Extraction of Information from the Text of Chemical Patents. 1. Identification of Specific Chemical Names

Nick Kemp and Michael Lynch*

Department of Information Studies, University of Sheffield, Sheffield S10 2TN, United Kingdom

Received January 6, 1998

Much attention has been paid to translating isolated chemical names into forms such as connection tables, but less effort has been expended in identifying substance names in running text to make them available for processing. The requirement for automatic name identification becomes a more urgent priority today, not the least in light of the inherent importance of patents and the increasing complexity of newly synthesized substances and, with these, the need for error-free processing of information from patent and other documents. The elaboration of a methodology for isolating substance names in the text of English-language patents is described here, using, in part, the SGML (Standard Generalized Markup Language) of the patent text as an aid to this process. Evaluation of the procedures, which are still at an early stage of development, demonstrates that even simple methods can achieve very high degrees of success.

INTRODUCTION

The design of algorithms for translating chemical names into the corresponding molecular formulas was one of the earliest points of departure in computer handling of chemical information−this was the subject of Gene Garfield's doctoral thesis in the late 1950s.[1] Since then, a variety of processes dealing with chemical substance names, for instance, the translation of names into connection tables begun by Vander Stouw[2] in the mid-1960s, and further developed by Cooke-Fox et al.,[3−8] has reached wide generality. The translation of connection tables into names on the basis of the IUPAC rules by Wisniewski, a facility which is now incorporated in the AUTONOM package from the Beilstein Institute, is generally viewed as a *tour de force*.[9]

Despite the strong focus on substance names in chemical structure information systems during the past 30 years, most processes developed so far require the prior isolation of the name, although Hodge et al.[10,11] briefly outlined methods intended to identify chemical names in bibliographic titles and supplementary terms to assign CAS Registry Numbers. In one otherwise notable project concerned with information extraction from the Experimental Sections of the *Journal of Organic Chemistry*, Zamora and Blower[12,13]and Ai et al.[14] found that they seldom needed to isolate the chemical names of central interest in the publications. Rather, numeric references to diagrams showing the substances, presented as graphics in earlier sections of the papers, were used as identifiers for the substances, so that the names to be dealt with were predominantly those of the more common reagents, solvents, reactants, and substances used during workup. Again, such work as already exists on information extraction from printed chemical information sources has often focused on the identification and interpretation of graphical structure diagrams (e.g., Ibison et al.[15] and Simon et al.[16]). On the other hand, the capture of descriptions of generic structures from the text of abstracts of chemical patents has been described by Chowdhury and Lynch.[17,18] This work dealt with the automatic extraction of generic structure descriptions from the text of abstracts of patents published by Derwent Publications, and was aimed at translating the descriptions into GENSAL, the representation language for generic chemical structures devised during earlier work at Sheffield on the development of data structures and search algorithms for these complex structures.[19] Substantial success was achieved in extracting generic structure descriptions from the texts of *Documentation Abstracts*, the success being due in part to the severely constrained nature of the language of the abstracts. This constrained nature is due both to the need for brevity and to the use of well-developed conventions, so that the application of rigorous processing rules achieves a substantial degree of success. This application of rigorous rules is what is now referred to as "template mining"; that is, it uses routines that anticipate regular structures.

This success led us to examine English language patent texts to evaluate the extent to which analogous procedures could be devised to extend the work to the full texts of patent documents, particularly because of the inherent importance of patent information.[20] The applicability of Natural Language Processing (NLP) to information retrieval processes has been well reviewed by Smeaton.[21] Our attention was first directed toward chemical patent texts as a whole, with the intention of applying NLP techniques to extract general information such as applications and uses as well as that relating to chemical structures and their synthesis and properties. Although some progress was achieved in identifying recurrent patterns or templates that characterize passages of various kinds, their scope was limited. Moreover, it quickly became apparent that NLP still has to develop general methods for Information Extraction (IE), and also that it is inadequate in regard to features such as the resolution of anaphora (i.e., references back to entities mentioned earlier in a document or passage). Indeed, Cowie and Lehnert[22] have reported a shift toward the use of template mining in IE, suggesting that for this purpose, NLP achieves

S0095-2338(98)00324-2 CCC: $15.00
Published on Web 05/30/1998

its greatest successes in areas where template mining is applicable; that is, where there is a consistency in text structure such that templates can be developed that enable consistent information identification and extraction. Given these drawbacks, it seemed unrealistic to attempt a major attack on this front. Instead, we chose to take a fall-back position and sought to identify chemical names, both those of specific substances and generic chemical names, in the full texts of patents. Here, the objective is to add value to the text of patents, from the point of view of providing improved retrieval and access tags, regardless of whether the information isolated is to be removed to undergo processes elsewhere, or by tagging the information in situ by application of Standard Generalized Markup Language (SGML), thus enabling rapid and accurate focusing on these items.[23] Similar work is known elsewhere in areas such as the identification and markup of proper names, for instance, by Wakao *et al.*[24] In analogous work, we have recently developed methods for identifying citations−to other patents, to the periodical literature, and to books−in the full text of English-language patents.[25]

The approach taken in our work on chemical name identification has been to reflect the methods of sublanguage analysis; that is, the identification of the necessary procedures by examining some corpus of data, rather than starting from the rules of substance name formation. Sublanguage analysis involves the examination of some selected corpus of language to determine the circumstances in which the language is applied. Our purpose is to identify a methodology that can be extended successively to wider areas. Clearly, in the longer term, and for the purpose of comprehensive coverage, the rules of nomenclature systems will also need to be fully reflected.

The aim of the work described here is thus the identification of names of nongeneric chemical substances within running text. The texts used were the sections describing the inventions in European patent applications in the field of organic chemistry; these are marked-up using SGML according to a WIPO standard,[26] but not in such detail that chemical names in running text are so distinguished. For the purposes of this study, alternative representations such as molecular formulas, line formulas, etc., including expressions such as "NaOH", "H_2SO_4", and "CH_3COOH", were excluded from attention. One sample of 100 patents was used as the training set to develop the methods, and a second set, of approximately the same size and composition, was used to test the methods.

The approach taken aimed to provide the greatest generality of solution possible and therefore the task was tackled first at the level of text microstructure. The strings of alphabetic characters that commonly appear within chemical names are conspicuously different in microstructure from general English language words. Consequently, the approach involved the identification of discriminant alphabetic substrings that can be used to differentiate between strings that may be part of a chemical name on one hand, and all other words. The next section therefore describes the production of three dictionaries to this end. A set of routines is then described that incorporate these dictionaries to isolate the names of chemical substances from text in two stages. The first stage involves the tokenization of the text, during which the dictionaries are applied to identify fragments of text that belong to chemical names. During the second stage, the tokens are recombined to reproduce the input text with tags added that delimit the chemical names that have been identified.

Table 1. Number of Chemical Name Fragment Types at Selected Frequency Threshold Levels

frequency threshold	number of types
>= 1	2330
> 2	1409
> 5	633
> 10	346

The description sub-documents of the 100 chemical patents of the development set were edited manually to remove passages that explicitly described generic chemical compounds; an extension of this work to deal with the isolation of these generic descriptions is under preparation. The text that remains contains information on the background to the invention, the use(s) to which the invention may be applied, and descriptions of experimental methods, all of which may contain names of specific chemical compounds. Some generic compound names or name parts still remain within the edited texts in general discussion, in descriptions of experimental methods, etc., and their presence required the development of some routines to identify and eliminate them.

CHEMICAL NAME FRAGMENT DICTIONARIES

Strings that are unique to the names of specific chemical compounds were identified by the following method. A database of 2703 chemical names was produced by manual extraction of chemical substance names from the description sections of the 100 patents of the development set. These chemical names comprise punctuation characters, numerical digits, entity references, and SGML tags in addition to alphabetic character sequences. A short program was written to extract and capitalize alphabetic character sequences of three or more characters from these database, and resulted in ~10 000 tokens. The list of strings produced was sorted alphabetically, and the frequencies of unique string occurrences were determined. This procedure yielded ~1000 string types. The STARLIST database was used as a further source of chemical names. Specific chemical names, 6500 in all, were extracted from it, and subjected to the same treatment as those names from the patents. Thus, a set of name fragments resulted from STARLIST that contained >23 000 tokens, of which 2330 were unique types.

The name fragments produced from the STARLIST database were thresholded at frequencies of two, five, and ten, as shown in Table 1. To assess the potential of these strings for identifying chemical name fragments, those with a frequency of five or greater were used as probes against the patent name fragments, by determining which of the STARLIST fragments were substrings of the patent name fragments. Those fragments that were assigned most frequently were the shorter strings of high frequency in the STARLIST database. Evidently, the possible variety of chemical fragments within chemical names is such that the rates of substring repetition above small string lengths is quite insignificant.

Accordingly, a second approach, with two variations, was developed, which proved to be much more successful. The

Table 2. Extract from the KLIC Index

PYRIDINE	METHANOL
DI	METHOXY
TRI	METHOXY
DI	METHYL
FLUORO	METHYL
	METHYL
PIVALOYLOXY	METHYL
TETRA	METHYL
TRIFLUORO	METHYL
TRI	METHYL
	METHYLAMIN
DI	METHYLAMINO
	METHYLAMINO
DI	METHYLAMINOETHYL
	METHYLBENZAL
	METHYL BENZYLYLCARB
DI	METHYLCARBAMAT
	METHYLCARBAMAT
	METHYLCARBAMIC
	METHYLEN
	METHYLENEBISPHENOL
	METHYLPHENYL
DIFLUORO	METHYLTHIO
	METHYLTHIO
	METHYLTRI
DI	METHYLTRIAZENYL
	METHYLTRIAZENYL
	METHYLTRIAZEN
	METHYLUREA
DI	METHYLUREYL
	METHYLUREYL

Table 3. Extract from the List of Discriminant Substrings

ACET	ANESULF
ACID	ANESULPH
ADAMANT	ANIL
ADENIN	ANISOL
ALANIN	ANOIC
ALCOHOL	ANOL
ALDEHYD	ANOS
ALLO	ANTHRAC
ALLYL	ANTHRAN
ALPHA	ANTIMON
ALUMIN	ASCORB
AMAT	ASPARTIC
AMIC	AZA
AMID	AZEN
AMIN	AZEPIN
AMMONI	AZID
ANDROST	AZIN
ANEDI	AZOL

Table 4. Extract from the List of Stopstrings

EPISODE	HYDROCARB
EXAMIN	HYDROLYS
FIED	HYDROLYZ
GALLON	HYPER
HALO	HYPO
HORMON	PHILIC
HYDRAULIC	PHOBIC

name fragments from the training set of patents were analyzed to identify shorter substrings that were characteristic of the fragments in the sense that they occurred as substrings within the names. These discriminant substrings were identified first by scanning the alphabetical list of name fragments and manually selecting those that appeared to represent a range of chemical name morphemes. Second, in a semiautomatic approach, a Key-Letter-In-Context (KLIC) index was produced; that is, an alphabetical listing of the unique name fragments in which each letter of each of the fragments becomes an index point in turn. Discriminant substrings were then identified by visual selection. For example, the variety of letters that follow the string "methyl" is much greater than that which follow the string "methy". This is illustrated in Table 2, which shows the strings from the KLIC index in the region in which the fragment "methy" occurs, where each occurrence of the fragment "methy" is a substring of "methyl". Though this method is nonalgorithmic in nature, it appears likely that this could be automated, because the desired cutoff points should show much greater varieties of subsequent characters. Thus, the determination of a suitable threshold level for the number of different subsequent characters following a substring would enable the automatic production of discriminant fragments. An extract from the list of discriminant substrings is given in Table 3.

In addition to the list of strings that are used to identify positively chemical name strings, two further lists were compiled. The first was a stopword list containing a set of words that are not chemical names. The stopword list was prepared by forming an inverse frequency list of word types, of three or more characters in length, appearing in the training set. Those strings appearing as chemical name fragments and those that met criteria, described later, for being treated as plural nouns were removed from consideration. The list of stopwords contains 750 high-frequency words arranged in alphabetic order, and includes

- function words (such as "and", "than", and "wherein")
- strings that are names of frequent generic groups (such as "alkoxycarbonyl", "halide", and "hydrocarbon")
- common words that are erroneously identified as chemical names (such as "vit*amin*", "*iso*mer", and "*tetra*hedron", the italicized parts being the substrings appearing in the list of chemical name fragments)
- other words appearing frequently in the descriptions of the patent applications that are not chemical strings, such as "formula", "solution", and "substituent"

During creation of the stopword list it was noted that many nonchemical name tokens contained prefixes or suffixes that precluded the name from being a chemical name. These affixes generally indicate processes or states in which some chemical treatment is designated. The addition of a dictionary of stopstrings (Table 4) enables tokens that would otherwise be falsely identified as chemical names to be successfully excluded. Therefore, 70 such alphabetic substrings were identified by manual scrutiny and were placed in a dictionary of stopstrings. Tokens containing these stopstrings were removed from the stopword dictionary. The dictionary includes:

- extended substrings, such as "hydrolys" and "hydrolyze", which are used to differentiate between chemical name strings (such as "hydrogen", "hydroxide", and "hydroquinone") and words that contain the same stem but are not name strings (such as "hydrolysis", "hydrolyse", and "hydrolyze")
- substrings that are characteristic of generic chemical names, such as "halo" and "alkyl"
- other morphemes that preclude the string from being a chemical name, such as the prefixes "hypo" and "hyper", and the suffixes "phobic" and "philic"

These dictionaries were developed by means of an iterative cycle of testing and result evaluation to improve differentia-

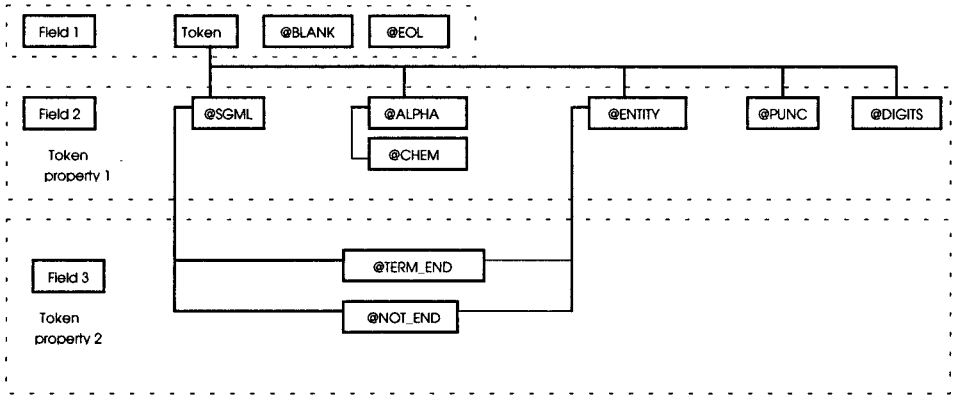

Figure 1. Data-structure of token records.

tion between alphabetic strings belonging to chemical names and other words. All three dictionaries were then used by the routines described in the next section.

IDENTIFICATION OF CHEMICAL NAMES IN RUNNING TEXT

Chemical names were identified and delimited in running text by a two-stage process. The first stage is tokenization of the text and includes the differentiation of chemical and nonchemical name fragments using the dictionaries just described. The second stage applies a series of heuristics in which the tokenized text is recombined and chemical names are delimited for subsequent use by the addition of SGML tags to the text.

The complex nature of the token types within the patent texts, including SGML markup and chemical names, requires a tokenizing process in which the input text is processed simply as a sequence of characters. The tokenizer differentiates between SGML tags, entity references (references to non-ASCII characters, for example, "&beta" represents "β" and "&sec" represents double prime), sequences of numeric digits, sequences of alphabetic characters, blank and end-of-line characters, and sequences of punctuation from the punctuation set that may appear in chemical names. The set of punctuation marks found in chemical names comprises comma, period, semi-colon, colon, left and right parentheses, braces and brackets, hyphen, asterisk and apostrophe. Each token is written to a data structure (Figure 1) that contains the token and one or two property fields indicating the type of token, for example, alphabetic character sequences have @ALPHA added to their token record, sequences of digits have @DIGITS added, and the end-of-line marker is indicated by the token @EOL. The SGML tags and entity references are matched against a dictionary containing a list of those SGML tags and entity references that cannot appear within a chemical name. The dictionary includes some tags that contain variable attributes (for example the document page tag, "<DP=...>"), and for these a partial match is made against the start of the tag that identifies the element type. Tags successfully matched against this dictionary are labeled @TERMEND in the second property field of the token record, as they may indicate the end of a chemical name, otherwise @NOT_END is written to this field.

The procedure to determine which alphabetic strings are chemical name string fragments was implemented as part of the tokenization phase. Alphabetic strings of three or more characters, temporarily capitalized so that the pattern matching that follows is performed on normalized tokens, are subjected to the following tests. The first test applies a simple suffix matcher to remove tokens with suffixes that are ostensibly common plurals and past or present participles from undergoing further tests. All strings ending in "-s", other than those ending in "-us", are removed from further consideration, as they are either plural nouns or are not nouns, and therefore not part of a specific chemical name. Tokens ending in "-us" are often found to be chemical in nature; examples from the training set include "nitrous" and "amaranthus". Few exceptions to this rule have been found, one being "aminomethylenebis". Strings ending in "-ed", such as "brominated" and "hydrogenated", are taken to be past participles, and those ending in "-ing", such as "chlorinating", as present participles. No exceptions to these rules were found. Strings that are matched successfully against a suffix are eliminated without further tests. The strings are then matched against the stopword dictionary and then with the dictionary of stopstrings. If the input string matches either, the procedure terminates, otherwise the input string is matched against the dictionary of discriminant name fragments. Strings identified as belonging to a chemical name have the property @CHEM substituted for @ALPHA in their token record.

TEXT RECOMBINATION AND CHEMICAL NAME DELIMITATION

The token records are stored as an array across which a pointer moves. A set of heuristics is applied in two phases to identify chemical names and to distinguish between designations of single chemical compounds and those that are generic in nature. The first phase makes an initial attempt to delimit names, whereas the second phase verifies the delimitation of the first phase. SGML tags are then added to the output text to delimit the names of single chemical compounds. This two-phase approach was dictated by storage limitations imposed by the compiler used for this work.

As the token records are scanned, a token with the property @CHEM triggers the search for the delimiters of a chemical name. The system first scans backward through the array to find the start of the name, because that may not have been identified as a chemical name element itself (e.g., a numerical locant), and then forward to find the end of the name. The backtracking is halted, indicating that the start of the name

Table 5. Example of Tagging with the Sequence "and *n*-Propyl Bromide Was"

and @ALPHA	@BLANK	n @ALPHA	- @PUNC	Propyl @CHEM	@BLANK	bromide @CHEM	@BLANK	was @ALPHA

Table 6. Intermediate Stage in Tagging an Example Text

and @ALPHA	@BLANK	n @START	- @PUNC	Propyl @CHEM	@BLANK @C_END	bromide @CHEM	@BLANK	was @ALPHA

Table 7. Intermediate Stage in Tagging an Example Text

and @ALPHA	@BLANK	n @START	- @PUNC	Propyl @CHEM	@BLANK @C_END	bromide @ START	@BLANK @C_END	was @ALPHA

Table 8. Final Stage in Tagging an Example Text

and @ALPHA	@BLANK	n @START	- @PUNC	Propyl @CHEM	@BLANK	bromide	@BLANK @C_END	was @ALPHA

has been found, when a blank character, an SGML tag with the property @TERM_END, or an end-of-line marker is met. The token found to be the start of the chemical name has its property provisionally updated to @START. The procedure then scans forward across the array to find the end of the chemical name. The forward scanning is halted by one of 10 terminating conditions (Appendix 1). The token that meets the terminating condition is given the property of @C_END.

To exemplify the method, if the text "and n-Propyl bromide was" were being processed, the tokenizer would produce the tokens shown in Table 5. The system is triggered by the @CHEM associated with "Propyl", but this is not the start of the name, thus the system must scan back to find a delimiting condition − in this case the @BLANK between the "and" and the "n". Similarly it must scan forward to find the end of the name. The first delimiting condition found is the @BLANK immediately after the "Propyl". Thus the tokens are updated as shown in Table 6. This process is repeated when the system is triggered by the @CHEM associated with "bromide" (Table 7).

At this point, the initial name delimitation is complete. However, the delimiters in use at this point do not allow chemical names to contain a blank character nor to run across an end-of-line marker, as in the case of the worked example. The verification phase described in the following section corrects this flaw and applies routines that prevent generic substances from being tagged.

CHEMICAL NAME VERIFICATION

The verification phase first checks that chemical names have not been split incorrectly. Delimited chemical names separated only by an @BLANK token, an @EOL, or by the start of a new page are treated as if they are two parts of the same name. (The sequence of tokens indicating a new page, other than the first page of the document, is; an @EOL token followed by the SGML tag for a new page, which is <DP=IJK>, where IJK represents the new page number. The SGML tag has the properties @SGML and @NOT_END.) The end-delimiter property of the first name and the start-delimiter property of the second name are therefore updated. Thus, in the worked example, the token records are updated as in Table 8.

Hyphenated words such as "hydrogen-rich" and "labeled-iodine" would, at this stage, be incorrectly delimited as chemical names, in addition to correctly delimited names such as "*n*-butane". A routine is therefore implemented that tests all such delimited chemical names. Only names in which the first alphabetic string is either a single character, or belongs to the following set of strings remain delimited as chemical names: "iso", "sec", "tetra", "tri", "bis", "cis", "ortho", "meta" and "para".

As mentioned previously, the system is not intended to tag generic names. To this end, those strings that are ostensibly plural are not given the @CHEM property, and the dictionary of negative substrings contains some strings such as "alkyl" to discriminate specific from generic names. Four further routines are required to ensure that generic names are not tagged.

The first routine is concerned with the removal of chemical names that, when appearing in isolation from other chemical names, are essentially generic. For example, "acetic acid" is the name of a specific chemical substance, whereas "acid" is not. These terms usually appear in the text as anaphora. When the system finds a chemical name that is a single alphabetic string, the name is compared with a dictionary containing a list of such terms, and those that are matched successfully are untagged.

Zamora and Blower[11] suggested that determiners are not used to refer to specific chemical names within experimental descriptions in the *Journal of Organic Chemistry*. Scrutiny of the training set confirmed that this suggestion also applied to the more general text of patent applications, and also to the experimental descriptions within the descriptions. The determiners identified are "a", "an", and "the". Thus, a routine was added that removes the chemical name delimiters from the token records if such a determiner immediately precedes a chemical string.

Ostensibly, plural forms of chemical names are considered as generic and therefore should not be tagged in the system output. In instances of plural compound nouns, such as "sodium fluoroacetates", the system is likely to have assigned the delimiters incorrectly at this point in processing (e.g., "sodium" is tagged and "fluoroacetates" is not). Thus, a routine in this procedure identifies delimited chemical names immediately precede common plurals (i.e., those ending with a consonant followed by "-s" or ending "-es"). The record properties that delimit the preceding chemical name are then updated so that the string is no longer marked as a specific chemical name.

The penultimate verification routine attempts to match the token that follows the name string terminating record with one of a set of characteristic tokens, such as "group" or "compounds", which explicitly denote generic definitions. If a match is successful, the records that are the start and end points of the name string have @START and @END removed from their property fields. Furthermore, it is often the case that these characteristic tokens are preceded by several generic chemical names rather than by a single name. A backtracking routine is therefore implemented to remove the delimiters from the property fields of the appropriate records.

The final routine within the verification phase checks that chemical names ending with a symbol from the punctuation set are correctly delimited. The routine is required as a consequence of the tokenization phase that reads strings of punctuation into a token record rather than a single punctuation mark per record. The system then writes the tokens of the records to an output file in such a way that the output file resembles the initial description input file with SGML tags added, the tag ⟨*CHEM*⟩ indicating the start of a chemical name and </CHEM> indicating the end of a chemical name. Thus, the worked example would be output as "and ⟨*CHEM*⟩ *n*-Propyl bromide</CHEM> was".

RESULTS

The system was run against a test set of 70 patent descriptions taken from documents from the IPC class CO 7D, which were part of a randomly selected set of this classification provided by the European Patent Office. As noted earlier, the texts were first edited manually to remove the descriptions of generic compounds, denoted by the inclusion of radical groups such as "where R_1 is.."., leaving ~4.5 MB of ASCII text.

The text contained 14 855 specific chemical names, identified by manual scrutiny. Of these, 14 467 (97.4%) were correctly identified by the routines, 178 (1.2%) were missed, and 210 (1.4%) were partially delimited. In addition, 618 tokens (4.2%) that were not chemical names were falsely identified. In judging whether a chemical name was correctly delimited or not, certain qualitative decisions had to be made. Chemical names can be preceded by adjectives that describe the chemical nature of the compound that follows; for example, "*red* iron oxide" as opposed to "*black*" or "*yellow* iron oxide", or the terms "aqueous" or "anhydrous" preceding a chemical compound name. In such cases, the system delimits the compound name only, and the adjective is unmarked; the compound name is judged to have been correctly assigned.

Nesting (*i.e.*, the nonrepetition of some components of names with components in common) is problematic; occasionally, a chemical name that was delimited correctly is preceded by an alternate prefix (e.g., a substituent, a locant, or a multiplier, as in "di- or ⟨*CHEM*⟩ tri-chlorobenzoate</CHEM>", resulting in only one of the compounds being tagged as shown). In evaluating the system, such instances are counted as one correctly assigned chemical name ("trichlorobenzoate") and one missed name ("di-chlorobenzoate"); clearly, the nesting exemplified here can occur more generally and needs to be taken into account more fully.

Words that were incorrectly identified as chemical names include several distinct groups of words:

- proper names containing *chemical* substrings such as "W*ester*n", "W*ether*all", and "L*aza*rus", and trade names such as "*Iso*nox" and "Not*val*l" (the chemical substring is italicized)
- other words containing chemical substrings including "el*imin*ate", "har*vester*", and "ens*ure*"
- ions and radical groups that are not included in the list of chemical names that are not delimited if they appear in isolation, such as "phosphate" and "sulfamate"
- generic chemical names that contain two or more words identified as chemical strings separated by a blank character, such as "carboxylic acid ester", "acid chloride", and "acid anhydride"

Names that are only partially assigned usually contain two or more strings of nonblank characters, one of which does not contain a substring identified as chemical. Two hundred of the 210 partially assigned chemical names were in this category, examples include

- "⟨*CHEM*⟩ isopropyl</CHEM> myristate"
- "⟨*CHEM*⟩ sodium</CHEM> dodecyl ⟨*CHEM*⟩ sulfate</CHEM>"
- "⟨*CHEM*⟩ ferric</CHEM> citrate"

The system described here also lacks routines to process inorganic chemical names that include the oxidation states of the metal ion. As a result, the system fails in five of the six cases where such names occur. Incorrect delimitation occurs in those instances where the oxidation state is separated from one or both of the alphabetic strings by blank character(s).

There are several reasons why a chemical name may be missed by the system, the most common being the incompleteness of the dictionary of chemical name substrings, because that used here was derived solely from the sources described. Examples of failures of this kind include "squalane", "sorbitan sesquilate", "picoline", "mannitol", and "maleic" in "maleic acid". Further, the rule that if a chemical string follows a determiner it is not marked because it is generic is the cause of further errors, these being caused exclusively by the word "the".

The routine that removes the chemical delimiters from strings that are followed by a plural noun is responsible for the system missing some chemical names, due to its method of implementation rather than the nature of the rule itself. The system checks the token following the delimited chemical name. If the last characters of the token are a consonant followed by an "s" or "es", then the token is considered a plural noun. This implementation is crude, but suffices in the vast majority of cases.

GENERIC CHEMICAL NAMES

The rules developed so that generic names were not delimited proved to be reasonably effective; however, as noted earlier, most of the text containing generic structure descriptions had previously been manually edited from the texts. An extract taken from European Patent Application number 92201668 after processing illustrates a variety of points:

"...Specific examples of such substituents include, for example, "halogen, especially *fluorine*, *chlorine* or *bromine* atoms, nitro*, ⟨*CHEM*⟩ cyano</CHEM>, hydroxyl*, alkyl*, haloalkyl* (especially CF&sub3;), alkoxy*, haloalkoxy*,

<H1><U>Example 2</U></H1>

<H1><U><CHEM>1-Chloro-4-n-propoxythioxanthone</CHEM></U></H1>

<P><CHEM>1-Chloro-4-hydroxythioxanthone</CHEM> (7.8g; 0.03 mol) prepared as above and <CHEM>potassium carbonate</CHEM> (5.0g; 0.036 mol) was stirred and refluxed for 10 minutes in acetone (50 ml). <CHEM>n-Propyl bromide</CHEM> (5.5g; 0.045 mol) was added and the resulting mixture heated under reflux for 16 hours. The mixture was then cooled and quenched on to water (250 ml). The solid was filtered and washed with water the resulting damp solid recrystallised from <CHEM>ethanol</CHEM> (50 ml) to afford the title compound (6.9g; 75.5%) of mp 102-3ø.

Figure 2. Sample output text.

amino*, alkylamino*, dialkylamino*, *formyl*, alkoxycarbonyl*, *phenoxycarbonyl*, *benzyloxycarbonyl*, *carboxyl*, *alkanoyl*, alkylthio*, alkylsulfinyl*, alkylsulfonyl*, *carbamoyl* and alkylamido* groups."

The tokens that appear underlined belong to the list of generic identifiers and trigger the backtracking routine that removes the delimiters from chemical names described earlier. The tokens which appear in italics have their tags removed as a result of this routine. The tokens labeled with * are not tagged because they appear in the dictionary of terms not to be tagged when appearing in isolation ("hydroxyl"), or are in the stopword list ("alkoxy"), or contain strings in the list of negative sub-strings ("alkyl"). The backtracker stops at the token "especially", resulting in the tagging of the token "cyano" which is identified as a chemical name fragment but does not appear in the isolation dictionary. The system proved particularly successful in identifying chemical names within descriptions of experimental preparations. An example of the output of such a description is given in Figure 2.

DISCUSSION AND CONCLUSIONS

Despite the relatively simplicity of the routines described here, they are sufficient to isolate the vast majority of specific chemical names in the texts studied and to differentiate them from generic names in the general discussion parts of the description sections of patents. The greatest difficulty is likely to ensue when names that are derivatives of substances with which trivial chemical names have already been associated. Examples of the latter are "Derivatives of saccharin". However, many of the European patents that include large numbers of substances tend to use names that are devised *de novo*, rather than as derivatives, and this may have contributed to the perhaps surprisingly high degree of correct isolation of chemical names.

These routines could readily be extended, with similar degrees of success, to deal with chemical names in other natural languages, at least those of European origin, with account being taken of the differences in name formation in each language. Before any implementation, the addition of a large dictionary of chemical names and name fragments in addition to the substrings identified here is necessary to ensure higher success rates, and use of a comprehensive chemical name parsing routine would ensure a still higher rate of accuracy.

ACKNOWLEDGMENT

N.K. thanks the UK Department of Education for an Information Science Research Studentship. We thank the European Patent Office for kindly providing many patent documents in machine-readable form, and STARLIGHT Company for use of the STARLIST file as a source of chemical names.

APPENDIX 1

1. The current token is an SGML tag with the property @TERM_END.
2. The current token is a string of punctuation and the following token is a blank character.
3. The current token is a period that is followed by an end-of-line marker, and the first token of the next line is not a string of digits.
4. The current token is a comma that is followed by an end-of-line marker, and the first token of the next line is not a string of digits.
5. The current token is a semi-colon that is followed by an end-of-line marker, and the first token of the next line is not a string of digits.
6. The current token is a colon that is followed by an end-of-line marker, and the first token of the next line is not a string of digits.
7. The current token is a string of punctuation that is followed by an end-of-line marker, and the first token of the subsequent line is an SGML tag with the property @TERM_END.
8. The current token is a string of punctuation that is followed by an end-of-line marker, and the first token of the subsequent line is an alphabetic string with a an uppercase initial letter.
9. The current token is a blank character.
10. The current token is an end-of-line marker.

REFERENCES AND NOTES

(1) Garfield, E. An algorithm for translating chemical names to molecular formulas. *J. Chem. Doc.* **1962**, *2*, 177−179.
(2) Vander Stouw, G. G.; Naznitsky, I.; Rush, J. E. Procedures for Converting Systematic Names of Organic Compounds into Atom-bond Connection Tables. *J. Chem. Doc.* **1967**, *7*, 165−169.
(3) Cooke-Fox, D. I.; Kirby, G. H.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature, Part 1. Introduction and Background to a Grammar-based Approach. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 101−105.
(4) Cooke-Fox, D. I.; Kirby, G. H.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature, Part 2. Development of a Formal Grammar. *J. Chem. Inf. Comput. Sci*. **1989**, *29*, 106−112.
(5) Cooke-Fox, D. I.; Kirby, G. H.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature Part 3. Syntax Analysis and Semantic Processing. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 112−118.
(6) Cooke-Fox, D. I.; Kirby, G. H.; Lord, M. R.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature, Part 4. Concise Connection Tables to Structure Diagrams. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 122−127.
(7) Cooke-Fox, D. I.; Kirby, G. H.; Lord, M. R.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature, Part 5. Steroid Nomenclature. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 128−132.

(8) Cooke-Fox, D. I.; Kirby, G. H.; Lord, M. R.; Rayner, J. Computer Translation of Systematic Organic Chemical Nomenclature, Part 6. (Semi)automatic Name Correction. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 153−160.
(9) Wisniewski, J. L. AUTONOM−a Chemist's Dream: System for (Micro)computer Generation of IUPAC-compatible Names from Structural Input. In *Chemical Structures 2*; Warr, W. A., Ed.; Springer-Verlag: Heidelberg, 1993; pp 55−64.
(10) Hodge, G. M.; Nelson, T. W.; Vleduts-Stokolov, N. Automatic Recognition Of Chemical Names In Natural-Language Texts. *Abstracts of Papers of the American Chemical Society*. **1989**, *197*, Apr., P17−CINF.
(11) Hodge, G. M.; Enhanced Chemical Name Identification Algorithm. *Abstracts of Papers of the American Chemical Society*. **1991**, *202*, Aug., P41−CINF.
(12) Zamora, E.; Blower, P. E. Extraction of Chemical Reaction Information from Primary Journal Text using Computational Linguistics Techniques. 1. Lexical and Syntactic Phases. *J. Chem. Inf. Comput. Sci*. **1984**, *24*, 176−181.
(13) Zamora, E,; Blower, P. E. Extraction of Chemical Reaction Information from Primary Journal Text using Computational Linguistics Techniques. 2. Semantic Phase. *J. Chem. Inf. Comput. Sci.* **1984**, *24*, 181−188.
(14) Ai, C. S.; Blower, P. E.; Ledwith, R. H. Extraction of Chemical Reaction Information from Primary Journal Text. *J. Chem. Inf. Comput. Sci*. **1990**, *30*, 163−169.
(15) Ibison, P.; Jacquot, M.; Kam, F.; Neville, A. G.; Simpson, R. W.; Tonnelier, C.; Venczel, T.; Johnson, A. P. Chemical Literature Data Extraction−The CLiDE Project. *J. Chem. Inf. Comput. Sci*. **1993**, *33*, 338−344.
(16) Simon, A.; Johnson, A. P. Recent advances in the CLiDE Project: Logical layout analysis of chemical documents. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 109−116.
(17) Chowdhury, G. G.; Lynch, M. F. Automatic Interpretation of the Texts of Chemical Patent Abstracts. 1. Lexical Analysis and Categorization. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 463−467.
(18) Chowdhury, 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.
(19) Dethlefsen, W.; Lynch, M. F.; Gillet, V. J.; Downs, G. M.; Holliday, J. D. Computer Storage and Retrieval of Generic Chemical Structures in Patents, Part 11, Theoretical Aspects of the Use of Structure Languages in a Retrieval System, *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 260−270.
(20) Warr, W. A.; Suhr, C. *Chemical Information Management*; Verlag Chemie: Cambridge, 1992.
(21) Smeaton, A. F. Progress in the Application of Natural Language Processing to Information Retrieval Tasks. *Computer J.* **1992**, *35(3)*, 268−278.
(22) Cowie, J.; Lehnert, W. Information Extraction. *Commun. Assoc. Computing Machinery* **1996**, *39(1)*, 80−91.
(23) Goldfarb, C. *The SGML Handbook*; Clarendon: Oxford, 1990.
(24) Wakao, T.; Gaizauskas, R.; Wilks, Y. Evaluation of an algorithm for the recognition and classification of proper names. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING 96)*, Copenhagen, 1996; pp 418−423.
(25) Lawson, M.; Kemp, N.; Lynch, M. F.; Chowdhury, G. G. Automatic Extraction of Citations from the text of English-language patents−an Example of Template Mining. *J. Information Sci.* **1996**, *22*, 423−436.
(26) World Intellectual Property Organization. Permanent Committee On Industrial Property Information. PCIPPI/P 949/91 Revision 1, Annex 4. Geneva, 1993.

CI980324V