Principle-Based Structured Concept Generation

A Contribution to Knowledge-Based Document Indexing

Bas van Bakel and Reinier T. Boon

University of Twente, Knowledge-Based Systems Group

Abstract

This paper discusses the NLP module of Condorcet, an information retrieval project at the University of Twente, the Netherlands. The Condorcet system indexes scientific documents by mapping title and abstract of the documents to concepts and relations, defined in modern versions of classical indexing thesauri, i.e. *ontologies*. After a brief sketch of Condorcet's approach to document indexing and the design and development criteria used in the research, the linguistic principles are discussed that form the conceptual basis of the NLP presented in this paper. As engineering issues are of equal interest in this application-oriented research, formalization and implementation of the principles are discussed as well.

1 Introduction

Condorcet[1] is a domain-specific information retrieval (IR) project carried out at the University of Twente, the Netherlands. Its main objective is to build a prototype indexing system for large volumes (tens of thousands) of documents. The prototype will be developed and tested on 300 documents covering two scientific domains: *mechanical properties of engineering ceramics* as a subfield of engineering, and *epilepsy* as a subfield of medicine. The documents in the development corpus are taken from machine-readable one year volumes of two scientific journals: the 1988 volume of *Excerpta Medica* from Elsevier Science Publishers, and the 1990 volume of *Engineered Materials Abstracts* from Materials Information.

1.1 Condorcet's approach to indexing

Two approaches can be distinguished in document indexing, i.e. the *controlled-term* approach, and the *uncontrolled-term* approach[2]. In the controlled-term approach (often referred to as the classical approach), indexes are taken from a predefined and fixed thesaurus or classification system, whereas uncontrolled-terms are usually derived directly from documents by minimal processing like stopword removal and stemming. The uncontrolled-term approach is perfectly suited for corpora with a high redundancy rate (e.g. the internet), and when working with fairly superficial search requests. These circumstances hardly justify high investments required when adopting a controlled-term approach. However, in a good deal of retrieval situations – e.g. when retrieving patents, or in case of a litigation support

[1]Condorcet is funded by the Dutch Technology Foundation (STW), through the *Werkgemeenschap Informatiewetenschap*.

[2]These approaches to indexing may be applied both in retrieval on *abstracts* as conducted by Condorcet, and *full-text retrieval*.

system as evaluated by Blair and Maron (cf. Blair & Maron 1985) – high investments may very well pay off if they lead to significantly better performance than current, uncontrolled-term-based systems.

As Condorcet deals with corpora with a low redundancy rate, it has adopted the controlled-term approach to document indexing. Thus it is faced with the problem of high indexing costs. Condorcet proposes to combat these costs by partly automating the indexing process, which is one of its main research themes.

Another major research theme of Condorcet is to employ modern versions of thesauri and classification systems, which are called *ontologies*. In 'classical' terms, a structured ontology is a structured thesaurus involving a fair amount of precoordination. Structured ontologies consist of – possibly structured – concepts, and a number of clearly defined relations over these concepts. Concepts can be coordinated by using the relations. Condorcet claims that using structured concepts as index terms will lead to a higher precision in document retrieval than is currently achieved by state-of-the-art IR systems (cf. Sparck Jones 1995), as they are language-independent and non-ambiguous. For example, simple concepts like aspirin and headache point to all documents in which these two concepts are mentioned. However, by using structured concepts we can distinguish documents discussing aspirin as a cause of headache (causes(aspirin,headache)) from documents on aspirin as a cure for headache (cures(aspirin,headache)). Searching for the former group will exclude the latter, which is not possible when documents are indexed with simple, unstructured concepts.

1.2 Design and development

Condorcet's approach to indexing implies that the costs of building the system will be substantially higher than those of current systems, in particular the ones that employ the uncontrolled-term approach. In Condorcet, the development costs are reduced as much as possible by reusing existing systems. The *Unified Medical Language System* of the U.S. National Library of Medicine (National Library of Medicine 1996) is used as the domain knowledge source for the Excerpta Medica corpus, and in constructing the ontology for the ceramics domain, Condorcet draws on experiences gained in earlier projects of the Knowledge-Based Systems Group at Twente, in particular Plinius (Speel 1995). The NLP module of Condorcet is based on the NLP tool developed in the ELSA research project (van Bakel 1996). And thirdly, Condorcet's general (i.e. domain-independent) lexicon is derived from the CELEX lexicon (Burnage 1990).

Another way of keeping development costs low is by building a modularly structured system. Modularity also contributes to low costs in maintaining and extending the system. The knowledge-based approach entails that processes and knowledge resources are separated. As in Condorcet modularity and the knowledge-based approach are combined, reusability is facilitated to a great extent. Figure 3 in section 3 depicts the modularly structured index process developed in Condorcet.

As regards Condorcet's extendibility, it is required that solutions to practical

problems should also work for large-scale applications (i.e. for tens of thousands of documents). This is an important issue in building the actual implementation of the Condorcet system. We address this issue in section 4 of this paper.

2 NLP and knowledge-based indexing

As was stated in the previous section, the main objective is to build a prototype domain-specific, knowledge-based indexing system for scientific documents. Document indexing by Condorcet basically consists of mapping title plus abstract (henceforth referred to as the *description*) of a document to the concepts and relations defined in the ontology. Figures 1 and 2 present examples of document descriptions, taken from the epilepsy and materials science domains, respectively.

AN: 88100203
TI: Effects of zonisamide in children with epilepsy
AB: The effects of zonisamide (1,2-benzisoxazole-3-methanesulfonamide: AD-810) were studied in 50 children with epilepsy, ranging in age from 3 months to 20 years (mean, 10.5 years). The types of epilepsy were primary generalized in one case, secondary generalized in 32, and partial in 17. The initial dose was 1-6 mg/kg/day and the dose was increased to 1.5-15 mg/kg/day. Four cases (8%) showed a complete disappearance of seizures and thirteen patients (26%) had a disappearance rate of 50% or more of seizures. Disappearance or improvement of seizures was obtained in 31% of the cases of generalized epilepsy and in 41% of the cases of partial epilepsy. Zonisamide was effective in 39% of cases of Lennox-Gastaut syndrome. Seizures completely disappeared in three of the four new cases. Spike discharges disappeared or significantly decreased in 22% of the cases that had undergone electroencephalograms. The blood levels of zonisamide were 10.8-18.8 mu g/ml in the three new cases when the seizures were controlled. Side effects such as drowsiness, ataxia, and salivation were observed in 42% of the children, more particularly in children receiving polypharmacy.

Figure 1: Epilepsy document description

The major problems Condorcet encounters when mapping descriptions to concepts and relations involve problems related to natural language. This means that 'solutions' to these problems must be based on knowledge of language, i.e. on linguistic principles. This can hardly be considered a new approach to document indexing and, with it, to information retrieval. There are many examples of IR systems in which NLP plays a prominent role – e.g. ADRENAL (Lewis, Croft & Bhandaru 1989), FERRET (Mauldin 1991), CLARIT (Evans et al. 1991), MEDLEE (Friedman et al. 1995)), and AIMS (Hodges et al. 1996). However, according to Harman, Schäubele & Smeaton 1996, NLP still has to make its first significant contribution to improving document retrieval systems. As Smeaton 1997 states, as for substantial NLP contributions to operational IR systems (i.e. other than low level

processes as spelling error correction or stemming and word normalization), there is not too much work to report on. In Smeaton's view an important reason for this is that IR and NLP are inherently different processes. IR is inexact whereas NLP is not, and "these fundamental differences in approach seem to point to an uncomfortable alliance." Only a change of approaches in both IR and NLP will lead to progress, as the current approaches only cause "the 'butting of heads', which we see at present with IR attempting to cherry-pick any appropriate techniques from NLP" (cf. Smeaton 1997, p. 136).

019001C1-C-0019
02 Influence of Ambient Temperature Sliding Velocity Under Unlubricated Sliding Conditions on Friction and Wear of Si sub 3 N sub 4 Up to 1000 deg C.
03 The tribological behaviour of Si sub 3 N sub 4 /Si sub 3 N sub 4 sliding pairs in pin-on-disk configuration for sliding velocities between 0.03-3 m/s, constant load of 10 N and environment-temperatures between 22-1000 deg C is dependent on the overlap ratio, the temperature and the sliding velocity. An influence of the phase composition was not observed for the three tested commercial Si sub 3 N sub 4 materials. The results are: (1) Coefficient of friction lies for solid state friction under steady state conditions between 0.5-1. (2) Wear rate increases with rising ambient temperature–especially at sliding speeds < 1 m/s. (3) The tribological behaviour for temperatures => 400 deg C is characterized by a high wear/low wear transition with increasing velocities. (4) The influence of overlap ratio on wear increases with increasing ambient temperature. A small overlap ratio is tribological disadvantageous for Si sub 3 N sub 4 sliding pairs. Si sub 3 N sub 4 /Si sub 3 N sub 4 sliding pairs do not meet for the described sliding claims without lubrication.

Figure 2: Materials science document description

From Smeaton's diagnosis we learn that Condorcet's approach to document indexing – both in its linguistic and non-linguistic aspects – should do away with the fundamental differences of IR and NLP. Only then will it be possible to successfully combine the two. Condorcet does just that. The proposed document indexing in terms of structured concepts, and the possibilities of deductive matching that come with it, make the retrieval part of Condorcet more exact than those of current IR systems (cf. Vet & Mars 1996). Likewise, the NLP module is tuned to the specific needs that apply to knowledge-based document indexing, causing it to differ from general-purpose NLP systems. NLP within Condorcet is highly application-oriented; the knowledge-based approach guarantees that the NLP system is based on linguistic principles, and that therefore no ad hoc solutions will be applied.

3 Structured Concept Generation

We said before that the major problems in mapping descriptions to concepts and relations are linguistic in nature. Therefore we need knowledge on how concepts and relations can be expressed in natural language. It appears that there are many possible ways, by using different syntactic constructions. Consider the following sentences (these examples are partly based on the text in Figure 1):

(1) Effects of zonisamide in children with epilepsy.

(2) Zonisamide affects epilepsy.

(3) Epilepsy was affected by using zonisamide.

(4) Zonisamide was effective in 39% of cases of epilepsy.

Given the coarse granularity of the ontology used, these sentences all express the same relation over the same two concepts, i.e. affects(zonisamide,epilepsy), only in a different syntactic form. This entails that in order to produce proper structured conceptual representations, we not only need to determine the syntactic surface structures of these sentences, but also their underlying, deep structure, as deep structures rather than surface structures contain the necessary information for mapping sentences to concepts and relations.

To obtain deep structures we will use syntactic principles from Government & Binding (GB) theory (Chomsky 1981). This theory is chosen for theoretical and practical reasons. First and foremost, Chomsky's *Principles & Parameters* framework can explain a wide variety of language phenomena using just a few assumptions (cf. Fong 1991). Using GB therefore makes it possible to develop a relatively small and elegant, principle-based NLP system. As it is of secondary interest how these principles should be formalized in GB (cf. Chomsky 1990), we can freely formalize and implement them, and separate the linguistic knowledge resources from the processes as required.

Another advantage of GB is that it has an autonomous syntax: syntactic rules operate independently of any other subcomponent of Universal Grammar, e.g. the Logical Form component. Therefore we can use the syntactic component of GB theory without having to adopt other GB components as well. A more practical reason for using GB theory is that it has been used successfully in ELSA, an NLP tool for a chemical information extraction system (van Bakel 1996 ,Postma, van Bakel & Kateman 1995). Although ELSA's task differs from Condorcet's, most of its conceptual basis can be reused. The major difference between ELSA and Condorcet is that ELSA's main objective was the distribution of thematic functions, whereas the central task of Condorcet's NLP is to produce structured conceptual representations for document representation sentences. It can be said that in Condorcet, structured concepts more or less take the place of thematic functions in ELSA. The required preparatory phases of both are highly similar, which makes reuse of ELSA's set of principles and parameters in Condorcet an obvious choice.

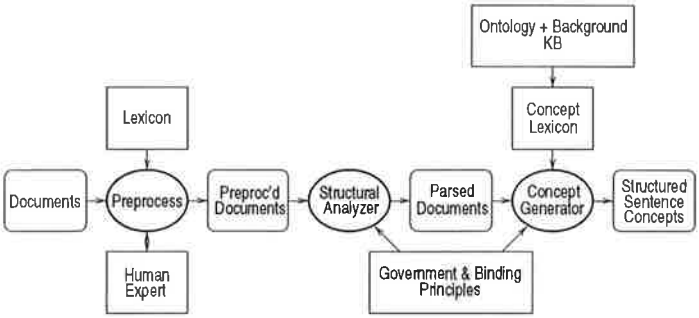

Figure 3: Condorcet's Index Process. The structural analyzer and the concept generator are discussed in this paper.

3.1 Structural analyzer

We saw before that in order to produce structured conceptual representations, we need to know the deep structures of sentences. Deep structures can be generated from syntactic surface structures. In other words, we will first have to generate structuralistically motivated constituent structures for all sentences in the document representation. This task is performed by the *Structural Analyzer* (cf. Figure 3). Structural analysis is primarily based on $\bar{X}$*-theory* (cf. Chomsky 1981). In this process, N, V, A, and P are regarded as *lexical heads*, and I (Inflection) and C (Complementizer) are the *non-lexical heads*. At structural level, the major categories are analysed in accord with the $\bar{X}$ Conventions. The *maximal projections* for the lexical heads are represented as NP, VP, AP, PP, IP and CP, respectively.

Condorcet's Structural Analyzer (discussed in detail in Oltmans 1998) produces tree structures as depicted in Figure 4.

3.2 Generating enriched surface structures

The next step in structured concept generation consists of generating deep structures. Actually, *Enriched Surface Structures* (ESS) rather than deep structures are generated. ESS are constituent structures in which constituents are linked to their original deep structure positions, without changing the word order of the sentence. ESS generation is performed by the *Concept Generator* (cf. Figure 3).

ESS generation is based on a set of subgrammars and rules that can be divided into *Move* α rules and *Control Structure* rules, reflecting the several principles and parameters of GB theory involved.

For the moment, rules for *Move* α are restricted to *Move NP* rules, as the descriptions in the present development corpus[3] do not contain WH-questions. A

[3]Condorcet follows a concentric work programme: the system will initially be developed for a small

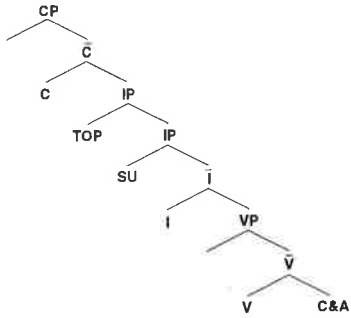

Figure 4: Canonical surface structure. In this tree, the syntactic subject resides under SU. Topicalized elements (CPs, PPs, NPs, adverbs) are collected under TOP. All verb complements are under C&A (complements and adverbials), using underspecification in transparency situations (cf. van Bakel 1996,Oltmans 1998).

crucial condition for all Move α rules is that they obey the *Subjacency Condition*, thus adhering to the principle of strict cyclicity (cf. Chomsky 1981). Linking constituents to deep structure positions is making use of *Case Theory* and *Theta theory*. In GB theory, surface structures are generated from deep structures by moving NPs from caseless positions to positions where they receive Case (i.e. [SPEC,IP] and under C&A), in order to escape the Case Filter for NP, stating that every lexical NP needs Case (cf. Chomsky 1981). What is more, the NPs are moved from theta role positions to non-theta role positions.

When generating the ESS of a sentence, constituents in `<+case,-theta>` positions are therefore linked to their `<-case,+theta>` deep structure positions, by creating argument chains. The result of this process is illustrated in Figure 5. Theta positions are [SPEC,VP] for the external theta role, and positions under C&A for internal theta roles. Although distribution of thematic roles is not an issue in Condorcet, theta role positions are of importance, as we will see in the discussion of concept coordination.

Move α rules help in determining the deep structure subject of a sentence. In case of nonfinite clauses, however, deep structure subjects cannot be determined by Move α rules, as an overt syntactic subject is lacking in these structures. The following are examples of sentences with nonfinite clauses, taken from the Condorcet development corpus:

(5) The purpose was *to inquire into the determinants of psychopathology*

(6) Several risk factors were found *to be related to each behavioral measure*

corpus of documents. The full process as well as the required resources are developed and tested on a corpus that grows larger in each phase of the research.

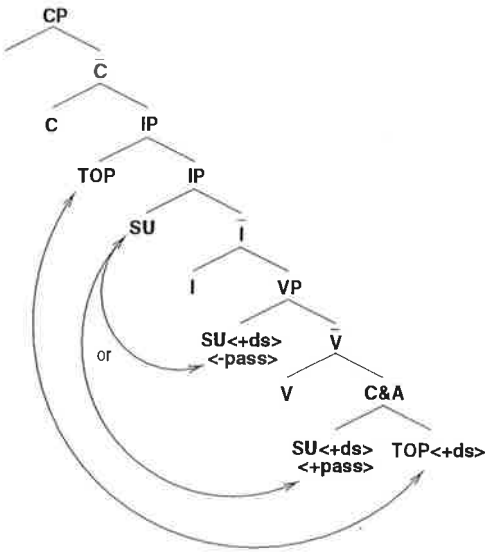

Figure 5: Enriched surface structure. TOP and SU are linked to their deep structure positions. SU is linked to [SPEC,VP] in case of active, and to the leftmost position under C&A in case of passive.

(7) Animals *receiving IV injections of 5 and 10 mg/kg cocaine* experienced convulsions

(8) Cocaine preference *developed for the side* associated with the drug

For these cases we use *Control Theory*, as this theory deals with the subjects of infinitival clauses. According to Control Theory, PRO must appear where an NP is required but no Case is assigned. This follows from the *Projection Principle*, which states that representations at each level (Logical Form, deep structure, surface structure) are projected from the lexicon, in that they observe the subcategorization properties of lexical items. Next, the reference of PRO must be determined, i.e. the constituent by which PRO is 'controlled.' This is either the complement or the subject of the matrix verb (cf. Chomsky 1981).

In Condorcet's structural analysis, all clauses (relative, infinitival) are analysed as CPs. Lexical information of the matrix verb determines whether it is a control verb, and, if so, whether it is the subject or the object that controls the [SPEC,IP] of the subordinate clause. PRO is inserted in [SPEC,IP] in CP if the matrix verb of the sentence is a control verb, and if [SPEC,IP] does not receive Case (i.e. if the CP is an infinitival clause). Depending on the lexical information of the matrix verb, the [SPEC,IP] is then linked to the subject or the object of the matrix verb, and the controlling NP is linked to the [SPEC,IP] position. Moreover, PRO is also inserted in case of NPs with a postmodifying infinitival clause (cf. example sentence 8).

3.3 Concept retrieval and coordination

After ESSes have been generated, concepts can be retrieved and coordinated. Concept retrieval is a simple process: every lexical head is looked up in the domain-specific concept lexicon (cf. Figure 3), and its corresponding concept is placed under a CC (Concept Cluster) node, which is created rightmost under $\bar{X}$. Lexical heads can correspond with simple concepts (e.g. zonisamide – zonisamide), or coordinating concepts (e.g. effects – affects(,)) expressing relations. The latter have argument slots with semantic type conditions.

Coordination of concepts is carried out after concept retrieval, as follows. For every XP of which the lexical head corresponds with a coordinating concept, the theta role positions (i.e. [SPEC,XP] and [COMPL,$\bar{X}$]) are checked for (links to) lexical heads with corresponding concepts (simple or coordinating). If these are found, then it is checked whether these concepts can fill the argument slots of the coordinating concept, by checking their semantic types. If the concepts have the proper semantic type, they will fill the argument slots. In Figure 6 concept retrieval and coordination are illustrated.

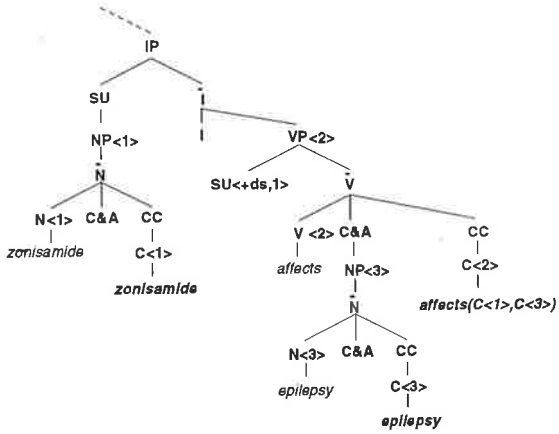

Figure 6: Concept coordination. The concepts zonisamide and epilepsy are coordinated by affects, as the words zonisamide and epilepsy both are in theta role position of affects, and the concepts have the proper semantic type. Constituents are linked to deep structure positions by indexes (e.g. <1>). Concept coordination is done similarly.

The process of concept retrieval and coordination handles semantic (conceptual) and syntactic ambiguities rather elegantly. We are dealing with semantic ambiguity if there are more possibilities regarding argument slot filling by concepts. If this is the case, all possible structured concepts at a specific XP level will be packed in one conceptual structure containing all possible readings. Underspecification is

used in dealing with syntactic ambiguities, e.g. in sentences with transparent constituent boundaries like PP-sequences, etc. The Structural Analyzer produces underspecified surface structures for sentences with PP-sequences under C&A, in accord with the Principle of Minimal Attachment. Reanalysis of these sequences will only take place if the ESS including the retrieved concepts meets the following conditions:

- The lexical head of the NP (or the main verb of the CP) preceding the PP to be lowered, corresponds with a coordinating concept.
- The lexical head of the NP (or the main verb of the CP) within the PP to be lowered, corresponds with a concept (simple or coordinating).
- The corresponding concept of the head of the NP (or the main verb of the CP) that is within the PP to be lowered, could not be coordinated at matrix level.

In all other cases, transparency situations are ignored. This highly domain-driven strategy proves to be very efficient, as structural ambiguities will only be dealt with if there is reason to do so. This way, the Structural Analyzer only has to produce one structural representation for each sentence, thus doing away with the exponential, and therefore laborious and time-consuming task of producing all possible structural analyses.

4 Tree Manipulator

In the previous section we saw that structured concept generation consists of two subprocesses: Structural Analyzer and Concept Generator. The structural analyzer consists of a parser[4], and a transformational reanalyzer (cf. Oltmans 1998).

Tree reanalysis and structured concept generation are performed by transforming parse trees. The design criteria of Condorcet indicate that specification of these transformations should be made explicit. Therefore we decided to specify the translations independently, i.e. separate them from the program that will perform these translations, rather than building a program in which the translations are implicitly defined.

ELSA was built using the tool GRAMTSY (cf. Coppen 1991), a transformational system that allows for separate specification of transformations, quite similar to the way transformations are written down in theoretical linguistics. GRAMTSY is well-suited for developing relatively small-scale proof-of-concept prototypes. However, its performance is very low when working with large volumes, as it is string oriented. This is why we decided to build a new tool to support structured concept generation, called TREMA, in which GRAMTSY's functionality was to be adopted and improved.

[4]The parser is based on an AGFL grammar. AGFLs are simple CF grammars extended with features. Parsers generated from AGFLs are fast, flexible and can easily be incorporated in larger software programs.

TREMA transforms a given input tree, making use of a transformational grammar. A TREMA specification consists of a number of grammars and rules. A grammar is the main structure, and it is used to group rule calls, although a grammar can also call sub-grammars. Rules are used to describe tree translations. Each rule consists of a structural description (SD), a set of conditions (COND) and a structural change (SC)[5]. Unlike GRAMTSY, TREMA describes trees as data structures, which makes matching and translation operations much more efficient in comparison with the string-oriented approach. What is more, we supplied TREMA with the facility to create links to knowledge-based systems. This makes it possible to specify tree translations making use of semantic inferences, rather than inspecting lexica as is the present situation (described in the previous section).

4.1 Tree manipulations

Various operations may be performed on a tree. These operations consist of the basic operations known from transformational linguistics. TREMA is employed to check the conditions specified in the transformational rules, and perform the operations when necessary, which will result in zero or more output trees. The basic transformational operations are:

- adding, removing, moving and/or copying (sub)trees, depending on certain conditions
- changing the label of a (sub)tree
- adding, removing or changing features of a (sub)tree
- Chomsky adjunction of a (sub)tree[6]

Similar to GRAMTSY, transformational operations have to be defined in terms of rules that may be clustered in sets of sub-grammars.

4.2 Cyclicity

A crucial characteristic of TREMA is that grammars can be applied *cyclically*. Cyclic application of grammars has been explicitly or implicitly assumed from the very beginning of transformational grammar, which is reflected in the principle of *cyclicity*. This principle states that rules apply to sentential subdomains in tree structures, in such a way that no rule can affect elements outside the domain being processed (cf. Coppen 1991, p. 327). This is in contrast with non-cyclic grammars, that may apply to the entire tree. As cyclicity is a crucial aspect of transformational grammars, it is provided in a principled fashion in TREMA.

[5]It is beyond the scope of this article to give a detailed account of how GB principle & Parameters are formalized in a Trema specification. This is why we only stipulate the general outline of Trema specifications here.

[6]Chomsky adjunction can be used to transform non-binary branching trees into binary branching trees (cf. Chomsky 1996). If a node A is Chomsky-adjoined to a node S, the latter is copied, and A and the original S will become the children the copy. This can be done in two ways: Chomsky left-adjunction or Chomsky right-adjunction.

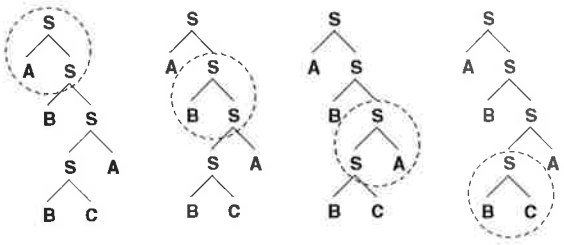

Figure 7: Trees with cycles of depth 0

An important aspect of a cyclic grammar is a parameter that represents the *depth value* of the grammar. The depth value states the *scope* of a cyclic grammar. If the depth value is zero, this means that, at any cyclic level, the next cyclic node is visible, but its content is not. If the depth value is one, the content of the next cyclic node is visible. Given the cyclic domain of S, the circles in the trees in Figure 7 show the scope of a cyclic grammar with a depth value of zero. Figure 8 shows the scope of a cyclic grammar with a depth value of one.

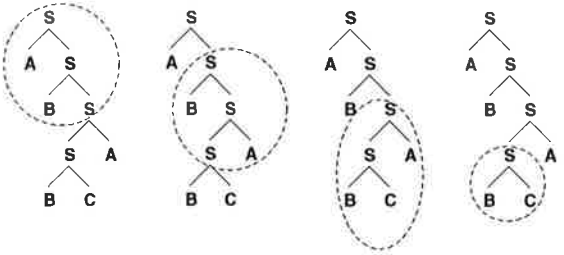

Figure 8: Trees with cycles of depth 1

If a grammar is applied cyclically, it means that for each cyclic node, TREMA applies the grammar with the cyclic node as top node. Nodes above and next to that node are invisible inside the cyclic grammar, as are cyclic nodes that are beyond the scope of the cyclic grammar (cf. Figure 8). In this way, TREMA provides an elegant way of constructing transformational systems that adhere to the principle of strict cyclicity as discussed in the previous section.

4.3 Implementation and provisional results

TREMA is built in C++. It consist of various logical machines, which are implemented separately, but these machines interact to accomplish the complex task of

a transformational system. The main machine controls the grammar and rule execution. Other machines take care of matching, condition checking and applying changes to trees.

Provisional tests on a development corpus of twenty-four documents (on a SUN/SPARC 5) show that the resulting tool is up to 50 times faster than GRAMTSY. The results of these tests are presented in Table 1. We should stress that Condorcet's entire indexing process has not yet been tested on unseen material; such tests are foreseen to be performed in the final stage of the research project[7]. Therefore, we realize that it would be premature to draw concrete conclusions from these tests. Nevertheless, we think the results indicate that the performance of the completed NLP system will most likely meet the standards for reality-level applications, in that it will be able to handle large volumes of data.

5 Conclusions

We have discussed an NLP system for generation of structured concepts, as a part of a domain-specific document indexing system that uses controlled terms. In developing the NLP system we took a principle-based approach, which has led to a system seemingly fit to perform its task at a reality-level scale, i.e. handling tens of thousands of documents. In combination with the AGFL parser generator, the Tree Manipulator has enabled us to implement structured concept generation in a principle-based fashion, in accordance with the design criteria of Condorcet.

Although we have enough reason to be optimistic about the outcome of the Condorcet research, there is a potential bottleneck. In its approach to controlled-term indexing, Condorcet relies to a great extent on large-scale knowledge resources. It is assumed that such resources (i.e. lexica, concept lexica and ontologies), which should be consistent and ready-to-use, are available. However, in practice it appears that usable resources are scarce. What is more, even resources specifically designed for supporting NLP tasks seem to be inconsistent (cf. van Bakel et al. 1997), which is why they can only be used after having been thouroughly tested and evaluated. From these findings it follows that there is a need for consistent knowledge resources that can readily be used. Provided that knowledge sources will be improved in this respect, we can say that Condorcet's approach to (semi-)automatic document indexing is viable, in particular for corpora with a low redundancy rate.

[7] The present article discusses Condorcet's findings after two years of a planned total of four.

Table 1: Test results of structured concept generation for 24 documents.

Statistics of parsing 24 documents	
Total number of abstracts	24
Total number of sentences	197
Total number of words	4643
Average number of sentences per text	8.21
Average number of words per sentence	23.57
Structural Analysis I: Parsing	
Sentences not parsed	0 (0.00%)
Sentences parsed	169 (85.79%)
Sentences robustly parsed	28 (14.21%)
Average number of parsings per sentence	1.00
Total parse time	8.23 seconds
Average parse time (sec/parse)	0.04
Number of sentences per second	23.93
Number of words per second	563
Structural Analysis II: Canonization and Robustness	
Regular Sentences	169 (85.79%)
Sentences succesfully transformed	27 (13.71%)
Sentences not canonical	1 (0.51%)
Total parse time (sec)	4.83
Average parse time (sec/parse)	0.02
Number of sentences per second	40.79
Number of words per second	961
Structured Concept Generation	
Total parse time (sec)	35.59
Average parse time (sec/parse)	0.18
Number of sentences per second	5.54
Number of words per second	130

References

Bas van Bakel (1996), *A Linguistic Approach to Automatic Information Extraction*, PhD Thesis, University of Nijmegen, The Netherlands.

Bas van Bakel, Reinier T. Boon, Nicolaas J.I. Mars, Jeroen Nijhuis, Erik Oltmans & Paul E. van der Vet (1997), Condorcet Annual Report, University of Twente, CTIT Technical Report Series, No. 97-30 , Enschede, The Netherlands.

David C. Blair & M.E. Maron (1985), An evaluation of retrieval effectiveness for a full-text document-retrieval system, *Communications of the ACM* 28, 289–299.

Gavin Burnage (1990), *CELEX - A guide for users*, Centre for Lexical Information, Nijmegen, The Netherlands.

Noam Chomsky (1981), *Lectures on Government and Binding*, Foris Publications, Dordrecht, The Netherlands.

Noam Chomsky (1990), On Formalization and Formal Linguistics, *Natural Language and Linguistic Theory* 8, 143–147.

Peter-Arno Coppen (1991), *Specifying the Noun Phrase*, Dissertation, University of Nijmegen, Thesis Publishers, Amsterdam, The Netherlands.

David A. Evans, Kimberly Ginther-Webster, Mary Hart, Robert G. Lefferts & Ira A. Monarch (1991), Automatic indexing using selective NLP and first-order thesauri, *Conference proceedings RIAO91. Intelligent text and image handling, 2–5 April 1991, Barcelona, Spain*, Centre de Hautes Études Internationales d'Informatique Documentaire, Paris, 624–644.

Sandiway Fong (1991), *Computational Properties of Principle-Based Grammatical Theories*, PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA.

C. Friedman, G. Hripsak, W. DuMouchel, S.B. Johnson & P.D. Clayton (1995), Natural language processing in an operational clinical information system, *Natural Language Engineering* 1, 83–108.

Donna Harman, Peter Schäubele & Alan Smeaton (1996), Document Processing, *Survey of the State of the Art in Human Language Technology (http://www.cse.ogi.edu/CSLU/HLTsurvey/)*, Giovanni Battista Varile & Antonio Zampoll, eds., Centre for Spoken Language Understanding, Oregon Graduate Institute of Science and Technology.

Julia Hodges, Shiyun Yie, Ray Reighart & Lois Boggess (1996), An automated system that assists in the generation of document indexes, *Natural Language Engineering* 2, 137–160.

David D. Lewis, W. Bruce Croft & Nehru Bhandaru (1989), Language-oriented information retrieval, *International Journal of Intelligent Systems* 4, 285–318.

Michael J. Mauldin (1991), *Conceptual information retrieval. A case study in adaptive partial parsing*, Kluwer Academic, Boston, MA.

National Library of Medicine (1996), UMLS Knowledge Resources. Seventh experimental edition, Bethesda, MD.

Erik Oltmans (1998), A Two-Stage Model for Robust Parsing, *Proceedings of the International Conference on Natural Language Processing and Industrial Applications (NLP+IA 98)*, Chadia Mohgrabi, ed., Moncton, New Brunswick, Canada.

Geert J. Postma, B. van Bakel & G. Kateman (1995), Automatic extraction of analytical chemical information. System description, inventory of tasks and problems, and preliminary results, *Journal of Chemical Information and Computer Science* 36, 770–785.

Alan F. Smeaton (1997), Information Retrieval: Still Butting Heads with Natural Language Processing?, Springer-Verlag, Information Extraction - A multidisciplinary approach to an emerging information technology.

Karen Sparck Jones (1995), Reflections on TREC, *Information Processing and Management* 31, 291–314, also published as Cambridge University Computer Laboratory Technical Report TR347.

Piet-Hein Speel (1995), *Selecting Knowledge Representation Systems*, Ph.D. Thesis, University of Twente.

Paul E. van der Vet & Nicolaas J.I. Mars (1996), Coordination recovered, *Informatiewetenschap 1996. Wetenschappelijke bijdragen aan de vierde Interdisciplinaire Conferentie Informatiewetenschap*, K. van der Meer, ed., Werkgemeenschap Informatiewetenschap, Delft, 139–151.