(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
4 January 2001 (04.01.2001)

PCT

(10) International Publication Number
WO 01/01289 A1

(51) International Patent Classification⁷: G06F 17/27

(21) International Application Number: PCT/US00/17444

(22) International Filing Date: 23 June 2000 (23.06.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/345,547 30 June 1999 (30.06.1999) US

(71) Applicant: INVENTION MACHINE CORPORATION, INC. [US/US]; 133 Portland Street, Boston, MA 02114 (US).

(72) Inventors: TSOURIKOV, Valery; 770 Boylston Street, Boston, MA 02199 (US). BATCHILO, Leonid; 40 Agassiz Avenue, Belmont, MA 02478 (US). SOVPEL, Igor; Apt. 193, 3/1 Voronyanskogo Street, 220039 Minsk (BY).

(74) Agent: DREYFUS, Edward; 608 Sherwood Parkway, Mountainside, NJ 07092 (US).

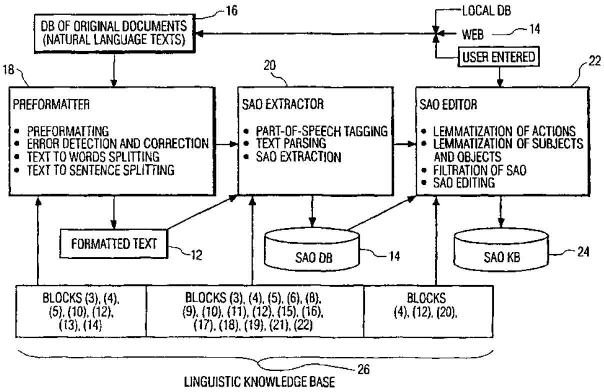
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*
— *With amended claims.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: SEMANTIC PROCESSOR AND METHOD WITH KNOWLEDGE ANALYSIS OF AND EXTRACTION FROM NATURAL LANGUAGE DOCUMENTS


16
DB OF ORIGINAL DOCUMENTS (NATURAL LANGUAGE TEXTS)
LOCAL DB
WEB
14
USER ENTERED
18
20
22
PREFORMATTER
• PREFORMATTING
• ERROR DETECTION AND CORRECTION
• TEXT TO WORDS SPLITTING
• TEXT TO SENTENCE SPLITTING
SAO EXTRACTOR
• PART-OF-SPEECH TAGGING
• TEXT PARSING
• SAO EXTRACTION
SAO EDITOR
• LEMMATIZATION OF ACTIONS
• LEMMATIZATION OF SUBJECTS AND OBJECTS
• FILTRATION OF SAO
• SAO EDITING
FORMATTED TEXT
12
SAO DB
14
SAO KB
24
BLOCKS (3), (4), (5), (10), (12), (13), (14)
BLOCKS (3), (4), (5), (6), (8), (9), (10), (11), (12), (15), (16), (17), (18), (19), (21), (22)
BLOCKS (4), (12), (20),
26
LINGUISTIC KNOWLEDGE BASE

(57) Abstract: A method of semantically processing natural language representations in a general purpose computer including entering and storing a user criterion, retrieving from remote and local databases (12, 14) and storing representations of the texts of a plurality of natural language documents that have some relationship with the stored user criterion (16), formatting said representations and storing the formatted representation (18), identifying and extracting from the formatted representation subject-action-object (SAO) extractions and storing the SAO extractions (20), processing the SAO extractions into normalized SAO structures and storing the SAO structures (22), designating the AO portions as substantially the names of Folders of at least some of the SAO structures, and storing in association with each the Folder name the identity of one or more subject portions (S_1, S_2, ...S_n) that are associated with the respective AO portion of stored SAO structures. The method further includes storing in association with each respective S_1, S_2, ... S_n the full sentence in which the respective SAO appears and highlighting each S-A-O portion that appears in each said full sentence. The list of subjects (S_1, S_2 ...S_n) stored in association with a respective AO portion is displayed in response to the user selecting the displayed AO portion or Folder name.

TITLE: SEMANTIC PROCESSOR AND METHOD WITH KNOWLEDGE ANALYSIS OF AND EXTRACTION FROM NATURAL LANGUAGE DOCUMENTS

REFERENCE TO PRIOR APPLICATION:

This is a continuation-in-part of US Patent Application SN 09/321,804, filed May 27, 1999, which matured from US Provisional Patent Application SN 60/099641, filed September 9, 1998, both applications of which are incorporated herein by reference.

BACKGROUND:

The present invention relates to natural language processing systems, and more specifically to a method and system for converting natural language texts into Subject-Action-Object Knowledge Database (SAO KB). This database can form the heart of various new applications or methods of natural language processing and analysis.

Different implementations of parsers are included in known natural language processors, such as Ergo Linguistic Technologies parser (U.S. Pat. No. 5878385), which has the following features: Part-of-Speech (POS) identification; Parts of Sentences identification; Passive to Active and Active to Passive mode conversion; Statement to Question conversion; Sentence Type identification; Tense conversion. A functional

Dependency Grammar Parser is also known (Pasi Tapanainen, Timo Jarvinen, A Non-Projective Dependency Parser: Proceedings of Fifth Conference on Applied Natural Language Processing, Washington, D.C., 1997), which builds the dependency tree of a sentence (i.e. which word depends on which word in the sentence) and extracts parts of the sentence. Syntactical parsing is also used in U.S. Patent Nos. 5,060,155 (J.M. van Juijlen) and 5,424,947 (K. Nagao, et al.). Another useful technique includes Parts-Of-Speech Tagging (POST), examples of which are disclosed in U.S. Patent No. 5,146,405 to K.W. Church, and U.S. patent No. 4,887,212 to A. Zamora, et al.

Prior art systems such as these and others analyze the text mostly grammatically or syntactically and do not have significant ability to consider semantics of natural language. Subjects, verb chains and objects of the sentence are extracted syntactically but not semantically. As a result, semantic actions (verb chains) can be recognized only if they are described by finite verbs and generally can not be recognized if the actions are described by non-finite verb forms, like Infinitive, Participle I, Participle II, Gerund and Verbal Nouns. Because prior known systems or methods lack a meaningful semantic analysis capability, the problem of recognition of semantic relations for subjects, actions, objects that go beyond one sentence and the problem of hierarchical and synonymical relations between triplets are not addressed by the prior art. Even grammatical analysis is often incomplete for complex sentences and sentences with unknown words (not listed in the dictionary in advance).

Furthermore, prior art systems are often too inaccurate, extremely slow, and are not suitable for working with considerable amounts of text which limits their value for industrial or commercial level applications.

SUMMARY OF EXEMPLARY EMBODIMENT OF THE PRESENT INVENTION

One exemplary embodiment according to the principles of the present invention includes a system or method of semantically processing natural language representations in a general purpose computer including entering and storing a user criterion, entering into a first storage area representations of the texts of a plurality of natural language documents that have some relationship with the stored user criterion, formatting said representations and storing the formatted text in a second storage area, identifying and extracting from the second storage area subject-action-object (SAO) triplets and storing the SAO extractions in a third storage area, lemmatizing and storing the SAO extractions, generating a Problem Folder for each stored lemmatized SAO extraction and designating the AO portion as the name of the Problem Folder, storing in the Problem Folder a list of one or more possible solutions comprising the subject portions of the one or more stored lemmatized SAO triplets.

DRAWINGS:

These and other objects, features and benefits of the method and system according to the principles of the present invention will become apparent with the following detailed description when taken in view of the appended drawings, in which:

Figure 1 is a pictorial representation of one exemplary embodiment of the system according to the principles of the present invention.

Figure 2 is a schematic representation of the main architectural elements of the system and functional links according to the present invention.

Figure 3 is a structural and functional schematic representation of Unit 18 of Figure 2.

Figure 4 is a structural and functional schematic representation of Unit 20 of Figure 2.

Figure 5 is a structural and functional schematic representation of Unit 22 of Figure 2.

Figure 6 is a schematic representation of Unit 42 of Figure 4.

Figure 7 is a schematic representation of Unit 44 of Figure 4.

Figure 8 is a schematic representation of Unit 46 of Figure 4.

Figure 9 is a schematic representation of Unit 26 of Figure 2.

Figure 10 is a typical example of the text to be semantically processed.

Figure 11 is a representation of formatted text of Figure 10.

Figure 12 is a representation of error corrected text of Figure 11.

Figure 13 is a representation of word-splitted text of Figure 12.

Figure 14 is a representation of sentence-splitted text of Figure 13.

Figure 15 is a representation of tagged text of Figure 14.

Figure 16 is a representation of parsed text of Figure 15.

Figure 17 is a representation of SAO DB extracted from parsed text of Figure 16.

Figure 18 is a representation of lemmatized SAO DB of Figure 17.

Figure 19 is a typical example of relevant SAO DB entry of Figure 18.

Figure 20 is a representation of a Problem Folder generated in response to the relevant SAO DB of Figure 18.

Figure 21A is a representation of three original input texts from various sources.

Figure 21B is a representation of the output structured SAO KB resulting from process step 60.

DETAILED DESCRIPTION OF AN EXEMPLARY EMBODIMENT

Note the glossary at the end of this detailed description which will assist the reader.

One exemplary embodiment of SAO Semantic Processor according to the principles of the present invention includes (Fig.1) a CPU 4 with MODEM and/or cable box 5 that could comprise a general purpose computer or networked server or minicomputer with standard user input and output device such as keyboard 10, mouse 8, printer 6 and monitor 2 and/or other user data entry device 9.

With reference to Figures 2-9, the SAO Semantic Processor (Fig.2) includes a database 16 of original documents for receiving and storing documents downloaded from the web 14 or local database 12 or generated as a user request text with the use of keyboard 10 or other input devices, such as a scanner or voice recognition device.

Preformator 18 receives the document data 28 from the database 16, removes formatting symbols and other symbols that are not part of natural language text (Unit 30), identifies and corrects automatically the mismatches and mistakes (Unit 32), divides the text into words (Unit 34) and sentences (36) and supplies the output 38 to the SAO Extractor 20.

Unit 30 in preformator 18 (Fig.3) removes from the input text 28 all the formatting, images, tables, converts different text formats to ASCII text, and recognizes unknown words not stored in one of the dictionaries described below. Preformator's Unit 32 detects in the input text 28 all the spelling errors in accordance with four basic types of errors (substitution, omission, transposition, insertion) and corrects the above errors automatically. Also the Preformator splits the text into words (Unit 34) and sentences. Complex sentences are divided or organized into simple sentences (Unit 36).

SAO Extractor Unit 20 (Figure 4) tags the text with part-of-speech tags (Unit 42), parses (Unit 44) the text 40 syntactically, recognizes Subjects, Actions and Objects, their attributes, Cause-Effect relations between SAO-triplets and builds the Syntactical Tree of each sentence of the text 40 (unit 46), which then outputs to the SAO Editor (Unit 22). The Preformator supplies the Formatted text 38 to the input of the SAO Extractor (Fig.4). SAO Extractor uses Linguistic Knowledge base in order to tag the Formatted text 40 with part-of-speech tags (Unit 42). There are preferably three stages in POS tagging process. First, a context-independent analysis module (Unit 68) assigns each word of the text 66 a set of one or more part-of-speech tags. Then the disambiguation context-dependent module based on statistical Hidden Markov Model algorithm assigns each word of the text a unique part-of-speech tag (unit 70). Next, the Unit 72 uses a rule-based POS tagging module to perform the correction of the output of the Unit 70 and recognition of unknown words, forming the POS-tagged text 74 as the output. Then the SAO Extractor performs the Text Parsing (Unit 44, Figure 4), which includes the steps presented in Figure 7. Theses steps are verb group (Unit 80) and noun group (Unit 82) extraction and Functional Tree construction (Unit 84).

The parsed text 86 is supplied to the Unit 46 which extracts SAOs from the Parsed text 90 (Fig. 8). At first, SAOs with finite verbs as Actions are extracted where Action type recognized in Unit 100 enables Unit 102 to extract Subject and Object from the right or the left part of the Action. Then SAOs with Actions as non-finite verb forms are recognized in Unit 104 and as verbal nouns recognized in Unit 106. All Subjects and

Objects attributes (location, composition, etc.) are recognized in Unit 108. Next, Unit 109 recognizes Cause-Effect relations for SAO-triplets. As the result, the SAO Extractor 20 outputs the Functionally analyzed text 112 which serves as input to the SAO Editor (Unit 22).

SAO Editor Unit 22 (Figures 2 and 5) performs the lemmatization of Actions (unit 54), Subjects and Objects (Unit 56), filters SAOs (unit 58), structures (unit 60) them and forms the SAO Knowledge base (Unit 24). SAO Editor performs Action lemmatization (Unit 54, Fig.5), i.e. converts all Actions to infinitive, Subject and Object lemmatization during which Subject and Object are represented or described by noun(s) Unit 56. For example:

sets → set

sets of processors → set (of) processor

In addition to all the above mentioned features, SAO Editor provides the possibility to filter SAOs, i.e. to remove from SAO database SAOs (Unit 52) not irrelevant to the analyzed text (Unit 58), to structure SAOs (Unit 60), i.e. to set the relevant SAOs in synonymical and hierarchical structures. The resulting SAO Knowledge Base (Unit 24) includes SAO database and various tools for analyzing SAOs and building synonymical and hierarchical relations.

System modules (Units 18,20, 22) access Linguistic Knowledge Base (Unit 26), one example of which structure is presented in Figure 9. Linguistic Knowledge Base includes Database section (1) and Database of Recognizing Linguistic Models section (2), which describes algorithms for recognizing linguistic objects and relations in the text. Preformator (Unit 18) accesses and is controlled by information stored in blocks (3), (4), (5), (10), (12), (13), (14). SAO Extractor (Unit 20) accesses and is controlled by information stored in blocks (3), (4), (5), (6), (8), (9), (10), (11), (12), (15), (16), (17), (18), (19), (21), (22). SAO Editor (Unit 22) accesses and is controlled by information stored in blocks (4), (12), (20).

The method and apparatus of the present invention provide the user with the possibility of automatically extracting World Knowledge from text and storing it in the form of SAOs where SAOs can be lemmatized and unified into complex hierarchical structures using their attributes and meanings which in turn can help extract other types of knowledge which will use natural language facts and dependencies to reflect the real world regularities and information.

The Linguistic KB (Figure 9) will now be described.

Classifier (3)

The classifier contains a list of tags which are traditionally called part-of-speech tags. The list includes tags for nouns, verbs, adjectives, adverbs, prepositions, etc. But these tags should be more detailed than traditional parts of speech. For example, words that combine noun and adverb functions have a separate tag. Punctuation symbols also have appropriate tags. Other tag classes are names of devices, systems, enterprises that can be treated as semantic tags. See US Pat. No. 4,868,750 (Kucera , et al.) and US Pat. No. 4,887,212 (Zamora , et al.) for examples of implementation of a classified database.

Examples of tags:

NN	— common noun, singular
NNS	— common noun, plural

These part-of-speech tags are used for assigning each word in the main Dictionary (4) a set of tags that it can have. For example, as the word "well" can be a noun(NN), an adverb(RB), and an adjective(JJ), the dictionary entry can be the following:

Well — RB, NN, JJ

One suitable list of codes is published in Publication No. 1 that discloses 154 codes in the list.

Main Dictionary (4)

In the Main Dictionary, each stored word is linked with a set of part-of-speech tags that it can have in the text, for example,

Absorb — VB

Abstract — JJ, NN, VB

Well — RB, NN, JJ

It is used for spelling correction, automatic part of speech tagging, syntactic analysis, semantic analysis — anywhere part of speech analysis is used.

If we describe the text as a chain of words

$w_1 \quad w_2 \quad w_3 \quad \ldots \quad w_n$

Then after a first look-up in the Main Dictionary we obtain:

$w_1K_1 \quad w_2\,K_2 \quad w_3\,K_3 \quad \ldots \quad w_n\,K_n$

where K_i, i = 1, n is a set of tags from the Main Dictionary which, in one exemplary embodiment of the present invention, contains about 60,000 English words. Further details of a Main Dictionary are published in Publication No. 1.

Dictionary of Abbreviation (5)

This is a database list of abbreviations. Each abbreviation is assigned one part of speech tag, e.g.

A.C. — NNU

where AC means alternating current

i.e. — RB

Because abbreviations act just like ordinary words in the text, abbreviations dictionary is quite similar to the Main Dictionary.

Idiomatic Dictionary (6)

This is a database list of idioms. The Idiomatic Dictionary comprises set expressions and idioms. Each idiom or unit is assigned a part-of-speech tag or a set of part-of-speech tags, e.g.

go into detail — VB

a great deal of — ABL

In one exemplary embodiment of the present invention, the Idiomatic Dictionary contains 2200 idioms. It is well known that part of speech properties of idioms can not be obtained by analyzing words that constitute idioms. So, the use of idioms can dramatically improve the accuracy of part-of-speech tagging.

Dictionary of Parameters (7)

The Dictionary of Parameters is a database list of parameters that characterize objects of the outer world, i.e. inherent properties (features) of an object measured by a numerical number, i.e. weight, temperature, density, current strength, etc. This Dictionary contains in one exemplary embodiment of the present invention about 1250 parameters.

Syntactic Classifier (8)

This is a database of Syntactic Classifier of lexical items and relations. Includes syntactic classes (codes), which are used for classification of structural elements of syntactically analyzed sentences which are optimized for further SAO extraction.

One of the most widely used syntactic classifier of this kind is described in Penn Treebank Project [Bracketing guidelines for Treebank II style Penn Treebank Project, January 1995, University of Pennsylvania]. More information about Penn TreeBank Project can be found at http://www.cis.upenn.edu/~treebank/home.html. Another example of syntactic classifier can be found in U.S. Patent No. 5,878,385. Syntactic classifier (8) differs from the prior art classifiers, because it includes new unique codes, for example, the following codes:

w_NN — code for noun group

w__VBN_XX — code for one verb chain pattern

w__Sentence — code for sentence

These codes enable generation of or formatting of certain linguistic structures (noun groups, verb chain patterns and sentences as basic elements of semantic analysis) that are important for further SAO extraction.

Semantic Classifier (9)

This is a database list of Semantic Classifier of lexical items and relations. It includes semantic classes (codes) of SAO triplets, their components S,A,O, A-O and their attributes and relations, including cause-effect, object-parameter and other relations. It is used for classifying structural elements of sentence trees.

Probabilistic Grammar (10)

This is a database of probabilistic Grammar of Natural Language. This Grammar uses main lexicon, Idiomatic Dictionary and abbreviations, an algorithm of looking up the main lexicon, Idiomatic Dictionary and abbreviations and also an algorithm for word part of speech disambiguation, i.e. determining word part of speech using context. This Probabilistic Grammar provides means for automatically annotating the text with part of speech information(Units 66, 68, 70). The algorithm is based on the known Hidden Markov Model and uses statistical data from block 12.

Rule-Based Grammar (11)

This is a database of Rule-Based Part of Speech Tagging module. It includes rules and rules processor which detects erroneous output of Unit 70 and corrects it. Example of rule is: *If the Unit 70 outputs a sequence of article, such as "a" or "the", and verb, this sequence should be replaced by the sequence of article and noun.*

This Rule-Based Grammar is used as the final step of part-of-speech tagging process.

Linguistic Facts (12)

The Linguistic Facts module contains Filters Database, Dictionary Word-Code-Frequency, Statistical Matrix Code-Code and so on. Filters database includes a list of lexical items and their codes which are considered to be non-informative by knowledge engineers. This information is used by SAO Editor (Fig.5) which checks if it should include a given SAO into SAO Knowledge Base. Other above mentioned components of the Linguistic Facts module are used in Probabilistic Grammar (Unit (10)) for part-of-speech tagging of text.

Error Detection and Correction (13)

The Error Detection and Correction module contains Recognizing Linguistic Models (algorithms) for automatic spelling corrector. Algorithms of the automatic spelling corrector are detailed in Publication No. 1. Similar algorithms are described in Publication No. 3. This module is used to detect and correct four basic types of spelling errors (substitution of a symbol in a word, omission of a symbol, shift of adjacent symbols, insertion of a superfluous symbol). Unit 32 (Fig.3) uses Recognizing Linguistic Models in order to find errors, determine their types and select a set of words for correction without using context. The Probabilistic Grammar, Unit (10), calculates the most likely word from the above mentioned set of words and corrects the spelling error automatically. If the word length is more than 5 letters, the word is checked for a combination of any two out of four basic types of errors.

Splitter (14)

These are stored Recognizing Linguistic Models for Text to Words and Text to Sentences splitting. The Unit (14) uses formal characteristics like spaces, capital letters and punctuation for determining word and sentence boundaries. The splitter is used by Preformator (Figure 3).

Idiom/Set Expression Recognizer (15)

These are stored Recognizing Linguistic Models for idiom recognition. The model described in-depth in Publication No. 1 can be used. It provides an Idiomatic dictionary (Unit 6) to recognize idioms in the text during the first stage of part of speech tagging (Unit 42). Each idiom is assigned a part of speech tag from a list of tags that it can have. The algorithm tends to recognize the longest idiom with a given first word.

Verb Chain Recognizer (16)

This Recognizer includes Recognizing Linguistic Models for Verb Chains Recognition. These Models use part-of-speech tagged text (Unit 78) and rules for extracting verb chains in the text. Rules can be described in Backus Naur Form, e.g. <present perfect passive>::=<HVZ><BEN><VBN> is a pattern for extracting verb chains like “has been produced”.

Noun Group Recognizer (17)

This Recognizer includes Linguistic Models for Noun Group Recognition. They can also be described in Backus Naur Form. Noun group recognition rules use part-of-speech tagged text and lexemes (such as prepositions, conjunctions and adverbs) in order to extract noun groups, keeping the information on internal structure of noun groups, which is used during next steps of SAO analysis(Subject and Object extraction, Subject and Object lemmatization).

Tree Construction (18)

This module includes are stored Recognizing Linguistic Models for Functional and Syntactic Phrase Tree Construction. They describe rules for structurization of the sentence, i.e. for correlating part-of-speech tags, syntactic and semantic classes, etc. which are used by Text parsing (Unit 44) and SAO extraction (Unit 46) for building Syntactic and Functional phrases.

SAO Recognition (19)

These are stored Recognizing Linguistic Models for Subject, Action and Object extraction. They describe rules that use part-of-speech tags, lexemes and syntactic categories which are then used to extract from the parsed text (Unit 90) SAOs with finite actions (Units 100, 102), non-finite actions (Unit 104), verbal nouns (Unit 106). One example of an Action extraction is:

<HVZ><BEN><VBN> □ (<A>=<VBN>)

This rule means that "if an input sentence contains a sequence of words w_1, w_2, w_3 which at the step of part-of-speech tagging obtained HVZ, BEN, VBN tags respectively, then the word with VBN tag in this sequence is in Action". For example, has _HVZ been_BEN produced_VBN □ (A=produced)

There are more than one hundred rules in on exemplary embodiment of the present invention for action extraction.

Lemmatization (20)

These are stored Recognizing Linguistic Models for Lemmatization of Subject, Action and Object. They describe rules that use part-of-speech tags, lexemes and syntactic categories which are then used by SAO Editor (Fig. 5) while lemmatizing Actions (unit 54), Subjects and Objects (Unit 56).

Below are examples of such rules for Action and Object Lemmatization respectively:

<VBN>	□	<Infinitive>
produced_VBN	□	produce
<NNS>	□	<NN>
processors_NNS	□	processor

Subject-Object Attributes (21)

These are stored Recognizing Linguistic Models for detecting Subject and Object attributes. These models describe rules (algorithms) for detecting subjects, objects, their attributes (placement, inclusion, parameter, etc.) and their meanings in syntactic tree. These algorithms work with noun groups and act like linguistic patterns that control extraction of above-mentioned relations from the text. For example, for the relations of type parameter-object, basic patterns are

<Parameter> of <Object>

and

<Object> <Parameter>

The relation is valid only when the lexeme which corresponds to <parameter> is found in the list of parameters included in block (7).

These models are used by Unit 108 of SAO extraction module (Fig. 8).

SAO Semantic Relations (22)

These are stored Recognizing Linguistic Models for detecting semantic relations of SAOs. These models describe algorithms for detecting cause-effect relations between SAOs. These models use linguistic patterns, lexemes and predefined codes from a list of codes. These patterns describe the location of cause and effect in the input sentence. For example, the condition *when caused + TO + VB*

shows that the Cause is to the right of the word *caused* and is expressed by an infinitive and a noun group that follows it. The Effect is to the left from the word *caused* and is expressed by a noun group, e.g.

The network termination unit includes a plurality of semi-conductor switches electrically connected to conductors of the telephone line to establish a network of electrical paths capable of altering the electrical conduction of the telephone line when caused to assume a state of conduction.

These models are used by SAO extraction module (Fig.8, Unit 109).

Figures 10-19 show the results of various process steps designated in the respective figure for the sentence:

"As the ambubag is squeezed by the control unit, the pressure-sensitive device moves the air through the conducting lumen and into the intubated patient's airway."

The user in this example entered: "How to move air." Figure 20A shows related text from documents found on the web, for example, and stored.

Figure 20B shows the Problem Folder for this task with each of the four possible solutions listed along with their citations and quotes generated by the present system and method and displayed for the user on monitor 2.

Publications and Patents incorporated herein by reference:

1. *Sovpel, I.V.* Injenerno-lingvisticheskie prinzipi, metodi i algoritmi avtomatizirovannoi pererabotki teksta. — Minsk: Visheishaya Shkola, 1991, — 116 p.

2. *Zamora, Antonio* "Automatic Detection and Correction of Spelling Errors in a Large Data Base," Journal of the American Society for Information Science, 31(1), pp. 51-57, 1980.

3. All patents mentioned throughout this specification are incorporated herein by reference.

GLOSSARY:

1. ***Action***: is the constituent that is expressed either by a finite verb or non-finite verb or verbal noun and denotes a relation between Subject and Object.

2. ***corpus*** (pl. corpora or corpuses): a collection of text in machine-readable form, compiled to be representative of a particular kind of language and provided with some kind of additional information.

3. ***Functional Tree of Sentence***: is a syntactical tree of the sentence where SAO-triples and semantic relations for them are recognized.

4. ***lemmatization***: the process or result of dividing a text into sets of different forms of a word, such as the inflected forms of a verb. Ex. "sing", "sang", "sung" are one lemma, "boy", "boys" another.

5. ***Linguistic KB (knowledge base)***: a database of (i) Recognition Linguistic Models and (ii) a database of linguistic rules and dictionaries [see for example Figure 9].

6. ***NL***: Natural Language

7. ***Object***: is the constituent that is affected by the Action, e.g. John likes Mary. Object is "Mary" because "Mary" supports the Action.

8. ***parsing***: the process or result of making a syntactic analysis of the natural language text.

9. ***parser***: toll (often automatic or semi-automatic computer program) used for parsing the text.

10. ***part-of-speech (POS)***: word class, such as verb, noun, adjective.

11. ***part-of-speech tagging***: assigning part-of-speech tags to a text.

12. ***part-of-speech tag***: a label associated with a word (or other unit) providing information about the word, or the process of assigning tags. Ex: "run" can be tagged as a noun (run_NN) or verb (run_VB).

13. ***Problem Folder***: Computer storage address and area for storing structured SAO KB entries in problem statement form (e.g. "How to move air") and a plurality of possible solutions (i.e., subjects from all documents related to the problem), the document citations thereof, the full sentence in which the subject and SAO appears with the subject, action and object preferably highlighted.

14. ***Recognizing Linguistic Models***: are linguistic algorithms for recognizing and transforming certain linguistic objects and their relations in a text. The models are described as rules using lexical units, tags, syntactic and semantic categories.

15. ***SAO***: Subject-Action-Object

16. ***SAO-DB (database)***: is a database of SAO-triples and semantic relations.

17. ***SAO-KB (knowledge base)***: includes SAO-DB, set of rules for structurizing SAO-DB and tools for managing SAO-DB.

18. ***SAO Triple***: SAO-triplet

19. ***SAO Triplet***: is a set of Subject, Action and Object, related one with another.

20. ***Semantic Relations for SAO triples***: are semantic relations for separate components of SAOs (e.g. relations like Object-Parameter) and for SAO as a whole (E.g. relations like $SAO_1 \rightarrow SAO_2$ where SAO_1 is Cause and SAO_2 is Effect).

21. ***Storage Area***: either a separate storage facility in a general purpose computer or address designated storage within a general purpose computer.

22. ***Subject***: is the constituent that performs the Action, e.g. John likes Mary. Subject is "John" because "John" performs the Action.

23. ***Subject Attributes, Object Attributes***: is a property of a Subject (Object), e.g. parameter, placement.

24. ***Syntactic Tree of Sentence***: is a tree view of the sentence where nodes are syntactic categories and edges are dependencies between syntactic categories.

25. ***tag-classifier***: set of tags used for part of speech tagging.

WE CLAIM:

Claim 1. A method of semantically processing natural language representations in a general purpose computer comprising:

entering and storing a user criterion,

retrieving from remote and local databases and storing representations of the texts of a plurality of natural language documents that have some relationship with the stored user criterion,

formatting said representations and storing the formatted representation,

identifying and extracting from the formatted representation subject-action-object (SAO) extractions and storing the SAO extractions, processing the SAO extractions into normalized SAO structures and storing the SAO structures,

designating the AO portions as substantially the names of Folders of at least some of the SAO structures,

storing in association with each the Folder name the identity of one or more subject portions (S_1, S_2, ...S_n) that are associated with the respective AO portion of stored SAO structures.

Claim 2. The method of Claim 1 including storing in association with each respective S_1, S_2, ... S_n the full sentence in which the respective SAO appears.

Claim 3. The method of Claim 2 including highlighting each S-A-O portion that appears in each said full sentence.

Claim 4. The method of Claim 1 further including displaying the list of subjects (S_1, S_2 ...S_n) stored in association with a respective AO portion in response to the user selecting the displayed AO portion or folder name.

Claim 5. The method of Claim 1, further including storing in association with at least each subject (S) the respective sentence of the source document from which the respective SAO structure originated.

Claim 6. The method of Claim 1, further including storing in association with at least each subject (S) the citation to the source document from which the respective SAO structure originated.

AMENDED CLAIMS

[received by the International Bureau on 19 September 2000 (19.09.00);
original claims 1, 2 and 3 amended; new claims 7 and 8 added;
other claims unchanged (2 pages)]

Claim 1. A method of semantically processing, managing, and displaying natural language representations in a general purpose computer comprising:

retrieving from remote and local databases and storing representations of the texts of a plurality of natural language documents,

formatting said representations and storing the formatted representation,

identifying and extracting from the formatted representation subject-action-object (SAO) extractions and storing the SAO extractions, processing the SAO extractions into normalized SAO structures and storing the SAO structures,

designating the AO portions as substantially the names of Folders of at least some of the SAO structures,

storing in association with each Folder name the identity of one or more subject portions (S_1, S_2, ...S_n) that are associated with the respective AO portion of stored SAO structures displaying a plurality of Folder names, and

in response to user selection of a particular Folder name, displaying the subject portions (S_1, S_2, ...S_n) associated with the selected Folder name.

Claim 2. The method of Claim 1 including storing in association with each respective S_1, S_2, ... S_n the full sentence of the source document in which the respective SAO appears.

Claim 3. The method of Claim 2 including displaying in response to user selection of a particular S_1, S_2, ...S_n said full sentence and highlighting each S-A-O portion that appears in said full sentence.

Claim 4. The method of Claim 1 further including displaying the list of subjects (S_1, S_2 ...S_n) stored in association with a respective AO portion in response to the user selecting the displayed AO portion or folder name.

Claim 5. The method of Claim 1, further including storing in association with at least each subject (S) the respective sentence of the source document from which the respective SAO structure originated.

Claim 6. The method of Claim 1, further including storing in association with at least each subject (S) the citation to the source document from which the respective SAO structure originated.

Claim 7. The method of Claim 5, further including displaying said respective sentence in response to user selection of the respective subject (S).

Claim 8. The method of Claim 6, further including displaying said citation to the source document of said respective SAO.

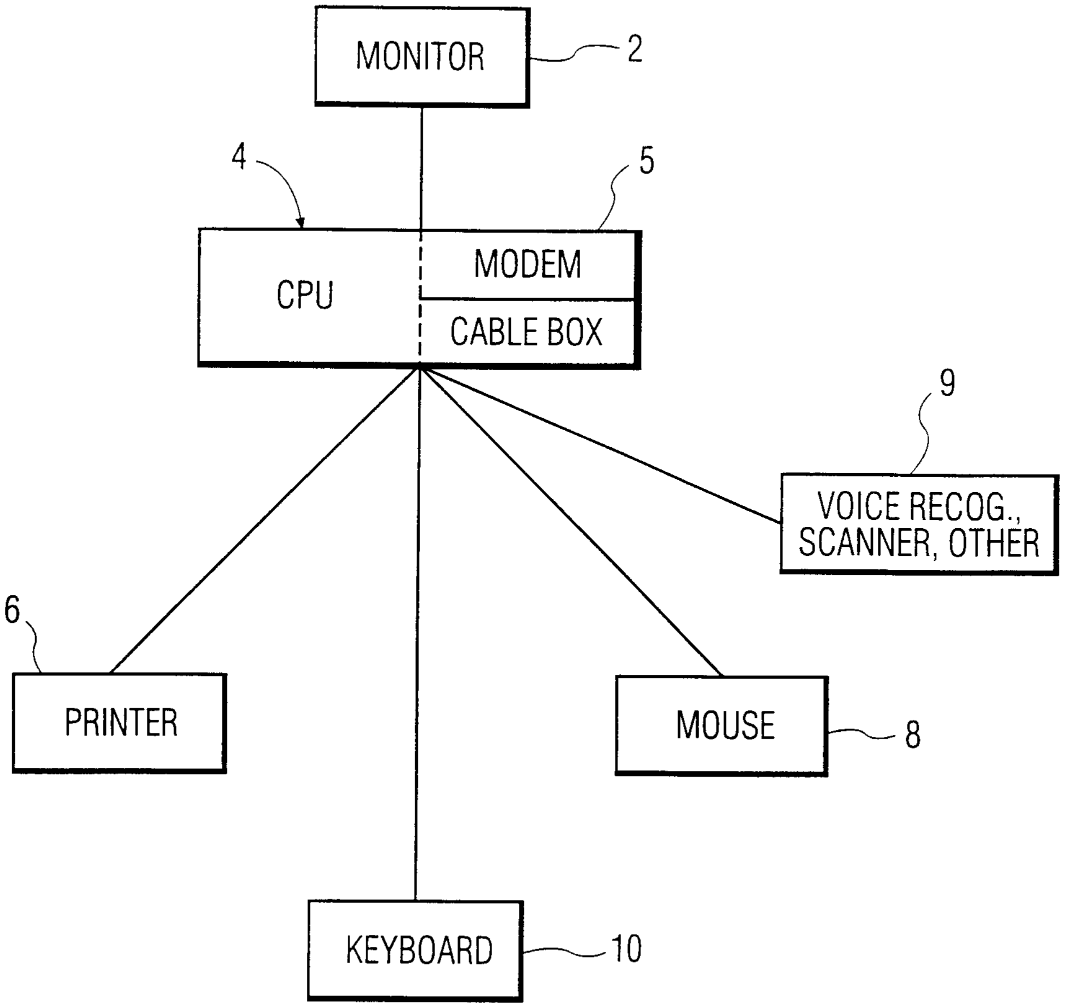

MONITOR
2
4
5
CPU
MODEM
CABLE BOX
9
VOICE RECOG., SCANNER, OTHER
6
PRINTER
MOUSE
8
KEYBOARD
10

FIG. 1

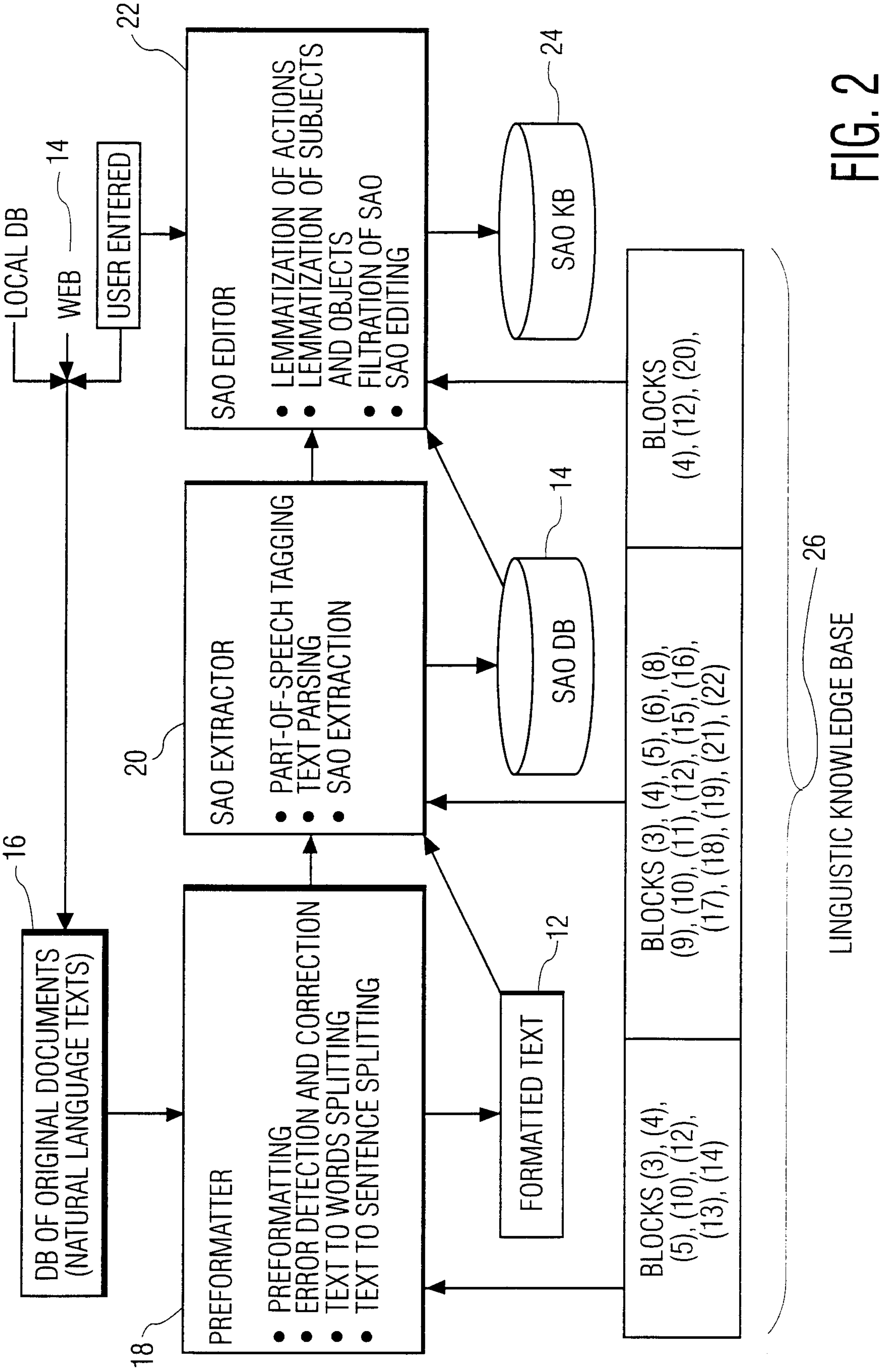

DB OF ORIGINAL DOCUMENTS (NATURAL LANGUAGE TEXTS)
16
LOCAL DB
WEB
USER ENTERED
14
PREFORMATTER
• PREFORMATTING
• ERROR DETECTION AND CORRECTION
• TEXT TO WORDS SPLITTING
• TEXT TO SENTENCE SPLITTING
18
SAO EXTRACTOR
• PART-OF-SPEECH TAGGING
• TEXT PARSING
• SAO EXTRACTION
20
SAO EDITOR
• LEMMATIZATION OF ACTIONS
• LEMMATIZATION OF SUBJECTS AND OBJECTS
• FILTRATION OF SAO
• SAO EDITING
22
FORMATTED TEXT
12
SAO DB
14
SAO KB
24
BLOCKS (3), (4), (5), (10), (12), (13), (14)
BLOCKS (3), (4), (5), (6), (8), (9), (10), (11), (12), (15), (16), (17), (18), (19), (21), (22)
BLOCKS (4), (12), (20),
26
LINGUISTIC KNOWLEDGE BASE

FIG. 2

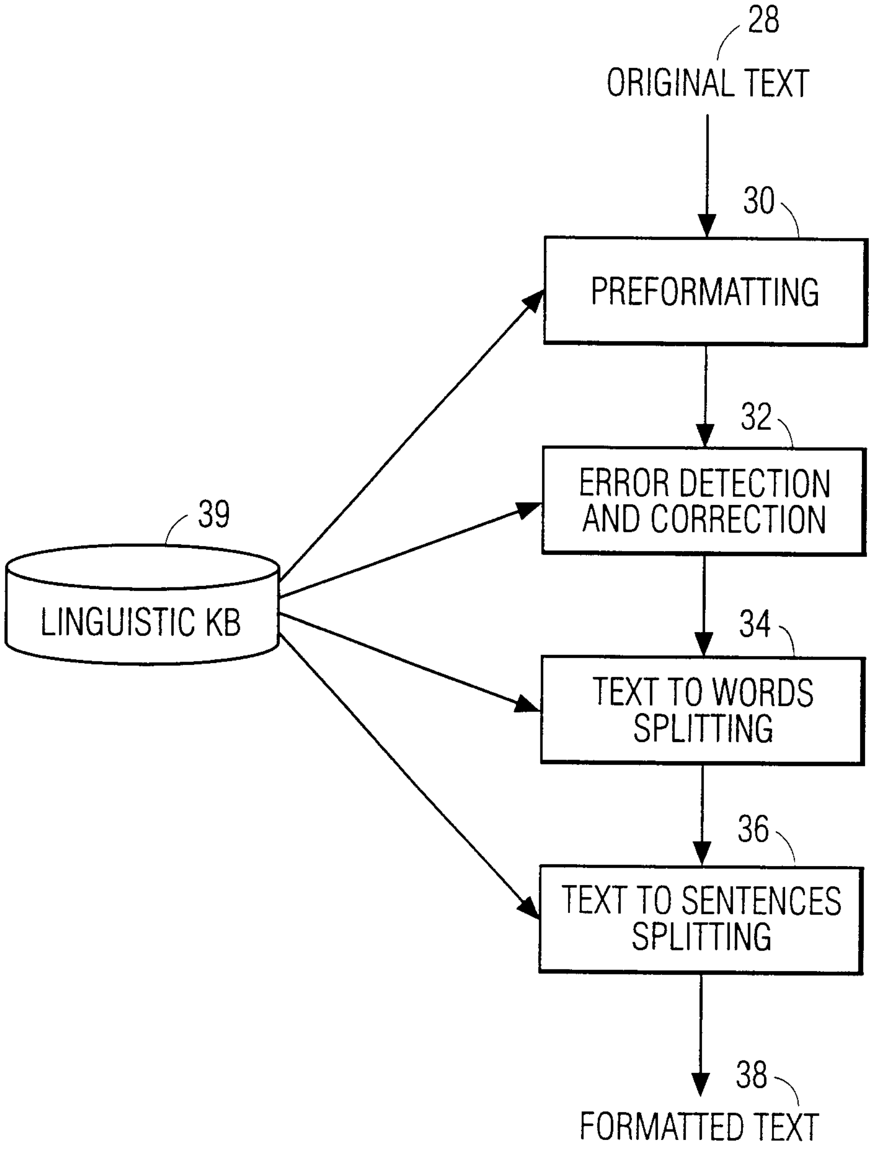

28
ORIGINAL TEXT
30
PREFORMATTING
32
ERROR DETECTION AND CORRECTION
34
TEXT TO WORDS SPLITTING
36
TEXT TO SENTENCES SPLITTING
38
FORMATTED TEXT
39
LINGUISTIC KB

FIG. 3

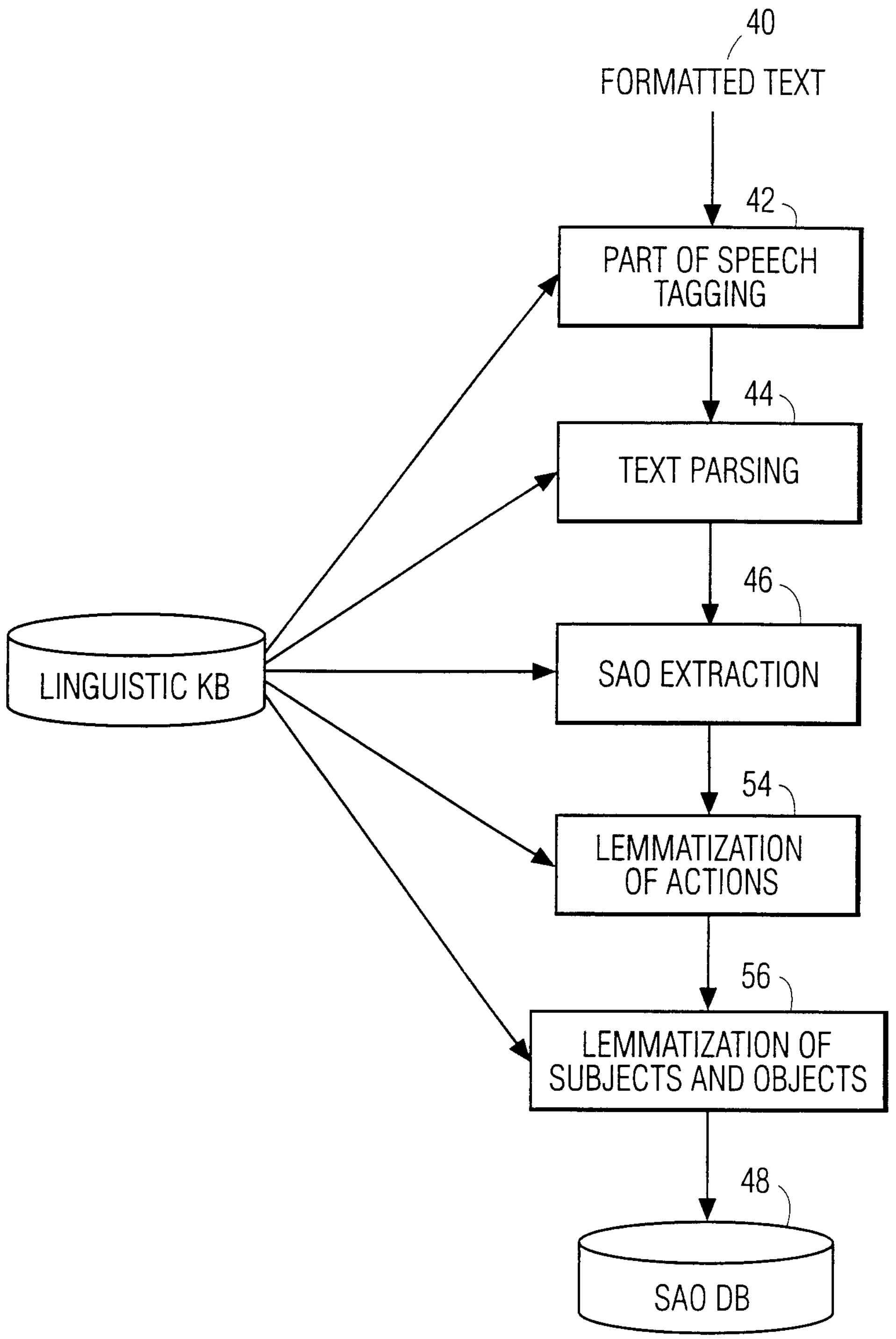

40
FORMATTED TEXT
42
PART OF SPEECH TAGGING
44
TEXT PARSING
46
SAO EXTRACTION
54
LEMMATIZATION OF ACTIONS
56
LEMMATIZATION OF SUBJECTS AND OBJECTS
48
SAO DB
LINGUISTIC KB

FIG. 4

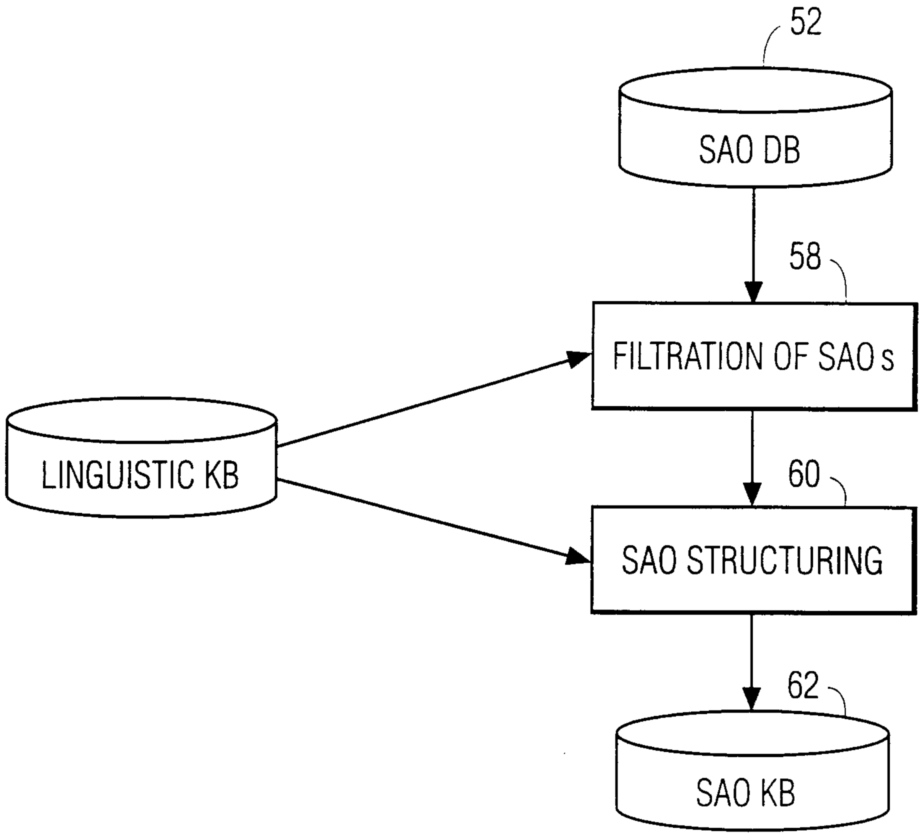

52
SAO DB
58
FILTRATION OF SAO s
LINGUISTIC KB
60
SAO STRUCTURING
62
SAO KB

FIG. 5

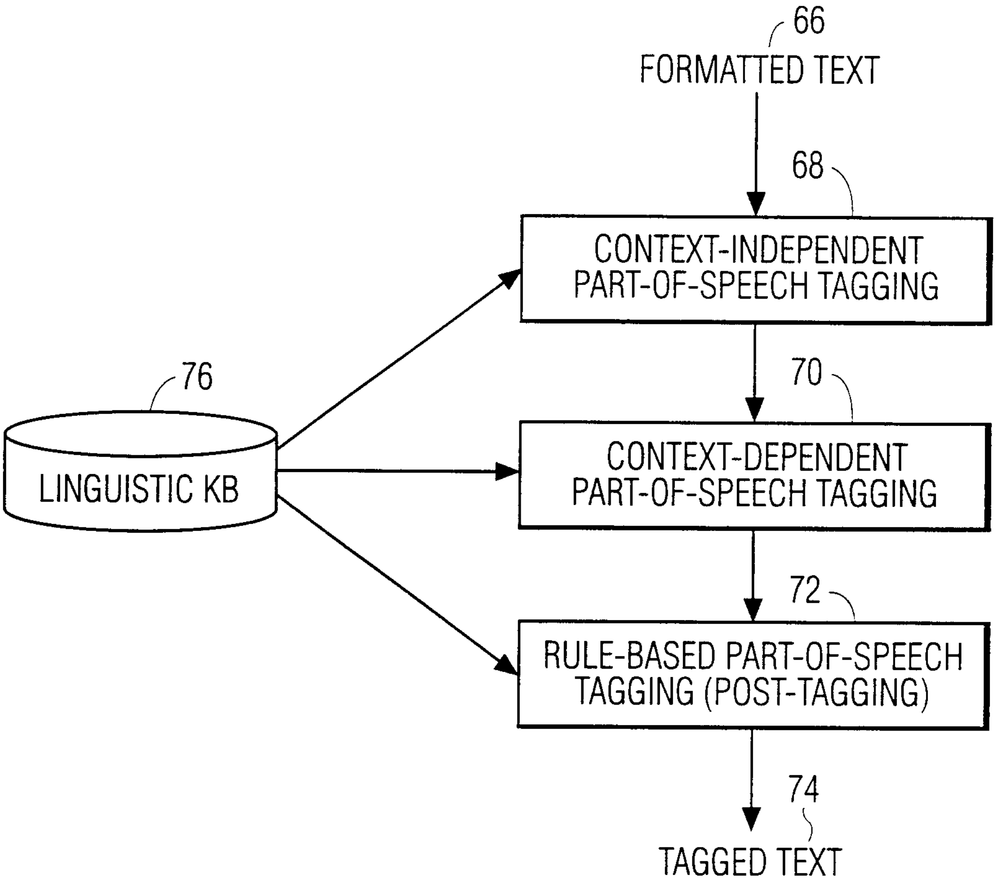

66
FORMATTED TEXT
68
CONTEXT-INDEPENDENT PART-OF-SPEECH TAGGING
70
CONTEXT-DEPENDENT PART-OF-SPEECH TAGGING
72
RULE-BASED PART-OF-SPEECH TAGGING (POST-TAGGING)
74
TAGGED TEXT
76
LINGUISTIC KB

FIG. 6

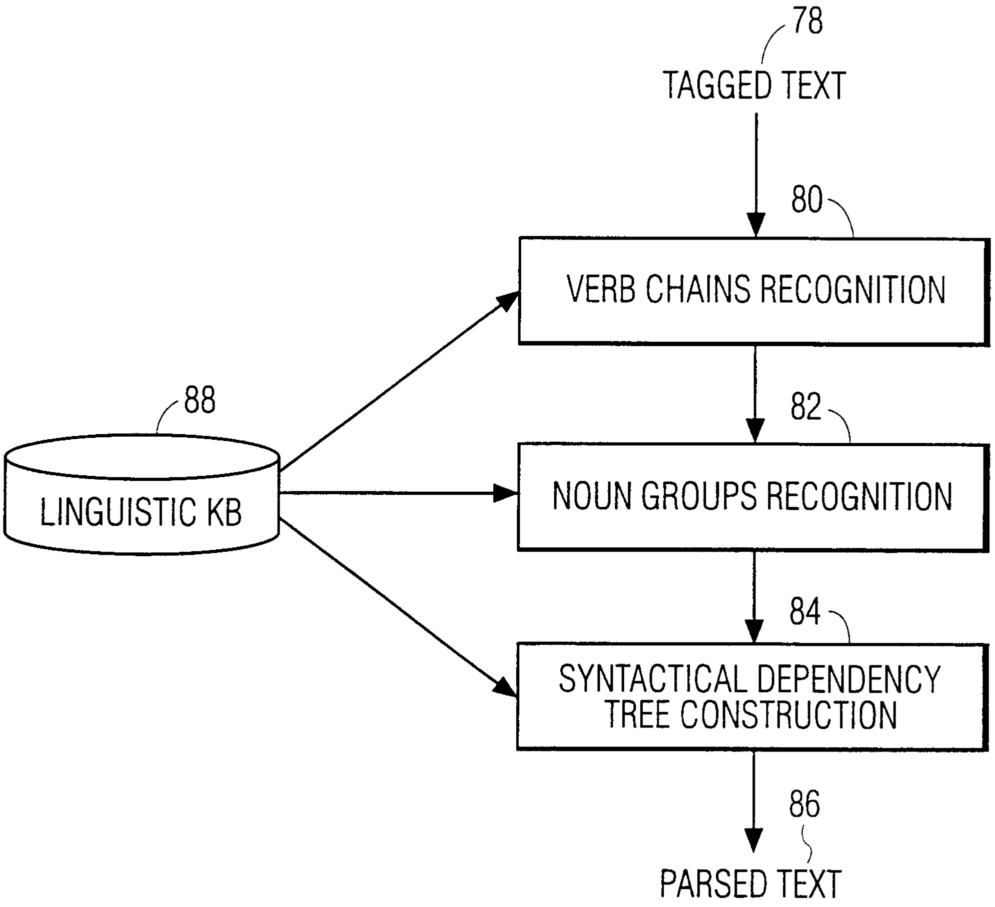

78
TAGGED TEXT
80
VERB CHAINS RECOGNITION
88
LINGUISTIC KB
82
NOUN GROUPS RECOGNITION
84
SYNTACTICAL DEPENDENCY TREE CONSTRUCTION
86
PARSED TEXT

FIG. 7

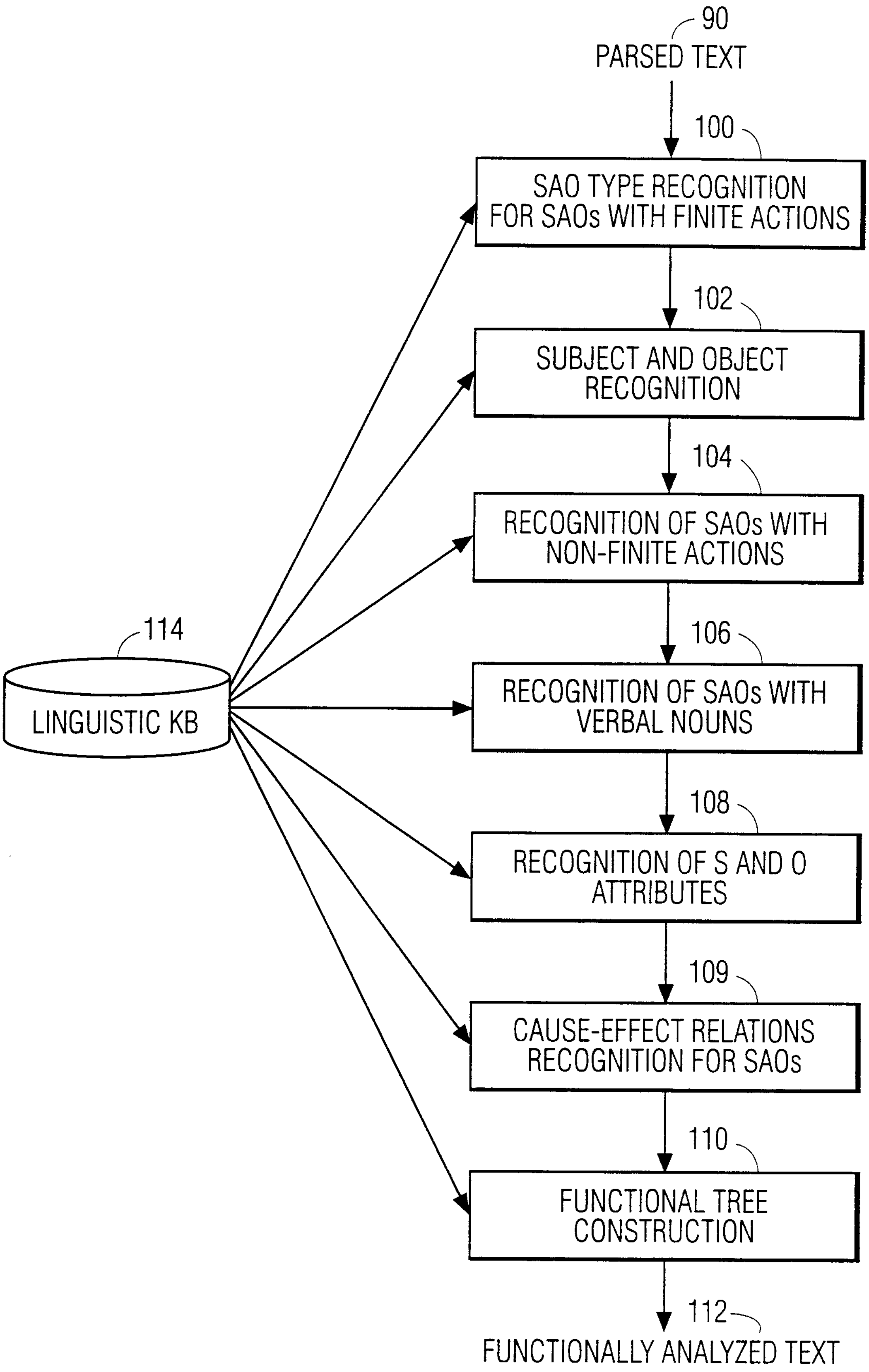
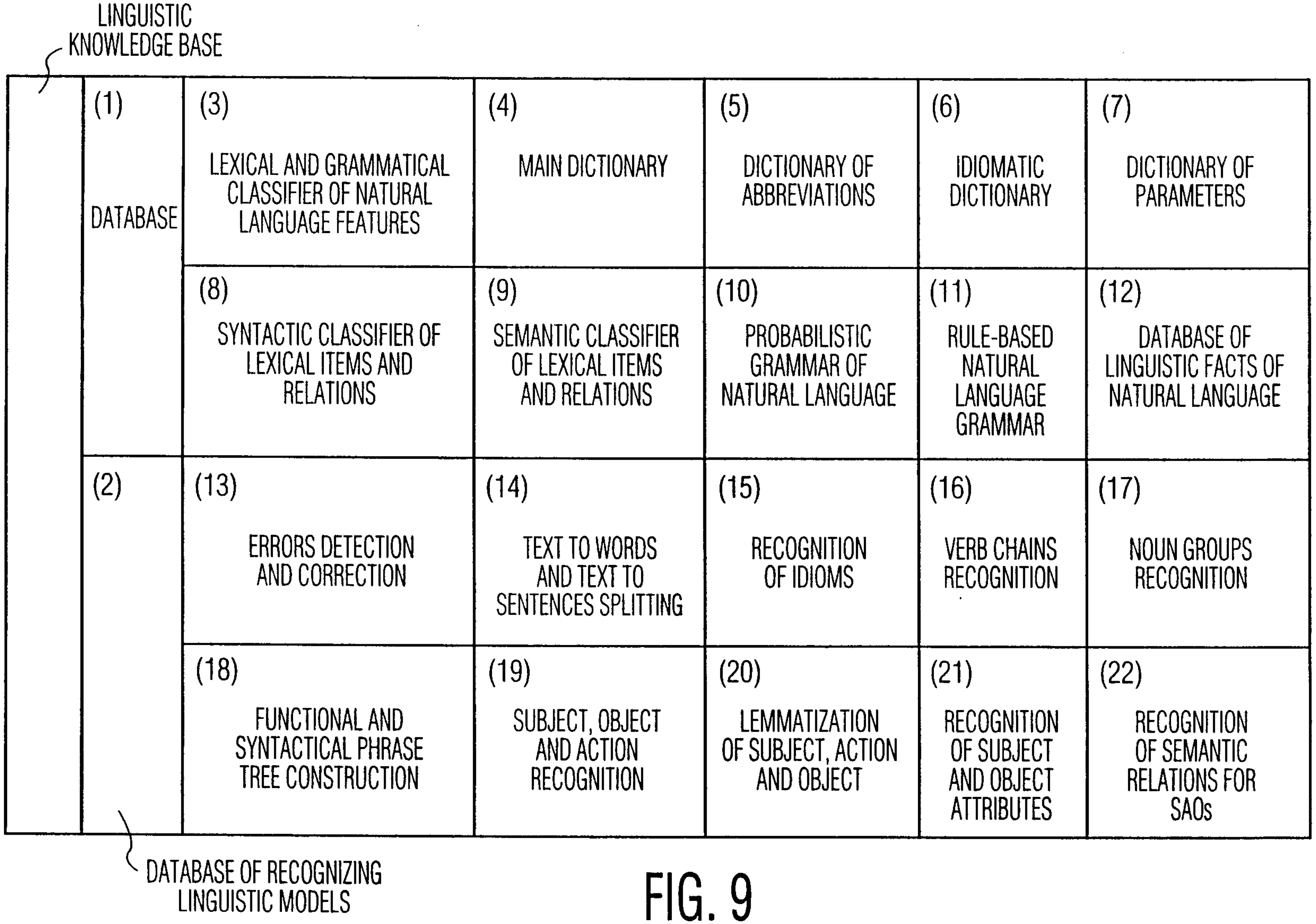

90
PARSED TEXT
100
SAO TYPE RECOGNITION FOR SAOs WITH FINITE ACTIONS
102
SUBJECT AND OBJECT RECOGNITION
104
RECOGNITION OF SAOs WITH NON-FINITE ACTIONS
114
LINGUISTIC KB
106
RECOGNITION OF SAOs WITH VERBAL NOUNS
108
RECOGNITION OF S AND O ATTRIBUTES
109
CAUSE-EFFECT RELATIONS RECOGNITION FOR SAOs
110
FUNCTIONAL TREE CONSTRUCTION
112
FUNCTIONALLY ANALYZED TEXT

FIG. 8



LINGUISTIC KNOWLEDGE BASE
DATABASE OF RECOGNIZING LINGUISTIC MODELS

(1) DATABASE	(3) LEXICAL AND GRAMMATICAL CLASSIFIER OF NATURAL LANGUAGE FEATURES	(4) MAIN DICTIONARY	(5) DICTIONARY OF ABBREVIATIONS	(6) IDIOMATIC DICTIONARY	(7) DICTIONARY OF PARAMETERS
	(8) SYNTACTIC CLASSIFIER OF LEXICAL ITEMS AND RELATIONS	(9) SEMANTIC CLASSIFIER OF LEXICAL ITEMS AND RELATIONS	(10) PROBABILISTIC GRAMMAR OF NATURAL LANGUAGE	(11) RULE-BASED NATURAL LANGUAGE GRAMMAR	(12) DATABASE OF LINGUISTIC FACTS OF NATURAL LANGUAGE
(2)	(13) ERRORS DETECTION AND CORRECTION	(14) TEXT TO WORDS AND TEXT TO SENTENCES SPLITTING	(15) RECOGNITION OF IDIOMS	(16) VERB CHAINS RECOGNITION	(17) NOUN GROUPS RECOGNITION
	(18) FUNCTIONAL AND SYNTACTICAL PHRASE TREE CONSTRUCTION	(19) SUBJECT, OBJECT AND ACTION RECOGNITION	(20) LEMMATIZATION OF SUBJECT, ACTION AND OBJECT	(21) RECOGNITION OF SUBJECT AND OBJECT ATTRIBUTES	(22) RECOGNITION OF SEMANTIC RELATIONS FOR SAOs

FIG. 9

```
<FONT FACE="Courier New" SIZE=5> <P>As the
ambubag is squezed by the control unit, </p>
<P>the pressure-sensitive device moves the air through the air</P>
<P>conducting lumen and into the</P>
<P>intubaetd  patient's airway.</P> </FONT>
```

FIG. 10

As the ambubag is squezed by the control unit, the pressure-sensitive device moves the air through the air conducting lumen and into the intubaetd patient's airway.

FIG. 11

As the ambubag is squeezed by the control unit, the pressure-sensitive device forces the air through the air conducting lumen and into the intubated patient's airway.

FIG. 12

As the ambubag is squeezed by the control unit, the pressure-sensitive device moves the air through the air conducting lumen and into the intubated patient's airway.

FIG. 13

```
<s>
As
the
ambubag
is
squeezed
by
the
control
unit
,
the
pressure-sensitive
device
moves
the
air
through
the
air
conducting
lumen
and
into
the
intubated
patient
's
airway
.
</s>
<s>
(...next sentence...)
</s>
```

FIG. 14

```
As_CS
the_ATI
ambubag_NN
is_BEZ
squeezed_VBN
by_IN
the_ATI
control_NN
unit_NN
,_,
the_ATI
pressure-sensitive_JJ
device_NN
moves_VBZ
the_ATI
air_NN
through_IN
the_ATI
air_NN
conducting_NN
lumen_NN
and_CC
into_IN
the_ATI
intubated_JJed
patient_NN
's_POS
airway_NN
._.
```

FIG. 15

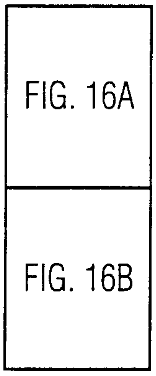


FIG. 16

```
<ComplexSentence_CE>
        <As_CS>                                  As_CS
        <w__Sentence>
            <w_NN>
                 <the_ATI>                       the_ATI
                 <ambubag_NN>                    ambubag_NN
            <w__BEX_xx>
                 <is_BEZ>                        is_BEZ
                 <w__VBN_XX>
                      <squeezed_VBN>             squeezed_VBN
                      <w__IN_N>
                           <by_IN>               by_IN
                           <w_NN>
                                <the_ATI>        the_ATI
                                <w_NN>
                                    <control_NN> control_NN
                                    <unit_NN>    unit_NN
       <,_,>                                  ,_,
```

FIG. 16A

```
 <w__Sentence>
     <w_NN>
            <the_ATI>                               the_ATI
            <w_NN>
                  <pressure-sensitive_JJ>           pressure-sensitive_JJ
                  <device_NN>                       device_NN
      <w_VBZ_XX>
           <moves_VBZ>                              moves_VBZ
           <w__N_XX>
                  <w_NN>
                      <the_ATI>                     the_ATI
                      <air_NN>                      air_NN
                  <w__IN_N_HP_CC>
                      <w__IN_N>
                           <through_IN>             through_IN
                           <w_NN>
                                 <the_ATI>          the_ATI
                                 <w_NN>
                                      <air_NN> air_NN
                                      <w_NN>
                                           <conducting_NN> conducting_NN
                                           <lumen_NN> lumen_NN
                     <w__CC>
                           <and_CC>                 and_CC
                     <w__IN_N>
                           <into_IN>                into_IN
                           <w_NN>
                                 <the_ATI>          the_ATI
                                 <w_NN>
                                     <w_NN$>
                                          <w_NN>
                                                 <intubated_JJed> intubated_JJed
                                                 <patient_NN> patient_NN
                                          <'s_POS> 's_POS
                                     <airway_NN> airway_NN
<._.>                                     ._.
```

FIG. 16B

Subject	Action	Object
the control unit	is squeezed	the ambubag
the pressure-sensitive device	moves	the air

FIG. 17

Subject	Action	Object
control unit	squeeze	ambubag
pressure-sensitive device	move	air

FIG. 18

Subject	Action	Object
pressure-sensitive device	move	air

Note: The user is interested in the knowledge related to an air.

FIG. 19

Problem Folder: how to move air.

--Solution: pressure-sensitive device

Note: In order to know how to move air, the user opens the folder and obtains the solution - pressure-sensitive device.

FIG. 20

(A) Input Original text from various sources (step 28):

Document #1 (Citation & Link)

To ensure that an excessive downward movement of the air owing to the swirl will be prevented and that the air will be displaced from the interior space through the gap at the top edge of the insulating glass pane, the nozzles of the V-shaped array are closed when they have been open for a short time and then the main nozzle is re-opened so that the air which has been moved by the swirl flow out of the top corners will be displaced upwardly through the gap.

...

Document #2 (Citation & Link)

When the disk 204 is rotated by the motor 206, air moved by the top of the disk together with the structure of the slider 212 causes the slider 212 to ride on a cushion of air, referred to as an air bearing.

...

Document #3 (Citation & Link)

The invention defined in claim 1 wherein the flow of air and material moved by the rotating blade is generally non-turbulent over the surfaces.

...

(B) Output structured SAO KB (step 62):

```
Problem Folder: how to move air
    |
    --Solution 1:  swirl flow
    |
    --Solution 2:  top of the disk
    |
    --Solution 3:  rotating blade
```

FIG. 21

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/17444

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/27
US CL :704/9

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 704/1, 7, 8, 9, 10; 707/2, 3, 4, 5, 104, 530, 531, 532

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Please See Extra Sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 4,829,423 A (TENNANT et al) 09 May 1989 abstract; col. 3, line 14 to col. 4, line 26; and col. 5, line 6 to col. 27, line 55	1-6
A	US 4,864,502 A (KUCERA, et al) 05 September 1989, abstract; figs. 2-3 &16; col. 2, line 22 to col. 3, line 59; and col. 34, line 8 to 36, line 17	1-6
A	US 5,331,556 A (BLACK, JR. et al) 19 July 1994 abstract; col. 1, line 15 to col. 2, line 68; and col. 3, line 23 to col. 8, line 46	1-6
Y	US 5,369,575 A (LAMBERTI, et al) 29 November 1994 abstract; figs. 1-3; col. 1, line 12 to col. 3, line 64; and col. 3, line 66 to col. 7, line 37	1-6

☒ Further documents are listed in the continuation of Box C. □ See patent family annex.

*	Special categories of cited documents:
"A"	document defining the general state of the art which is not considered to be of particular relevance
"E"	earlier document published on or after the international filing date
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O"	document referring to an oral disclosure, use, exhibition or other means
"P"	document published prior to the international filing date but later than the priority date claimed
"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&"	document member of the same patent family

Date of the actual completion of the international search 28 JULY 2000	Date of mailing of the international search report 13 SEP 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer JOSEPH THOMA James R. Matthews Telephone No. (703) 308-3900

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5,559,940 A (HUTSON) 24 September 1996 abstract; figs. 5-9; col. 2, line 7 to col. 3, line 18; and col. 3, line 48 to col. 9, line 15	1-6
A	US 5,696,916 A (YAMAZAKI et al) 09 December 1997 abstract; and col. 4, line 21 to col. 42, line 20	1-6
A	US 5,761,497 A (HOLT et al) 02 JUNE 1998 abstract; figs. 2-3, 6-18, & 22; col. 1, line 17 to col. 2. line 40; and col. 3, line 15 to col. 13, line 20	1-6
A	US 5,799,268 A (BOGURAEV) 25 August 1998 abstract; figs. 1-3 & 5; col. 1, line 7 to col. 6, line 16; col. 6, line 66 to col. 12, line 27; col. 39, line 40 to col. 42, line 34; and col. 55, line 11 to col. 65, line 60	1-6
A	US 5,873,056 A (LIDDY et al) 16 February 1999 abstract; figs. 1 & 6-11; and col. 1, line 5 to col. 10, line 28	1-6
Y	US 5,873,076 A (BARR et al) 16 February 1999 abstract; figs. 1-4C; col. 1, line 15 to col. 3, line 54; col. 3, line 61 to col. 7, line 10; col. 8, line 50 to col. 15, line 48; col. 23, line 9 to col. 24, line 26; and col. 31, line 43 to col. 35, line 26	1-6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/17444

B. FIELDS SEARCHED
Electronic data bases consulted (Name of data base and where practicable terms used):

EAST

search terms: SAO, SVO, subject/noun, ver/action, object, query, natural language, information/database retieval/filtering,