Automatic Extraction of Analytical Chemical Information. System Description, Inventory of Tasks and Problems, and Preliminary Results

G. J. Postma,*,† B. van Bakel,‡ and G. Kateman†

Departments of Analytical Chemistry and Language and Speech, Katholieke Universiteit Nijmegen, Postbox 9102, 6500 HC Nijmegen, The Netherlands

Received September 29, 1995⊗

A system for semiautomatic extraction of information from abstracts describing analytical methods is described. The system is based on the theory of Government and Binding for the syntactic part and Conceptual Graphs for the discourse analysis part. The system is modular and largely domain independent. The corpus of abstracts for which it is being developed contains abstracts from Analytical Abstracts Online. The current status of the system is that the grammar is finished and that the lexicon and the discourse module are under development. Preliminary results are that the system is capable of analyzing six abstracts on various analytical techniques.

INTRODUCTION

Textual information still has a increasing importance for the chemical scientist. New digital information systems like Gopher[1] and World Wide Web,[2] that use Internet, make more information accessible, and these new sources again stress the importance of having a grip on the contents of the texts. In order to make textual information accessible for computer programs to use, control, classify, or reason with its contents, text has to be converted to some structured meaning representation that links the concepts mentioned in the text in a meaningful manner. Natural language processing techniques can be used for this task.

Several authors have published work in the field of automatic information extraction within the chemistry domain. Nishida et al.[3] have developed a system for extraction and storage of information contained within patent claim sentences in the domain of semiconductor production. Ai et al.[4] developed a system for the extraction of (part of the) procedural synthesis information from the experimental section of one journal for organic chemistry. Chowdhurry and Lynch[5] worked on the extraction, representation, and storage of textual descriptions of compounds in a chemical reaction database. Mars and van der Vet[6,7] worked on a system for the information extraction from a set of abstracts on mechanical properties of ceramic materials. The authors have published work concerning an experimental system for information extraction from short analytical method descriptions concerning one analytical technique.[8] This research revealed that the approach that was used was not efficient enough and that more robust and better theoretically founded principles and techniques should be used for this task. Implicitly the work of the others revealed that information extraction produces useful results given that the domain is limited; most of the techniques used were, more or less, dedicated to a specific domain and to a specific text structure. The publications revealed, too, that the systems, thus far, could only cope with a relatively small set of short texts. This is published by Ginsberg[9] as well as among others. The requirement of a robust parser is implied by the work of Myaeng et al.[10] as well. The preliminary results of their information retrieval system for Wall Street Journal articles reveals that a large number of errors in the resulting semantic representation originated from the use of a superficial parsing technique.

The texts, from which the system that is being developed by the authors should extract information, are abstracts from Analytical Abstracts.[11] These abstracts contain relatively free text within the analytical chemistry domain. Beside this, part of the sentences are written in, more or less, telegram-style, and the sentences can contain a number of defined abbreviations. Although abstracts do not seem to be a reliable source of information,[12] they are selected because of the aforesaid text characteristics being an ideal test domain of the system for various applications. Given the aforementioned results, these text characteristics motivated a choice for more robust techniques to perform the task of information extraction. This in contrary to, for instance, the work of Chowdhurry and Lynch and Ai et al. They used more template-like structures to extract the required information, because of the limited structure of the target texts.

The information that is to be extracted are the characteristics of an analytical method (analyte, matrix, working range, technique applied, precision, accuracy and detection limit; see also the content requirements of analytical abstracts as described in ref 12) and the described actions that (roughly) comprise the analytical method together with the participants and circumstances of the actions.

Parts of the underlying theories of the current system (Government and Binding theory and Conceptual Graph theory) are used by a number of authors in a different manner and within other domains.[13−17] The METEXA system is being developed as an EC project for the information extraction and structured storage of radiological reports. The texts have similar characteristics as those for which the current system is being developed. Their semantic and pragmatic analysis is based on Conceptual Graphs.

Besides the aforementioned requirement for robustness, the starting points for the system described in this paper were

−modularity
−sound theoretical foundation

* Author to whom correspondence should be addressed.
† Department of Analytical Chemistry.
‡ Department of Language and Speech.
⊗ Abstract published in *Advance ACS Abstracts,* May 1, 1996.

S0095-2338(95)00206-X CCC: $12.00

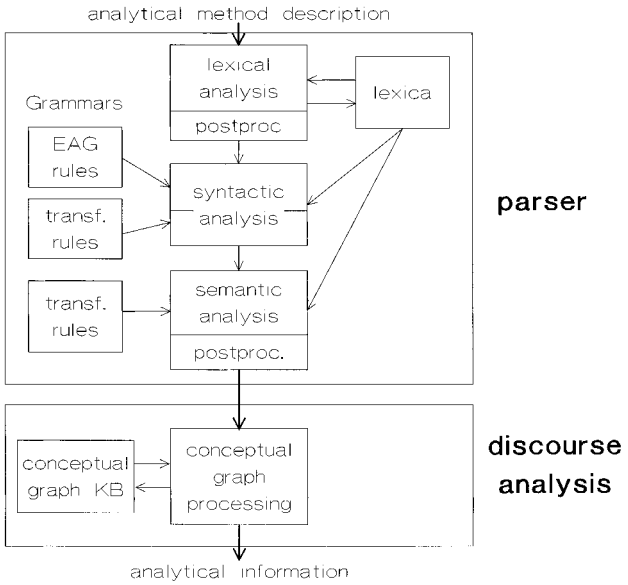

Figure 1. Overview of the different modules of the system.

- −domain independency
- −semiautomatic: the user is consulted if the system encounters problems.

The last starting point is based on the aforementioned publications concerning automatic information extracting which reveal that information extraction based on Natural Language techniques is still very difficult with varying percentages of accurate results. In such a situation a semiautomatic system is expected to perform better than a fully automatic system. The other starting points will be motivated in the next sections.

The system should initially be developed for a test-set of 124 abstracts.

THEORY AND IMPLEMENTATION

Text analysis normally consists of lexical, syntactic, semantic, and discourse analysis as is also depicted in Figure 1. The various tasks can be integrated in single modules that execute them concurrently or intertwined. An advantage of integration is that the semantics can be used as soon as possible in order to limit the number of possible solutions generated by the syntactic analysis. A disadvantage is that the maintainability decreases strongly as the system grows. This was experienced during the development of a previous system[8] as well. This, together with the aforementioned requirement of a robust syntax and semantics, motivated the choice for separated modules. Other advantages are the following: the modules can be developed independently (by different people), a module can be exchanged by one based on other principles, if this is required, and a modular structure gives a better insight of the specific types of knowledge that are necessary for the different modules (and submodules).

Another requirement of the system was that it should be as independent as possible of the domain. This was implemented by locating the domain dependent procedures and information in separate modules and files (mainly the lexica). In this way results of other investigations can be implemented more easily, in case of problems, its origin can be determined more easily (domain specific or general linguistic) and research can be directed to it, and parts of the system can be used for other domains as well. Also, it facilitates a better understanding of all phenomena that play a role.

The lexical module[18] consists of two lexica, a morphological analyzer for words and a lexical postprocessor mainly recognizing word groups. The first lexicon is filled with (domain dependent) single words. Its structure is domain independent: it contains entries for (the stem forms of) words and abbreviations, together with their syntactic categories (nouns, verbs, etc.), semantic categories, and if necessary a reference to an enumeration of the roles, the role identifying prepositions (if applicable), and the expected semantic classes of participants that are linked by the given roles to the words (semantic selection restriction frames). The semantic selection restriction frames are stored in a separate file and are indexed by numbers (in order to save space: a frame can apply to more than one verb). The second lexicon contains concepts that consist of more than one word.

The morphological analyzer contains the normal domain independent morphological rules (functions) that deal with declensions of words, the recognition of adverbs that are derived from adjectives, and the handling of plurals. Beside this, it contains functions for the recognition of numerals, and it contains separate domain dependent functions for the recognition of inorganic structural formulas (e.g., Na_2CO_3) so that these need not be given in the lexicon (besides those that need extra semantic subcategorization). The abstracts contain a number of complex strings that need special attention. These are in fact combinations of words which are not separated by spaces but need to be separated. Examples are .+−.0.049 and 0.02M-HCl (.+−. stands for ±; in general Analytical Abstracts encodes all non-ASCII characters as strings between dots). These are tackled by a separate procedure as well. It checks whether components occur in the lexicon, first taking into account the possible existence of abbreviations, punctuation marks, chemical formulas, and numbers. If a full stop occurs at the end and cannot be recognized as being part of the abbreviations it is recognized as sentence end.

The morphological analyzer is implemented as a SPITBOL[26] program. Each string between spaces is checked for occurrence in the first lexicon, and if there is no entry the various morphological rules and the above mentioned procedures are applied after which the first lexicon is consulted again.

The postprocessor is called after the morphological analyzer using its output as input. It deals with a number of (domain dependent) concepts that consist of more than one word. Compound words and idiomatic expressions (that syntactically can be viewed upon as one word, like “with respect to”) are recognized by consulting the second lexicon after which the component words and their categories are replaced by the compound term (or idiomatic expression). The postprocessor deals with complex chemical compound names by looking in the first lexicon for its parts. All possible parts are labeled, and if the labels agree with each other the parts are replaced by the compound name with one set of syntactic and semantic categories. This way complex chemical compound names need not to be stored in the lexicon. The background of this procedure is that the set of parts is limited, which is contrary to the size of the set of chemical compound names.

Lexical ambiguous words get all the word classes that are possible. A choice after the proper one is made during the parsing process. If a word cannot be processed and classified during the lexical phase the user is automatically asked for all the lexical information. A dedicated user-friendly lexicon editor is being developed.

The parser is based on Chomsky's principles of Government and Binding[19,20] for the syntactic part and Montague semantics for the semantic part. Chomsky's principles of Government and Binding are syntax oriented. It is based on general linguistic principles and this basis should lead to a more robust parser; its choice is motivated by research interest in the application possibilities of its theory as well. One of the features is that it works with general language-wide templates instead of far more language specific phrase-structure rules. Its appealing features are, for instance, described by McHale and Myaeng.[14] The theory does not postulate a strict formalism; it is implemented as a transformational grammar in GRAMTSY (a so called "transformational driver"; for more information, see ref 21).

The choice of Montague semantics is motivated by the solid logical foundation. The output of the parser is not an intensional logic representation, however, but a predicate logic representation. A discussion on possible critics and a motivation of the choices made are given in more detail by van Bakel.[22]

The parser uses "underspecification" as a principle in order to eliminate combinatorial explosion as a result of ambiguities that cannot be resolved in the different modules.[23] A choice between multiple possible solutions is postponed, using some general notation, to the module that is capable of resolving it. This prevents the generation and testing of numerous solutions in order to locate the correct one. An example for which it is used is "The determination of clemastine fumerate in .... with ... by ....": a number of prepositional phrases follow a verb or nounphrase, and the syntax cannot determine whether the second and third prepositional phrase is connected to the verb or one of the previous prepositional phrases, resulting in a reasonable number of possible combinations. In this example the semantics module will determine the correct connections using the selection restriction frames of the various words (see later in this section).

The parser consists of a syntactic module, a semantic module, and a postprocessor. The syntactic module consists of a submodule for a context-free analysis producing a surface structure and a second submodule which executes a transformational analysis. The first submodule is based on a context-free rewrite grammar according to the Extended Affix Grammar (EAG) formalism.[24] The grammar is converted into a parser by the parser generator GRAMMA.[24] The surface structure is a decomposition of the sentence into its syntactic categories (verb phrase, noun phrase, prepositional phrase, verb, etc.). It can be represented as a tree, see Figure 2 for an example (first decomposition tree). In that figure, S stands for sentence, NP for noun phrase, AUX for auxiliaries, VP for verb phrase, V for verb, etc. A number of intermediate nodes are added for grouping various nodes (syntactic categories) on various levels, some of which do not occur in the current sentence. The strings between the square brackets identify the various syntactic and semantic features of the nodes on the given positions or originate from the lexicon entries of the words. For instance,

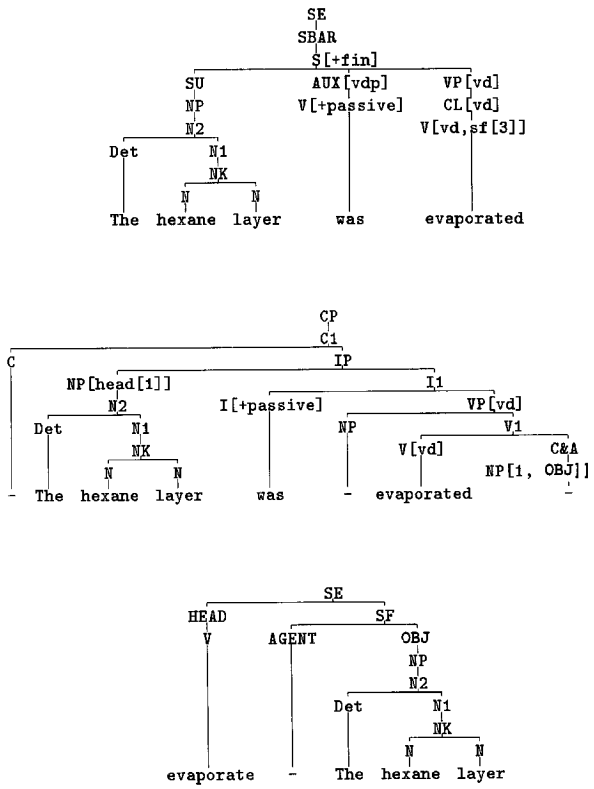

Figure 2. An example of the different intermediate results of the analysis of "The hexane layer was evaporated." by the parser, drawn as decomposition tree. The first tree is the surface structure, the second is the enriched surface structure, and the third is the semantic representation. The nodes are simplified by removal of a number of the syntactic and semantic agreement features. For meaning of the used abbreviations: see text.

vd stands for past participle and sf[3] stands for semantic frame number three, referring to a separately stored semantic selection restriction frame, which is consulted when needed.

The transformational analysis converts the surface structure to an enriched surface structure based on the principles of Government and Binding theory. This enriched surface structure contains added traces and empty positions linked to surface structure parts (noun phrase, prepositional phrase) which represent and, during the semantic phase, are filled with the semantic roles (representing the so called deep structure of the sentence; the semantic roles are called thematic roles as well). See also Figure 2 for an example of both structures. In the second structure empty nodes are added (the nodes that have a "−" at the sentence level) that identify possible role positions related to the main verb. For instance, the empty NP to the left of "evaporated" identifies the possible agent position and the empty NP to the right of the main verb identifies the potential object (or theme; also indicated by the "theta[OBJ]" feature). The links to the actual role fillers, which are also made during the transformational analysis, are indicated by the numbers in the node features ("1" references to "head[1]"). The CP and IP nodes originate from the Government and Binding theory. The third structure is a reconfiguration of the second one, in which the original positions of the role filling sentence fragments are transferred to the corresponding role positions and all behind the meaning kernel of the sentence (mainly the main verb). This is the result of the semantic phase. "SE" stands for sentence and "SF" stands for semantic frame. The transformational analysis and the semantic module are

implemented as transformational rules, which are applied by GRAMTSY. The rules consist of a structure description part, which should match (part of) a decomposed sentence structure, a condition part, which describes when to apply the rule and structure change part that defines how a structure is to be modified.

The syntactic module is largely domain independent but not completely because the grammar is developed for the given type of texts. One can argue, although, whether the possible domain specific constructions are really specific for this domain.

The semantic module produces a logical meaning representation of the original text from the enriched surface structure. The meaning representation is based on concepts and a limited set of relations. By means of general transformation rules and using the semantic classes and the selection restriction frames from the lexicon the semantic structure is produced. For the various roles defined in the selection restriction frame the semantic classes and prepositions (if applicable) are checked against those of the potential candidates in the sentence. During this process logical principles are taken into account as well, and intersentential referencing is marked and partly solved. The semantic module is domain independent: it takes care of the addition of the roles by application of the selection restriction frames related to the various concepts. The domain dependent selection restriction frames are taken from the lexicon. As is illustrated with the examples in Figure 2, the original sentence is gradually converted into a more abstract representation within which the semantic relations between (mainly) the verb (representing an action) and the other sentence components are revealed, firstly, by recognizing (mainly) the verbs, the noun phrases (and so grouping everything related to the nouns: its determiner and adjectives), and the prepositional phrases and then by restructuring the "tree", adding the semantic relations between the various clusters (verb phrases, noun phrases, etc.) and grouping everything around the verb as meaning kernel of the sentence. During the restructuring the initial syntactic information is gradually pruned (partly).

The semantic postprocessor converts the parse-tree, furnished with the thematic roles and semantic classes of the concepts, into the final meaning representation containing only the concepts, their semantic classes, the roles with which the concepts are related to each other, tense information, internal reference links, and logical operators. The last structure of Figure 2 will become like the following (the indentation is added for clarifying the nesting of the brackets; strings starting with capitals are quoted, because otherwise the pragmatic module will interpret them as variables):

```
[['SE', ['HEAD', evaporate],
    ['SF', ['OBJ', ['HEAD', layer],
            ['SF', [reference,def],
                ['CONT', ['HEAD',[hexane, [class, chemical]]]
                ]
            ]
        ]
    ]
]].
```

(The number of concepts (objects) is assumed singular, unless specified otherwise by means of e.g., "['NUM', plural]" within the SF frame.)

The semantic postprocessor takes care of the domain dependent structures like those that describe the composition of mixtures and solutions, as well. E.g., concentrations, masses, and volumes that go with chemicals are rewritten as attributive concepts with explicit identification of the numbers and units of those chemicals. An example of this notation is the representation of the phrase "0.1 N HCl" as

```
[['HEAD', ['HCl', [class, acid]]
],
['SF', ['ATTR', [concentration,'N'],
        ['SF', ['NUM',0.1]]
    ]
]].
```

In this notation "HEAD" identifies a concept and SF identifies the semantic frame that lists the information related to that concept. In the case of units HEAD and the class (type) of the unit is combined to a string identifying the type of attribute (based on the type of unit): concentration, volume, mass, or the more general "measure". The last module is programmed in SPITBOL.[26]

A detailed description of the parser components, the grammar, and the various development considerations can be found in ref 25.

After the parsing process the pragmatic analysis is executed. Its task is the construction of the "story" that is told by the abstract, using background information on analytical chemistry in general and on the various analytical techniques that are mentioned in the abstracts. This module is based on the theory of Conceptual Graphs developed by Sowa.[27] The attractive features of this theory are that it has a sound logical foundation and that it seems unlimited in the representation of conceptual structures. Others have implemented this theory for the purpose of natural language processing as well, see, e.g., refs 13 and 14. Their systems use the principles of conceptual graphs during the semantic phase of the parsing process as well as during the pragmatic phase. This seems more attractive than the approach taken by the authors, because a unified principle and system is used for semantics and pragmatics. The current choice is motivated by the starting points and previous experiences and corresponds with the division of the expertise that exists with the developers of the different modules: the parser largely makes use of linguistic principles and is developed by linguists, whereas the pragmatics module largely is based on knowledge processing and analytical chemistry, which is the knowledge area of the analytical chemists that develop that module.

Conceptual graphs (CG) are semantic network-like structures of concepts and relations. An example is given in Figure 3. This figure represents parts of the concept "determine" as it is used within the analytical chemistry domain. The concepts can be organized in hierarchies or ontologies. By representing all concepts (actions, objects, and properties), that play a role in the description of analytical methods, in this formalism, a knowledge-base is generated. This knowledge-base functions as background knowledge of the system during the interpretation process of the abstracts. (Part of the knowledge is used by the semantic module of the parser and is represented in the selection

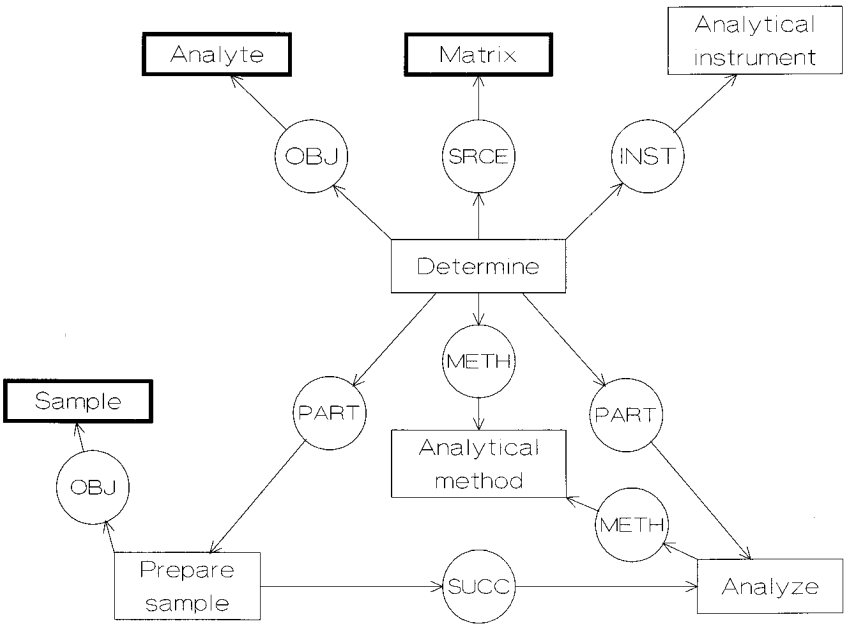

Figure 3. The definition of the verb determine. The relations are encoded within circles; the roles are marked with thick lines.

restriction frames and semantic word classes in the lexicon.) The software, that is used for the development of the discourse module, is developed during this project as well. It mainly consists of a so called conceptual graph processor that takes care of the conceptual graph handling (inheritance, matching, merging, etc). All the operations on graphs are clearly defined in ref 27.

The tasks of the module are as follows:

- −read in the output of the parser;
- −compare the input with the background knowledge of the system and generate a memory model of the input. During the generation of the memory model
  - −resolve intrasentential references using the referential links that are already made during the parsing process;
  - −resolve intersentential references;
  - −make the in the original text suggested implicit information explicit and mark lacking information;
  - −try to generate a logic flow of actions and, in fact, a model of the described analytical procedure;
  - −mark missing information and draw attention of the user to it;
- −write the extracted information (completed as much as possible) to a file in a systematic format.

These tasks are executed within the discourse module by procedures that activate in various ways the functions of the conceptual graph processor. These procedures are embedded within an overall routine that reads the input line by line and activates these procedures. Which procedure is activated depends on the relation (sometimes), its value (more frequently), or the current position within the abstract. For instance, there are special procedures for the resolution of explicit references, for certain general relations, for so called reverse relations (see later on in this section), and for verbs that indicate a relation (see later on in this section). A systematic format in which the extracted information can be written is described in a previous paper.[28]

The processor of the semantic representation, that is output of the semantic postprocessor, is programmed in NU-Prolog.[29] The knowledge-base is represented in the same language as Prolog facts.

The knowledge base contains the (analytical) concepts organized in a "is-a" hierarchy ("is a kind of" hierarchy). Two parts of it are given in Figure 4. Associated with most concepts are definition graphs that contain the information or definition of that concept. The various senses of some words are represented by as many concepts and their definitions. The concepts inherit the information contained in the definition graphs of its parents so that each definition graph only contains the information in which it specializes its parents and differs from its sisters. The hierarchy contains three main types of concepts: relations, roles, and the other concepts.

The relations are the earlier mentioned themes or roles that concepts relate to other concepts. Relations can be defined in terms of other relation or in terms of other concepts. E.g., the relation CONT, standing for "containing" (in, for instance, "a solution containing a chemical"): this relation can be defined in terms of the verb "contain" with an actor (or dative) and an object being the concept that contains and the concept that is contained, respectively. Relations have a direction, i.e., it is possible that some input concepts are linked to each other in the opposite direction. This is labeled by the same relation, but with a @ added to the relation name; they are called reverse relations.

Roles are the literal roles that concepts (frequently objects) can play in certain circumstances, remaining the same concept. E.g., the role "analyte": it is the role of a chemical in an analysis, but that same chemical could play other roles in other situations (being the titrant or the eluent, etc.). Within the conceptual graph processor roles have the semantics that they can be merged with any concept of the correct type, given that a merge is possible on the basis of the current environment of the concept (i.e., the conceptual graph in which it resides). The separate definition of roles prevents the definition of a role-version and a nonrole version of, for instance, each chemical object in the knowledge-base, which makes knowledge-base more manageable in size and maintenance. Roles play an important role in analytical

Figure 4. Part of the concept taxonomy, used by the pragmatics module. The concepts $CHEMICAL_ACT, $ANALYTICAL_ACT, $DETECT, and $END_POINT_DETECT contain more children. All end nodes correspond to an English word or term. E.g. $PH corresponds to pH, and $MIX1 and $MIX2 correspond to two different interpretations of mix, represented by two different definition graphs.

chemistry: they mark analytical interesting concepts that should be extracted and stored in a database and they are frequently used in abstracts.

The remaining concepts are those related to the verbs, nouns, adjectives, adverbs, etc. Linked to these concepts are definition graphs that in certain instances, more or less, play the role of schemata (or scripts) as defined by Sowa (see ref 27, p 129). The taxonomic structure of the ontology is mainly a tree. This type of structure is advised by Bouaud et al.[30]

The program acts as follows: the first concept of a sentence (or title) is read. If it is the first concept of the first sentence (title) its definition graph is copied to the so called focus (acting as a short term memory). Most of the time this is the word "determination", which sets up a graph expecting an analyte, a matrix, a technique, etc., see Figure 3. If it is the first concept of the other sentences, a procedure is started which tries to locate the best position in the current focus with which it can be merged. Then, recursively each combination of relation and concept is read from the input and compared with the definition of the current concept, which is a reflection of the concept in the input that "contained" the relation and concept. Normally, the class of the concept from the input is defined for the current concept (in its own definition or in one of its parents) in the focus along the same relation as well, and the definition graph of the input concept is added to (joined with) the CG in the focus along the input relation. If a definition graph cannot be joined with the focus, other definitions of the input concept (if available) are tried. If this fails, the program checks whether there are other definitions of previously merged concepts and these are tried out. If the concept is a reference to a previously mentioned concept, this reference is resolved. Sometimes the relation can be one of a more general nature, just linking two concepts with each other without exactly specifying what the exact relation is. In such a situation a search is performed which concepts and relations could link the concept from the input to the reflection of the concept, within which it occurs, in the focus. An example occurs in the sentence "The method is sufficiently precise and accurate for routine analysis" which gets the semantic analysis:

```
['SE', ['HEAD',['and',['HEAD',precise],['HEAD',accurate]]],
    ['SF', ['MOD',sufficient],
        ['DAT', [method, [reference, def]]],
        ['PURP',routine_analysis]
    ]
].
```

"DAT" stands for the dative role, "MOD" means modification and "PURP" means purpose. In this semantic analysis "routine analysis" is linked to the conjunction of "precise" and "accurate", but in the knowledge-base it is a role of an analysis or determination, namely an analysis that is used routinely. "Precise" and "accurate" are properties of analytical methods, and the search routine now tries to link the analysis mentioned in the "routine analysis" definition with the former method which is already referred to by "the method" in the sentence and which refers to the method (determination) that is the subject of the abstract and is already replicated in the focus.

If in the focus a role is defined, whereas the input gives a normal concept, the definition graph of the role as well as the graph of the concept are matched with and merged/joined with the focus on the given position. Verbs in the input that define relations (like "follow by", "contain") are normalized to that relation (if possible). The concepts and relations that explicitly occur in the input are marked in the focus (instantiated).

An example: given the title phrase and sentence

> "The determination of phosphorus in milk by electrothermal-atomization atomic-absorption spectrometry with L'vov platform and Zeeman background correction. The sample was mixed with 2 ml of a solution containing 6.25% La(NO3)3, and H2O was added to 25 ml."

the output of the parser will be

```
[['SE', ['HEAD', determine],
    ['SF', ['OBJ', ['HEAD', phosphorus]],
        ['SRCE', ['HEAD', milk]],
        ['METH', ['HEAD', 'EA-AAS'],
            ['SF', ['ATTR', [and,
                    ['HEAD', 'L' vov platform'],
                    ['HEAD','Zeeman background correction']
                ]]
            ]]
    ]
]].
[['SE', [and, ['SE', ['HEAD', mix],
            ['SF', ['MATR', ['HEAD', sample],
                    ['SF', [reference, def]]
                ],
                ['MATR', ['HEAD', solution],
                    ['SF', ['ATTR', [volume, ml],
                            ['SF', ['NUM', 2]]
                        ],
                        ['ATTR', ['HEAD', contain],
                            ['SF', ['OBJ', ['HEAD', 'La(NO3)3'],
                                    ['SF', ['ATTR', ['RELMEAS', %],
                                            ['SF', ['NUM', 6.25]]
                                        ]
                                    ]]
                            ]]
                    ]]
            ]],
        ['SE', ['HEAD', add],
            ['SF', ['OBJ', ['HEAD', 'H2O']
                ],
                ['TO', ['VOL', ml],
                    ['SF', ['NUM', 25]]
                ]
            ]]
]]].
```

EA-AAS is used as abbreviation of electrothermal-atomiza-

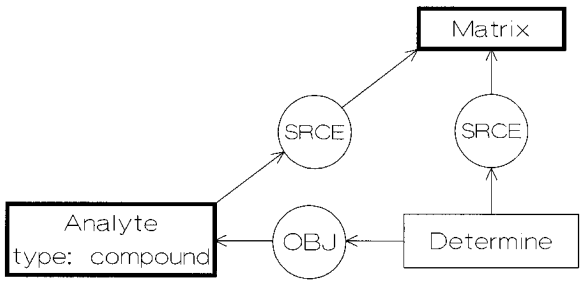

Figure 5. The definition graph of the role analyte.

tion atomic-absorption spectrometry. The "ATTR" relation within this concept is somewhat odd given its values: something like the INST (instrumental) role would seem more appropriate. Still ATTR is chosen because this was the easiest choice for the current version of the parser: everything related to objects is linked with an ATTR relation (this is appropriate for volumes and concentrations, which are frequently related to objects in the analytical chemistry domain). The sentence is a conjunction of the verbs "mix" and "add", each with their semantic SF frame. The "TO" relation and value in add represent the end volume of the addition.

The program first encounters determine. The definition of determine is retrieved from the knowledge base (see the graph of Figure 3). First a rough comparison of the input concept and its relations with the determine definition from the knowledge base takes place (in order to select the proper meaning in case of homonyms). Then this graph is copied to the focus. Determine contains a SF frame, so it contains a number of relations and values which can be added to the determine graph in the focus. The first relation is "OBJ". The determine graph in the focus contains this relation as well (otherwise the parents of determine would have been consulted), and the knowledge base is consulted for the definition of "phosphorus". For most chemicals only their type is stored in the knowledge base. The retrieved type is "inorganic compound" and then is checked whether this type is of the same type or more specific than the one defined for the "OBJ" relation in the determine graph. The defined value restriction is "analyte". This is a role type, so the definition graph of analyte is consulted for its nonrole type (see Figure 5). The retrieved type "compound" is, according to the hierarchy of the knowledge base, more general and the definition of phosphorus is allowed to be merged with the analyte concept in the focus. Because the analyte concept in the focus in fact defines a set of concepts (there may be more than one analyte), first a copy of this concept is made. Then the definition graph of analyte is merged with this copied analyte concept in the focus. After that, the definition of phosphorus is merged and the phosphorus concept is instantiated. The result can be seen in Figure 6. In the same way the "SRCE" relation and value and the "METH" relation and value in the input are compared with the corresponding definitions in the focus and knowledge base and merged with the focus, see Figure 7. The "EA-AAS" definition is in fact also a child of determine (it contains a determine concept) and it contains information concerning this technique (see the EA-AAS analyze concept and its parts "heating cycle", "absorption detection", and "calculate concentration"). By merging this definition with the focus, the point of view is limited to this technique. The act of merging (joining) is also exemplified by this figure. The EA-AAS definition contains concepts which are children of concepts existing within the determine definition on corresponding positions (see Figure 3). It contains EA-AAS, which is a child of "analytical method", "EA-AAS determine", which is a child of determine, and EA-AAS analyze, which is a child of analyze. By merging the EA-AAS definition graph with the analytical method concept of the focus (see Figure 6), analytical method is specialized to EA-AAS, determine is specialized to EA-AAS determine, and analyze is specialized to EA-AAS analyze. Beside it, the concepts heating cycle, absorption detection, and calculate concentration are introduced.

In the output of the parser EA-AAS contains an ATTR relation with a conjunction of "L'vov platform" and "Zeeman background correction". In the knowledge base they are stored as "instrument part" within the EA-AAS determine concept (which is a simplification, but suitable enough for abstract like texts). A special procedure takes care of this situation: if a concept is not located with the same relation on the corresponding position in the knowledge base (i.e., in EA-AAS) and this relation is ATTR, then a search is started in its direct environment for a possible matching concept. The conjunction is processed by storing each concept within the conjunction in a separate "INST" relation (referring to the default "and" interpretation of the program).

The sentence consists of a two actions within a conjunction. The first concept that is read from the input is and. A separate procedure takes care of the conjunction; in case of actions normally the conjunction is skipped. The next concept that is read is mix. This concept does not occur in the current focus and neither does its type. In such a situation a procedure is started that scans the knowledge base, starting from the concepts in the focus, for a possible connection. Starting from determine the procedure checks its related concepts and their definitions for a matching concept. The first possible match occurs within the definition of "sample prepare" which can be a PART of determine. In that definition, participating actions (within PART slots) are defined of the type "chemical actions", and this type agrees with the type of mix. The result is that first the definition graph of "sample prepare" is matched and merged with the sample prepare concept in the focus, and then the mix definition is matched and merged with a copied singular PART chemical action. Its OBJ value sample is a definite referent (which is indicated by the parser by [reference, def]) and this triggers the program to search for an antecedent of the same type within the focus. Initially, the instantiated concepts are scanned (explicit references), but when this fails the other concepts existing within the focus and their definitions are checked. In this way, implicit references are accounted for as well. In the current situation the sample prepare concept in the focus contains a sample concept as OBJ value, and this concept will be used as implicit antecedent. The "solution" concept is processed in the same way as described before. The notation '['ATTR', [volume, ml]' is an abbreviation of '[ATTR', ['HEAD', volume], ['SF', ['UNIT', ml], ['NUM', ...]]]' and is processed as such (see Figure 8). The next concept within solution is "contain". This verb is marked in the knowledge base as being part of

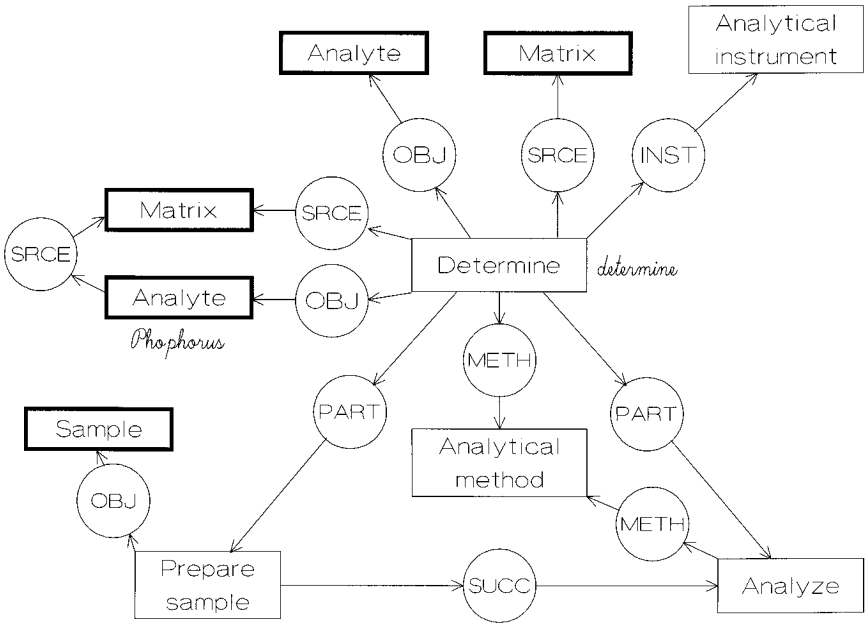

Figure 6. The focus as result of the merge (join) of the analyte definition (Figure 5) and the phosphorus definition with the analyte concept in the determine definition graph (Figure 3), which was already copied to the focus. The instantiated information is written in italics.

the CONT (contains) relation definition:

definition of [object X] → (CONT) → [object Y] =

[object X] ← (AGNT) ← [contain] → (OBJ) → [object Y]

(This definition can also be written in a format similar to that of the input, without SE and SF as

```
[contain,
    ['AGNT', [object X]],
    ['OBJ', [object Y]]
]. )
```

One of the procedures of the pragmatic analysis program takes care of the processing of these verbs, first by processing them as normal verbs and then by replacing them with the corresponding relations. The "AGNT" of contain fails. Another procedure, triggered by this situation, resolves this by taking the concept in which contain resides as AGNT value. Then the OBJ relation and value from the input are processed. This is "La(NO3)3", and its definition will be merged with the object after the CONT relation. The percentage of La(NO3)3 is processed in a similar manner as the volume mentioned before. The "%" is interpreted by the parser (the semantic postprocessor) as unit of "RELMEAS" being a relative measure. In the knowledge base a relative measure is a parent of concentration (see Figure 4). The concentration concept is linked to more specific concepts than those linked to relative measure and these concepts correspond to the types of those in the input. The program will automatically specialize the RELMEAS concept from the input to a concentration.

The next concept in the input is add. The processing procedure is not different from a situation in which both actions occurred in separate sentences. The normal procedure is that a new action will be stored after the previous one within the same procedure describing action (in this case sample prepare), as long as it fits with the defined type (and as long as the definition allows that more "PART"-s may be added). If this is not possible, the program searches for other connection points (matching concepts) within the current focus. Add is a child of the PART value chemical action of sample prepare and its definition will be inserted in the focus. Its OBJ and value and "TO" plus value are processed by the normal procedures. The "GOAL" of add is absent in the input, but is semantically implied as phrased by the sentence "... and H2O was added to the resulting solution to 25 ml.". The quote "the resulting solution" refers to the product of the previous action. A separate procedure takes care of these implied references and automatically generates a "RSLT" relation and value concept and links this concept to the GOAL relation of add.

The resulting focus graph is given in Figure 8. If a concept defines a set then a singular version is copied and used for processing of the input leaving the other concept as definition. In this way care is taken of sets and members. This representation is an approximation of a representation using contexts and is suitable enough for the current purpose. The instantiated information can be written to a separate file or database in a format like

technique: EA-AAS (or the full term)
analyte: phosphorus
matrix: milk
working range: ....
etc.

for the method parameters and in the referred systematic representation for the procedural part (the actions).

RESULTS AND DISCUSSION

The selected set of abstracts are selected from Analytical Abstracts online using the keystrings "determin" and "analy" and four techniques (HPLC, AAS, ICP, and titrimetry). Of each technique up to 40 of the most recent abstracts were stored on disk. The search took place in 1990.

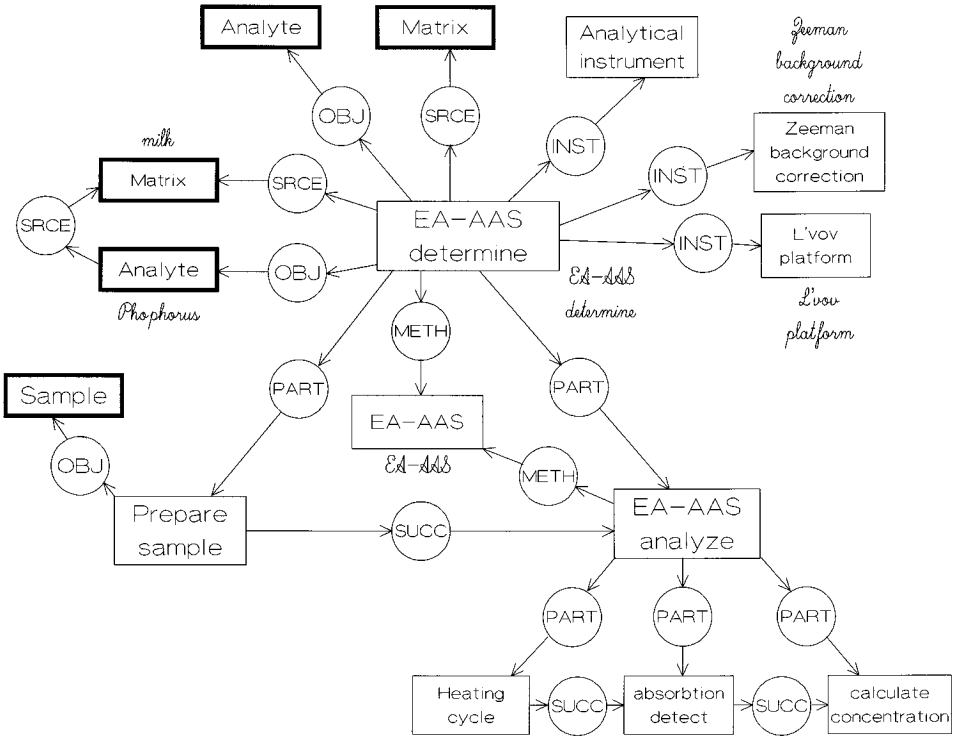

Figure 7. The focus as result of the application of the discourse analysis module on the title phrase "Determination of phosphorus in milk by electrothermal-atomization atomic-absorption spectrometry with L'vov platform and Zeeman background correction". For meaning of the various symbols: see Figures 3 and 6.

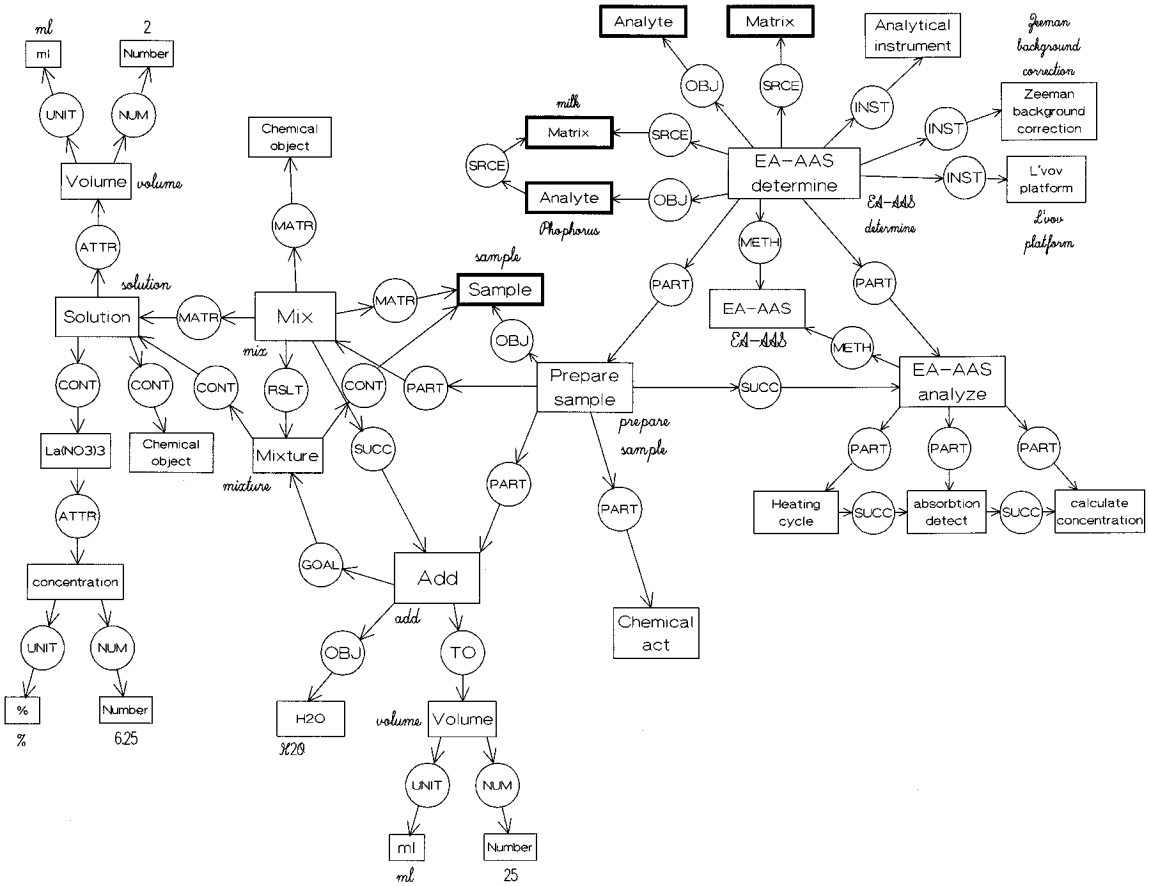

Figure 8. The focus as result of the application of the discourse analysis module on the example given in the Theory and Implementation section. For meaning of the various symbols: see Figures 3 and 6.

The current status of the lexical module is that it is finished for the initially chosen test-set of abstracts, containing (including the titles) 1806 sentences and 16 939 strings. After removal of all more than one time occurring strings 4049

TI Hexa-amminecobalt(III) tricarbonatocobaltate(III) as a redox titrant for the determination of certain thioxanthene derivatives.
AB Chlorprothixene (I) and thiothixene (II) sample soln. were prepared in 0.1M-HCl, and a portion containing 2 to 15 mg of I or II was mixed with 25 ml of 10% H2SO4 and two drops of ferroin indicator, and titrated with a 5mM soln. of the cited reagent (III) until the colour changed from red to pale blue. A portion of aq. soln. containing 2 to 15 mg of methixene hydrochloride IV was mixed with 25 ml of 40% H2SO4, and the soln. was titrated with 5mM-III until the orange colour disappeared. Recoveries of I, II and IV were quantitative and the coeff. of variation were 0.89, 0.67 and 0.99%, respectively. The method was applied to determine I, II and IV in dosage forms and results agreed with those from the official method.

Figure 9. An example of the abstracts that can be analyzed by the complete system, including the title (reprinted by permission of The Royal Society of Chemistry).

strings remain. The final lexicon contains 1896 entries.[18] Only a part of the entries are supplied with semantic information.

The current status of the syntactic modules is that they are finished, being capable of analyzing all sentences. It contains 410 rules for the rewrite grammar and 31 for the transformational grammar. The results are better than the results of a prototype system published by Zweigenbaum.[16] Their system gave for 60% of the sentences of 475 patient discharge summaries a syntactic analysis. Their problem is the type of sentences occurring in these summaries: frequently ill-structured. They expect to obtain a maximum of 80% fully parsed sentences. A check after the correctness of the analyses by our grammar is being performed; the results will be published in van Bakel.[25]

The semantic module is almost finished. At the moment it is capable of analyzing a limited set of sentences with a limited structure. The semantic postprocessor is finished.

The pragmatics module is capable of analyzing six complete abstracts within the domain of AAS, HPLC, and titrimetry (for the abstract content and bibliographic information, see refs 32−37; the given abstract numbers encode volume, issue, section, and order number; more on the characteristics of these abstracts is given toward the end of this section). One of the more difficult abstracts of this set is given in Figure 9 (ref 34).

The selected type of texts implies a number of difficulties. One of the requirements of abstracts is that they should be concise. On the one hand, this frequently turns out to result in telegram-style sentences, and, on the other hand, this frequently results in numerous conjunctions of (verb) phrases. In these conjunctions sometimes concepts are linked to other concepts that have hardly any relation to each other. An example of the latter situation is "The coeff. of variation (n=5) at a recovery of .simeq. 100% was 0.75%." [36] (.simeq. is the Analytical Abstracts code for ±). The former result poses problems to the parser because of ill-structured sentences for which rules have to be adjusted. Still, it turned out to be possible to develop grammar rules for those sentences. The latter situation posed problems for the pragmatics module. Normally, for such a type of prepositional phrase the relation ATTR (attribute) would be selected, but for the linked concept that would not make sense. As a consequence of this, a general relation is introduced that should be applied in such situations. That relation triggers the pragmatics module to look for a wider relationship between both concepts (up to: occurring at the same time anywhere in the same focus).

Another problem connected to the specific type of texts and the (analytical) chemistry domain is the procedure for recognition of the sentence end. Analytical abstracts contain dots within numbers, chemical formulas (e.g., CuCl2.2H2O) and around the notations used by Chemical Abstracts Online for non-ASCII characters (e.g. .+−., .gtoreq. used in conjunction with numbers). The full stop for a sentence end need not be followed by a space or new line. A special procedure was developed that took care of the aforementioned situations (see the Theory and Implementation section).

The Lexical Module. The terminological aspect of especially compound words required special attention. Frequently compound words in literature abstracts can be viewed upon as one term, although this term is not formally defined. People continuously invent new terms (frequently based on previous ones) for, for instance, new and/or hyphenated techniques, and it takes some time before these terms get an official status by recording in some compendium of nomenclature (e.g., the Compendium of Analytical Chemistry[38]) or thesaurus of, for instance, Chemical Abstracts.[39] A search was undertaken after possible rules and/or standard procedures for the identification of terms, but they were hardly available. The best (informal) procedure turned out to be the following: a group of words is identified as a (new) term if it is one of the main subjects of a paper (abstract) and its authors identify them as representing a concept that has added meaning with respect to the combination of meanings of the individual words. During the development of the lexicon the procedure was to label each word with the appropriate syntactic (and semantic) word-classes. Beside this, all the abstracts were scanned for compound words, and they were stored in a separate lexicon. First a file was set up containing all potential compound words (each group of words consisting on adjacent nouns with possible co-occurring adjectives), and then the real compound words were marked by an analytical chemist using thesauri, the Compendium of Analytical Chemistry[38] and the aforementioned procedure. In the future, a possible automatic term recognition procedure can be as follows: first marking potential compound words on syntactic grounds and then automatic searching for these terms in the term database of, for instance, Chemical Abstracts. If a potential compound word is not recognized this way the user can be consulted. A possible failure of the syntactic analysis of the sentence in which it occurs can be grounds for consulting the user in this respect as well. Ter Stal and van der Vet[40] worked out a procedure for the processing of a part of two-component compound nouns occurring in their corpus of material abstracts. Still, their procedure does not process chemicals and a considerable part of the other compound nouns (requiring too much background/domain knowledge and inferencing); they are lexicalized. This seems to necessitate manual assistance to the lexical phase as well. Concerning the construction and maintenance of the lexica: in the future online thesauri and computer readable dictionaries can be consulted or linked to the system. The same holds for the classification and possible conversion of chemical names and formulas. There exist programs for these tasks,[41] and Chemical Abstracts Online can be consulted for the assignment of CAS registry numbers. The selection restriction frames are translated from the knowledge-base of the pragmatic module (using the conceptual graph definitions of the corresponding concepts). At the moment this translation is done by hand, but in the future this translation can largely be performed automatically.

The Parser. The development of the grammar took extra effort because the telegram-style of the abstracts. Other

sources of difficulties were the occurrences of long sentences consisting of numerous conjunctions of verbs, which sometimes were combined with subordinate clauses. The difficulty then is to produce the proper analysis and only one analysis as well. The same problem of exponentiality does occur with these constructions as with prepositional phrase sequences: are the phrases located beside each other and/or are they embedded in each other. In case of prepositional phrases this is solved using underspecification (see the Theory and Implementation section). All prepositional phrases are linked on the same level to the main verb and by application of the selection restriction frame of the main verb and those of possible nominalizations they are assigned to those words. If there are prepositional phrases left, then they are assigned to the noun phrases located before them in the sentence. In case of conjunctions of a number of verbal clauses (verbs) and subordinate clauses the parser produces a number of solutions from which the correct one is selected by hand. The parser will be optimized in this respect.

Errors in the abstracts hinder a successful analysis of sentences. Observed were ill-spelled words (e.g., aliquat, neubulizer; the number of errors was relatively low: six were observed in the selected set of abstracts), and repetition of words (as in the title phrase "Determination of rimifon (isoniazid) by alternating-current by oscillopolarographic titration" [35]). In the abstract with ref number 34 the parentheses around a Roman number that should be used as reference to the accompanying chemical lacked. In the same abstract, in the title, a space missed between a chemical and its internal reference number, suggesting that the Roman number was its charge number. These errors were removed beforehand or during the grammatical analysis (observing analysis failures). In the future, with new texts, most of the times these errors will be observed during the morphological and grammatical analysis and can be repaired online.

The semantic module is partly finished. The subclauses between parentheses are not processed but passed to the semantic postprocessor. Some of these subclauses have a typical domain specific structure; others are more sentence like. It will take more effort to develop adequate rules for the latter type of subclauses. The same holds for range indications ("100 to 700"). Conjunctions and disjunctions of words are still partly processed: only the first word is semantically linked to the other constituents of the sentence. The processing is finished by the postprocessor. Empty referents between conjunctions of verbal clauses are not resolved (e.g., in the sentence "A solution is mixed with .... and .... is added." implicitly the words "to the solution" are assumed at the end of the sentence, being the goal of the addition and referring to the first solution). This is done by the pragmatics module. In the future, the syntactic module should make the proper links in case of subjects which are left out in conjunctions of verbal clauses. The semantic module does not make use of the concept taxonomy (hierarchy) of the pragmatics module. At the moment it uses a rough classification of concepts (human, not human, concrete, not concrete), which was sufficient up to now (during the application of the semantic selection restriction frames).

The semantic postprocessor is mainly domain specific. The conversion of the parse-tree to the above given semantic representation is a normal general task within a parsing process. Besides the conversion of the measure indications that go with chemicals (concentration, volume, mass), it takes care of a number of structures that are given between parentheses as well. They are given a semantic analysis and representation on a partly ad hoc basis. These structures are, for instance, "(1 to 225 ppm)", "(25 cm .times. 4.6 mm)", "(70:29:1)", "(n = 5)", and "(with and without flow of argon)". It also takes care of those structures that still cannot completely be processed by the semantic module and occur in the six abstracts that can be processed by the pragmatics module.

Before the actual discourse analysis another domain specific conversion takes place: descriptions of chemicals, together with concentrations, are converted to solutions of those chemicals having the given concentration. During the discourse analysis these solutions are supposed to be aqueous unless the contrary is given in the texts.

The Discourse Analysis Module. The discourse analysis module is a partial implementation of the Conceptual Graph theory of Sowa. The system lacks schemata and formal deduction based on rules that are written as conceptual graphs. The former is up to now viewed upon as not necessary because the definition graphs of concepts frequently act as schemata as well. They at the same time define concepts and set up expectations for other concepts to exist in their neighborhood along certain relations. This can, for instance, be seen in Figure 7. The "EA-AAS analyze" concept (the concept representing the actual instrumental analysis using an electrothermal-atomization atomic-absorption spectrometer) contains a heating cycle, an absorption detection, and a concentration calculation concept, and these concepts contain details concerning drying, ashing, atomization, wavelength settings, etc. In this way all necessary implicit analytical knowledge is available and words, occurring in the sentences, are linked to the concepts in the appropriate environment (context). A second motivation is that the selection restriction frames of certain concepts in the lexicon can, more or less, be viewed upon as the definition graphs of concepts, combining the right concept to the right (group of) word(s) during the parsing process. The selection restriction frames give the necessary conditions. Deduction of conclusions based on the information originating from the abstracts is done by means of Prolog rules (procedures) that use the graph-matching and graph-modification procedures of the conceptual graph processor. These rules can be fired during the working up of the input (i.e., output of the parser) or after the working up of each sentence. Examples of the latter are procedures that try to recognize roles in the focus-graph and label the appropriate concepts with these roles and procedures that try to simplify the focus-graph by recognizing definition graphs of relations and exchanging the appropriate concepts by the relation. According to Schröder[15] the method suggested by Sowa[27] proved to be unpracticable. His system (as well as the system of Fargues et al.[13]) uses Prolog-like inference rules with conceptual graphs as terms. Such an inference procedure would be useful for the current system.

The reason for not using an already existing Conceptual Graph processor for the discourse processing module is that most of the existing current implementations are partial implementations as well,[42] and if one experiences a shortcoming of an implementation that turns out to be necessary, then it is very time-consuming to modify the program of

others (if possible). Furthermore, most of the text-analysis programs that use Conceptual Graphs differ from our implementation: they apply these graphs during the parsing process as well, partially integrating the semantic analysis and the discourse analysis. A general usable Conceptual Graph processor is under development but not finished yet.[43] A very attractive feature of this program will be a graphic representation and editing module of the knowledge base. Such a module would enhance and speed up the development of a knowledge base, as was experienced during the development of the current knowledge-base. A useful type hierarchy has a considerable depth and width, and some graph definitions contain so many concepts and relations that a textual representations does not present the desired overview. At the moment an alpha-numerical tree-drawing program is used for the visual representation of the type-hierarchy, and a drawing program is used for the manual graphical representation of the most complex graphs. A number of potential improvements to the current system will be not be implemented if the general usable Conceptual Graph processor becomes available in a reasonable amount of time.

The development of the knowledge-base is very time-consuming and frequently the resulting representation (of the background knowledge) of a certain concept is the result of a number of considerations. This is experienced by, for instance, Schröder[15] as well. Bouaud et al.[30] classifies the work as handicraft. Zweigenbaum[16] states that the knowledge development process has proven to be error prone and time consuming. Our current version of the taxonomy is specific for the domain at hand and not finished. For instance, the verbs are not classified in detail according to the various time aspects of verbs (processes) or classified within classes like “mental action” and “physical action”.

A starting point for the development of the knowledge base would have been the use of knowledge bases or ontologies developed by others. Until recently the complete structures and contents of knowledge bases were hardly available. There exist a variety of ontologies for different domains.[44] As far as is known there is no one oriented to analytical chemistry. Another drawback of the various available ontologies is that they are represented in various formats and are based on various structuring principles. Still, parts of published ontologies could have been adopted as a starting point of the current knowledge-base. This is, for instance, done with parts of the one given by Sowa.[27] The use of larger parts of ontologies and knowledge bases can have the drawback that one invests as much time in modifying it to the current domain and structure as one would have invested by developing an new version, and because of these drawbacks a new knowledge base is developed. A potential useful structure of knowledge about substances would be the one developed by Tepfenhart.[45] Still, this structure is not adopted because of its complexity and because its full structure is not published yet. The set of relations is given in an earlier publication.[28] Starting points for the set of relations were those published by Sowa.[27] Recently, initiatives are developed to make ontologies generally available in a limited set of general accepted formats. Partial use of existing ontologies is recommended by Bouaud et al.[30] In their paper discussions on the reuse of ontologies are given, together with principles for the development of ontologies and requirements of their structure.

As was mentioned before, the ontology mainly has a tree structure. Deviating from this structure is the taxonomy of (chemical) substances, which has become a tangled hierarchy on certain positions. Defining and interpreting substances was not straightforward. For instance, the word “mixture” can refer to solutions, a mixture of solutions, or a homogeneous or inhomogeneous mixture of solvents or solids, each being mixtures as well, etc. Sometimes a definition is recursive. E.g., solutions can contain solutions (after the addition of one solution to another). Still, these definitions are sufficient for the interpretation of the current abstracts, because in these types of text the use and description of substances is not too detailed.

The interpretation of chemicals given in the abstracts is based on a number of clues: those chemicals that are given with mass or volume are interpreted as liquids or solids (substances in general), the chemicals that are given with concentrations are interpreted as (aqueous) solutions, and otherwise they are interpreted as compounds or elements unless other clues from the text suggest another interpretation (for instance, as participant of the verbs add or mix which indicate that substances are meant).

The relations used are one- or two-ary. Up to now this has been sufficient.

Some concepts and relations that are part of the concept definitions cannot be classified as essential properties of that concept. They are more peculiar properties. Still, they are given in the concept definition because they can occur and because of that (and because of the program structure) need to be given in the concept definition.

The system almost lacks temporal aspects. The order within which actions occur in definitions implicitly indicate their order of execution in the time-domain, and this order is maintained during the use of these definitions. The same holds for the order within which the different actions occur in the abstracts, as long as this order is not in contradiction with the order indicated in the concept definitions. The ordering can be stressed by the relation “SUCC” (meaning “is succeeded by”), which is also the interpretation of the phase “..., followed by ...”.

The use of roles proves to be useful during the discourse analysis process. In the knowledge base and in the focus they label a number concepts interesting for extraction. In the focus concepts are becoming labeled by the roles in three ways. Firstly they can be defined in the knowledge base for those types of concepts on the given position in a graph and because the graph is joined with the focus (as a result of the occurrence of the concept to which the graph belongs in an input sentence) they are copied to the focus. Secondly they can be literally mentioned in the input sentence and thirdly they can be automatically recognized in the focus by matching the role-definitions with the focus. This matching is triggered by the occurrence of concepts in the input sentence whose type is compatible (the same or more specific) with the type of concept that can have a certain role according to the role-definition. This matching is performed after the processing of each sentence.

The reference resolution is partly facilitated by the application of schemata-like concept-definitions. A number of implicit and explicit references are already resolved before they are processed because of the joining of the various

schemata of the input sentence occurring concepts. This is recognized by Willems[31] as well.

The influence of the sublanguage used in these abstracts is twofold. There appear a number of constructions which are typical for these types of texts (e.g., the codes for non-ASCII symbols, the molecular formulas and the quantity, concentration, and mix ratio indications) for which extra procedures had to be added (mainly located in the domain specific parts of the program; see above). On the other hand, the sublanguage alleviates the processing of the texts by allowing less possible interpretations of words (meanings). This results in, for instance, a smaller knowledge base and less selection restriction frames from which the proper one has to be selected (which depends on the context and the determination of the context is much simpler). Certain words can get a more domain specific classification, which facilitates the interpretation of the abstracts. For instance, verbs like add and mix are classified as chemical actions, and this assists in the positioning of these verbs within the sample preparation procedure. And the OBJ (object) of the verb determine in the title can directly be classified as the analyte (or analytes). The requirement that the system should be as much as possible domain independent is not in contradiction with foregoing. A lot of language constructions are general and can be processed using general linguistic theory (which is one of the starting points of Chomsky's principle of Government and Binding). During the development of the parser rules developed by other people are implemented. And the (language independent part of the) parser can be used by other people as well.

From the six abstracts that can be analyzed at the moment by the complete system, four are selected from the initial set of 124 abstracts to be containing a representative set of textual difficulties (see refs 32−35 for their Analytical Abstract numbers). They contain explicit and implicit references, explicit references to chemicals by means of Roman numbers, errors, telegram style sentences, sequences of conjunctions and disjunctions, improper linking of concepts, and "respectively" constructions and sequences of prepositional phrases. They cover three of the four analytical domains: AAS, HPLC, and titrimetry. The other two abstracts (see refs 36 and 37) are of the same kind, structure, and analytical domain as one of the four abstracts of the set of six. These two were used to have an indication of the capabilities of the system. They were analyzed after the system was capable of analyzing the first abstract of the same type. It turned out that they could be analyzed without modifying the system, but the knowledge-base for the concepts it did not contain. These results give an indication of the current status of the system: the core of the system is finished, and all effort is now given to the expansion of the lexica and knowledge base.

The output can be in the Theory and Implementation section referred to as systematic representation for the procedural part of method descriptions. Currently the instantiated focus is used to control the performance of the pragmatics module. Checked is whether it gives a correct representation of the abstract content, i.e., whether all concepts in the sentences are interpreted and positioned correctly (e.g., within the sample preparation) and whether all (reference) links are made.

Information on the accuracy (possible measures are given in refs 46 and 47) of the complete system cannot be given at the moment: the system is developed using a subset of the selected set of abstracts. The aforementioned six abstracts are processed correctly, given that for one sentence (containing a conjunction of a number of verb clauses) the correct analysis is selected by hand. One hundred percent coverage with 100% accuracy cannot be obtained given the type of text: it is sometimes quite a puzzle for a human to fully understand what is meant. The results of other systems indicate that further research is still necessary. A comparison of the performance of a series of natural language processing systems in 1991 revealed that for relatively free text (news articles on a specific subject) an average of 26% recall and 52% precision could be obtained.[46] A more recent investigation[47] showed figures of 46 and 52%, respectively. These figures were the means of the three best performing systems on one domain but cannot directly be compared with the older figures, because the information extraction task that resulted in the latter figures was evaluated to be more difficult and because the metrics used slightly changed. On the same corpus four human analysts obtained 77% recall and 79% precision. The type of texts is roughly similar to our abstracts, but their corpus is far more larger and seems more diverse. Nishida et al.[3] did not present figures on the results of information extraction from abstract on semiconductors and patent claim sentences. Concerning chemical texts Ai et al.[4] obtained extraction results of 60−90% depending on the complexity of the texts (without specifying to which measures these figures refer). Their type of texts (synthesis instruction paragraphs) is more simple than abstracts. Zweigenbaum[16] only lists a number of intermediate results of a prototype of the METEXA-system. Of a corpus of 475 English patient discharge summaries about 60% of the sentences obtained a full syntactic parse. A smaller French corpus gave comparable results. The complete information extraction system could identify in one test summary all the required information. He expects to obtain a maximum of about 80% of fully parsed sentences, and because of this the system can semantically process partial analyses as well. Given these intermediate results the intermediate results of our system can be classified as satisfactory.

Beside the accuracy that can be obtained with this information extraction program the applicability of the program rises or falls with the quality of the sources of information (e.g., abstracts and research papers). A previous paper on this subject[12] does not present much hope on this point. The selected corpus of abstracts contains typical examples on this subject as well. For instance abstract AAN4910C00023 8907[48] is hardly informative. A comparison of recent manuscript requirements (author instructions) of a number analytical research papers[49−54] with those checked in the aforementioned paper reveals that little has changed. The Analyst is still most complete and even improved. The Journal of Chromatography now gives detailed requirements concerning the "Material and Method" section but still does not mention figures on the performance of the method. The other journals hardly mention any requirements on the method description and its performance. As said in the introduction the program is not meant to be dedicated to information extraction from abstracts. Other application areas are the control of the content of abstracts, the extraction of the core of analytical method characteristics from abstracts, information extraction from larger method descriptions like those published by the AOAC[55] or ISO

standards and control of method descriptions on completeness.

Previous publications[28,56] have demonstrated the application of the acquired information for graphic representation and its usefulness for the control of analytical procedures. The information could be used to guide (instruct) analytical workstations, like the ones given in refs 57, 58, and 59, as well. A future possibility can be to represent (parts of) the extracted information according to the standards that are being developed within the ADISS project.[60]

Given the difficulties that are experienced during the development of the parser and the discourse module, the question arises whether other techniques would give better results. The current system is in the development phase, but the parser seems to be adequate. A complete other technique that is not based on the application of grammar rules is the application of Neural Networks. A recent publication[61] shows that this technique is still not useful for real text.

CONCLUSIONS

The development of a system for (semi-) automatic information extraction from texts is still a time consuming task. The initial starting points have proven to be useful up to now. The modularity facilitates the development of the different modules by people who have knowledge on those modules (i.e., linguists and chemists). The domain independency is facilitated by the modularity as well: at the moment only the lexicon and the knowledge base have to be extended. Concerning the theoretical foundation can be mentioned that the parser proves to be capable of analyzing the complex sentences of abstracts.

The system could be used for the information extraction from for instance the "Material and Methods" section of research papers. These texts are more simple than abstracts, but there is large risk that parts of relevant information are located in other sections of the papers (see ref 12). Although information extraction is possible from graphs and tables[62] this will require reasonable effort and research. The suggestions given in ref 12 could ease the task in this respect.

ACKNOWLEDGMENT

The authors wishes to thank A. J. Kemperman for his contributions to the pragmatic part of the system and P. A. Coppen for his advice and contributions to the parser.

REFERENCES AND NOTES

(1) The Gopher Internet Information System is a collection of information servers accessible via Internet using for instance Gopher, the Internet Gopher Development Team, University of Minnesota, Minneapolis, and is based on the Internet Gopher Protocal, gopher://boombox.micro.umn.edu/00/gopher/gopher_protocol/DRAFT_Gopher_FYI_RFC.txt (1993).

(2) The World Wide Web is a collection of information servers accessible via Internet using for instance the Mosaic software of the National Center of Supercomputer Applications, University of Illinois, Urbana-Champaign.

(3) Nishida, F.; Takamatsu, S.; Fujita, Y. Semiautomatic indexing of structured information on text. *J. Chem. Inf. Comput. Sci.* **1984**, *24*, 15−20.

(4) Ai, C. S.; Blower, P. E.; Ledwith, R. H. Extraction Of Chemical Information from Primary Journal Text. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 163−169.

(5) Chowdhurry, G. G.; Lynch, M. F. Automatic interpretation of the texts of chemical patent abstracts. 2. Processing and results. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 468−473.

(6) Mars, N. J. I.; van der Vet, P. E. A semi-automatically generated knowledge base for direct answers to user questions. In *TKE '90: Terminology and knowledge engineering*; Czap, H., Nedobity, W., Eds.; Indeks Verlag: Frankfurt am Main, 1990; pp 352−362.

(7) van der Vet, P. E.; Mars, N. J. I. Structured system of concepts for storing, retrieving, and manipulating chemical information. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 564−568.

(8) Postma, G. J.; van der Linden, B.; Smits, J. R. M.; Kateman, G. TICA: A System for the Extraction of Data from Analytical Chemical Text. *Chemometrics Intelligent Laboratory Systems* **1990**, *9*, 65−74.

(9) Ginsberg, A. A unified approach to automatic indexing and information retrieval. *IEEE Expert* **1993**, *8* (5), 46−56

(10) Myaeng, S. H.; Khoo C.; Li, M. Linguistic Processing of text for a large-scale conceptual information retrieval system, Conceptual Structures: Current Practices. Second International Conference on Conceptual Structures, ICCS'94; College Park, MD, U.S.A., August 1994; Proceedings; Springer-Verlag: Berlin, 1994; pp 69−83.

(11) Analytical Abstracts; The Royal Society of Chemistry: Letchworth, Herts, England.

(12) Postma, G. J.; Kateman, G. The Quality of Analytical Information Contained within Abstracts and Papers on New Analytical Methods. *Anal. Chim. Acta* **1992**, *265*, 133−155.

(13) Fargues, J.; Landau, M.-C.; Dugourd, A.; Catatch, L. Conceptual graphs for semantics and knowledge processing. *IBM J. Res. Dev.* **1986**, *30*, 70−79.

(14) McHale, M. L.; Myaeng, S. H. Integration of conceptual graphs and government-binding theory. *Knowledge-based System* **1992**, *5*, 213−222.

(15) Schröder, M. Knowledge-based processing of medical language: a language engineering approach. In *GWAI-92: Advances in Artificial Intelligence, Lecture Notes in Artificial Intelligence 671*; Ohlbach, H. J., Ed.; Springer-Verlag: Berlin, 1992; pp 221−234.

(16) Consortium Menelas, Menelas: An Access System for Medical Records using Natural Language, *Computer Methods and Programs in Biomedicine*, Special issue on AIM; Zweigenbaum, P., Ed.; 1994; Vol. 45, pp 117−120.

(17) Rassinoux, A.-M.; Baud, R. H.; Scherrer, J.-R. A multilingual analyser of medical texts. In *Conceptual Structures: Current Practices. Second International Conference on Conceptual Structures, ICCS'94*; College Park, MD, U.S.A., August 1994; Proceedings; Springer-Verlag: Berlin, 1994; pp 84−96.

(18) van Bakel, B. Lexicale analyse in ELSA, internal report; Department of Language and Speech; Katholieke Universiteit Nijmegen: 1992.

(19) van Bakel, B. Semantic analysis of chemical texts. In *Linguistic engineering: tools and products, Proceedings of the second Twente workshop on language technology (TWLT-2)*; ter Stal, W., Nijholt, A., op den Akker, H. J., Eds.; dep. SETI/IS, fac. of informatics, Technical University Twente, Enschede, The Netherlands, 1992; pp 23−29.

(20) Chomsky, N. Lectures on Government and binding: the Pisa lectures; Foris: Dordrecht, The Netherlands, 1981.

(21) Coppen, P. A. GRAMTSY 4.0, GRAMmatical Transformational System; KUN, Department of Language and Speech: 1991.

(22) van Bakel, B. De compatibiliteit van Chomsky en Montague, in *Zin dat het heeft; een liber amicorum voor Jan van Bakel*; van Bakel, B., Coppen, P. A., Rolf, P., Eds.; Department of Language and Speech, Katholieke Universiteit Nijmegen: 1992; pp 73−84.

(23) van Bakel, B.; Oltmans, E. A modular approach to handling syntactic ambiguity, Proceedings of CLIN V, **1995**, submitted for publication.

(24) Meijer, H. Programmar: A Translator Generator, Ph.D. Thesis, University of Nijmegen, The Netherlands, 1986.

(25) van Bakel, B. Ph.D. Thesis, 1996, in press.

(26) Griswold, R. E.; Poage, J. F.; Polonsky, I. P. SNOWBOL4 programming language; Prentice Hall, Inc.: NJ, 1971.

(27) Sowa, J. F. Conceptual structures: information processing in mind and machine; Addison-Wesley: Reading, MA, 1984.

(28) Postma, G. J.; Kateman, G. A Systematic Representation of Analytical Chemical Actions. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 350−368.

(29) Thom, J. A.; Zobel, J. NU-Prolog version 1.3; Department of Computer Science, University of Melbourne, 1986.

(30) Bouaud, J.; Bachimont, B.; Charlet, J.; Zweigenbaum, P. Acquisition and Structuring of an Ontology within Conceptual Graphs. In *Proceedings of the ICCS'94 workshop on Knowledge Acquisition using Conceptual Graph Theory*; University of Maryland: MD, 1994.

(31) Willems, M. Pragmatic semantics by conceptual graphs. In *Conceptual Structures: Current Practices. Second International Conference on Conceptual Structures, ICCS'94*; College Park, MD, U.S.A., August 1994; Proceedings; Springer-Verlag: Berlin, 1994; pp 31−44.

(32) Abstract AAN5204F00026 9003, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.

(33) Abstract AAN5204E00034 9003, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.

(34) Abstract AAN5105E00051 8904, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.

(35) Abstract AAN5203E00052 9003, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.
(36) Abstract AAN5204E00038 9003, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.
(37) Abstract AAN5204E00056 9003, Analytical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.
(38) Freiser, H.; Nancollas, G. H. Compendium of analytical nomenclature. Definitive rules of 1987; Blackwell Scientific Publications: Palo Alto, 1987.
(39) The American Chemical Society, Chemical Abstracts, Chemical Abstract Service, Columbus, OH, U.S.A.
(40) ter Stal, W. G.; van der Vet, P. E. Two-level semantic analysis of compounds. In *CLIN IV, papers from the forth CLIN meeting*; Bouma, G., van Noord, G., Eds.; Department Alfa-informatics, University of Groningen: Groningen, The Netherlands, 1994; pp 163−178.
(41) Eggert, A. A.; Jacob, A. T.; Middlecamp, C. H. Converting chemical formulas to names: an expert strategy. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 227−233.
(42) Munday, C.; Lukose, D. Object-Oriented design of Conceptual Graph Processor. In *Proceedings of the Fourth International Workshop on Peirce: A Conceptual Graph Workbench*; Ellis, G., Levinson, R., Eds.; University of Maryland: MD, 1994; pp 55−70.
(43) Proceedings of the Fourth International Workshop on Peirce: A Conceptual Graph Workbench; Ellis, G., Levinson, R., Eds.; University of Maryland: MD, 1994.
(44) Lehmann, F. CCAT: the current status of the conceptual catalogue (ontology) group, with proposals. In *Proceedings of the Fourth International Workshop on Peirce: A Conceptual Graph Workbench*; Ellis, G., Levinson, R., Eds.; University of Maryland: MD, 1994; pp 18−28.
(45) Tepfenhart, W. M. Representing knowledge about substances. In *Conceptual Structures: theory and implementation. 7th annual workshop*; Las Cruses, 1992; Proceedings; Springer-Verlag: Berlin, 1993; pp 59−71.
(46) Lehnert, W.; Sundheim, B. A performance evaluation of text-analysis technologies. *AI Magazine* **1991**, *12*(3), 81−94.
(47) Fifth Message Understanding Conference (MUC-5); Proceedings of a conference held in Baltimore, MD, August 25−27, 1993; Morgan Kaufmann Publishers: San Francisco, 1993.
(48) Abstract AAN4910C00023 8709, Analitical Abstracts Online, (STN Host), The Royal Society of Chemistry, Letchworth, Herts, England.
(49) The Analyst, Instructions to authors. *Analyst* **1995**, *120*, 219−225.
(50) Analytical Chemistry, Instructions to authors. *Anal. Chem.* **1995**, *67*, 229−234.
(51) Analytica Chimica Acta, Instructions to authors. *Anal. Chim. Acta* **1994**, *289*, 381−384.
(52) Analytical Letters, Instructions for preparation of manuscripts for direct reproduction. *Anal. Lett.* **1995**, *28*(1).
(53) Journal of Chromatography, Instructions to authors. *J. Chromatogr. A* **1994**, *657*, 464−469.
(54) Talanta, Instructions for authors. *Talanta* **1995**, *42*(1), v,vi.
(55) Official Methods of Analysis of AOAC International, 16th ed.; Association of Official Analytical Chemists, Arlington, 1995.
(56) Postma, G. J.; Hack, F. M.; Janssen, P. A. A.; Buydens, L. M. C.; Kateman, G. A database on analytical chemical methods applying fuzzy logic in the search-strategy and flowcharts for the representation of the retrieved analytical procedures. *Chemometrics Intelligent Laboratory Systems* **1994**, *25*, 285−295.
(57) Zhou, T.; Isenhour, T. L.; Marshall, J. C. Object-Oriented Programming Applied to Laboratory Automation. 2. The Object-Oriented Chemical Information Manager for the Analytical Director. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 569−576.
(58) Zhou, T.; Isenhour, T. L.; Marshall, J. C. Object-Oriented Programming Applied to Laboratory Automation. 3. The Standard Robot Interface Protocol for the Analytical Director. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 558−569.
(59) Beugelsdijk, T. J.; Hollen, B. M.; Erkkila, T. H. The standard laboratory module: an integrated approach to standardization in the analytical laboratory. *Chemometrics Intelligent Laboratory Systems: Laboratory Inf. Management* **1993**, *21*, 207−214.
(60) Lysakowski, R. The global standards architecture for analytical data interchange and storage, ASTM Standardization News, 1992; Vol. 20, March, pp 44−51.
(61) Koncar, N.; Guthrie, G. A Natural Language Translation Neural Network. In *Proceedings of International Conference on New Methods in Language Processing (NeMLaP)*, Manchester, 14−16th Sept 1994; pp 71−77.

CI950206X