced by Rassinoux et al [9]. RECIT accepts medical discharge summaries in any one of a number of European languages, and uses a method of analysis called proximity processing. This involves the invocation of various sets of rules, the most similar to our templates being the frequent-association rules. These involve the recognition of typical expressions such as temporal expressions or laboratory results. The internal representation of the input text produced by proximity processing is the conceptual graph model as defined by Sowa [10]. This internal representation serves as an "interlingua" or language-independent representation. Modules were produced for converting the conceptual graphs into any one of a number of natural languages. This approach to translation means that the complexity of the entire natural language does not need to be taken into account by the system.

Our system also takes advantage of a closed domain of knowledge to enable text to be reduced to a standard internal representation. However, unlike the RECIT proximity processing rules, our system requires no separate rule sets to analyse different languages: templates can be automatically created by our method whatever the source language. As was the case for RECIT, the production of rules for converting the internal representation into an abstract in the desired target language will be a much easier task than the full automatic translation of a source text directly into a target text, making a form of automatic translation possible.

2. Automatically generated templates can be used for subject identification. Templates generated from text in one domain will match other texts from the same domain much more frequently than they will match texts taken from outside that domain. This suggests that we can generate sets of templates for various domains, and then for an unknown text, find the number of matches produced by each set of templates. The set of templates which produces most matches will indicate the most probable domain of the unknown text. A similar approach might be used for genre identification, or for distinguishing between separate functional sections within a single document. Riloff and Lehnert [11] were able to use learned information extraction patterns found to be highly correlated with a training set of news articles about terrorism to determine whether or not other news articles were related to this topic.

3. A major advantage of generating templates automatically is that the technique can be transported to other domains. Although some analysis is required to identify the relevant roles for the new domain, the process overall is much speedier than the manual generation of templates for each new domain.

4. If cases are found where automatically-generated templates performed better than manually-produced templates on the same text testbed, the automatically-generated templates responsible for the retrieval of this additional information can be used in future to augment the original set of manually-produced templates. This way, the strengths of the two approaches can be combined.

5. In order to render the technique less domain-specific and applicable to any document describing a quantitative study, we have identified sequences of text which explicitly reveal the relationships between variables. For example, the sequence " A had a significant effect on B" can produce the template

```
? had a significant effect on ? /1:
        IV,-,DV/1.
```

meaning that the term represented by the ? on the left is the independent variable, and the term represented by the ? on the right is the dependent variable.

6. The concept-based abstracting process is related to the task of indexing, a central activity in information retrieval, since both involve the picking out of important concepts from a document. The effectiveness of the concept-based abstracting approach to indexing could be evaluated using traditional information retrieval experiments.

Acknowledgement

The Concept-Based Method for Automatic Abstracting project is supported by the British Library, award number RIC/G/360.

References

1. Pollock JJ, Zamora A. Automatic abstracting research at the chemical abstracts service. Journal of Chemical Information and Computer Sciences 1975; 15:226-232

2. Paice CD. Constructing literature abstracts by computer: techniques and prospects. Information Processing and Management 1990; 26:171-186

3. Rau LF. Knowledge organization and access in a conceptual information system. Information Processing and Management 1987; 23:269-283

4. Paice CD, Jones PA. The identification of important concepts in highly structured technical papers. In R.Korfhage et al. (eds), Proceedings of the 16th ACM SIGIR Conference, Pittsburgh, PA., June 1993; pp 69-77

5. Jones PA: Automatic Abstracting and Indexing of Technical Documents: an approach based on concept selection, PhD thesis, Lancaster University, Lancaster, 1995

6. Vickery A, Brooks HM. PLEXUS - the expert system for referral. Information Processing and Management 1987; 23:99-117

7. Jones PA, .Paice, CD. A 'select and generate' approach to automatic abstracting. In: McEnery T, Paice CD (eds), 14th British Computer Society Information Retrieval Colloquium, Springer Verlag, London, 1992, pp 141-154 (Workshops in Computing, ed. van Rijsbergen CJ)

8. Scott M. WordSmith tools manual. Oxford University Press, Oxford, 1996

9. Rassinoux A-M, Baud RH, Scherrer J-R. A multilingual analyser of medical texts. In: Second International Conference on Conceptual Structures (ICCS 94), University of Maryland, 1994

10. Sowa JF. Conceptual structures: Information processing in mind and machine. Addison-Wesley Publishing Co., New York, 1984

11. Riloff E, Lehnert W. Information extraction as a basis for high-precision text classification, *ACM Transactions on Information Systems* 1994; 12: 296-335