REASON: NLP-based Search System for the WWW

Natalia Anikina, Valery Golender, Svetlana Kozhukhina, Leonid Vainer, Bernard Zagatsky

LingoSense Ltd.,
P.O.B. 23153, Jerusalem 91231, Israel
valery@actcom.co.il
http://www.lingosense.co.il/ai_symp_paper.html

Abstract

This paper presents REASON, an NLP system designed for knowledge-based information search in large digital libraries and the WWW. Unlike conventional information retrieval methods, our approach is based on using the content of natural language texts. We have elaborated a searching strategy consisting of three steps: (1) analysis of natural language documents and queries resulting in adequate understanding of their content; (2) loading the extracted information into the knowledge base and in this way forming its semantic representation; (3) query matching proper consisting in matching the content of the query against the content of the input documents with the aim of finding relevant documents. The advanced searching capabilities of REASON are provided by its main constituents: the Language Model, the Knowledge Base, the Dictionary and the Software specially developed for content understanding and content-based information search. We characterize the particularities of the System and describe the basic components of the formal linguistic model, the architecture of the knowledge base, the set-up of the Dictionary, and the main modules of the System - the module of analysis and the logical inference module. REASON is an important contribution to the development of efficient tools for information search.

Introduction

The WWW is rapidly expanding and becoming a source of valuable information. At the same time this information is poorly organized, mixed with noise and dispersed over the Internet. Attempts to find the information using currently existing search engines such as Alta Vista, Lycos, Infoseek, etc. lead to unsatisfactory results in many cases. Usually, the response to the search contains a large amount of documents, many of which are totally irrelevant to the subject of interest. On the other hand, documents that are relevant may be missing if they do not contain the exact keywords. Refining the search by using "advanced" Boolean operations usually results in a few, or even no, documents. Serious problems are caused by homonymy, ambiguity, synonymy, ellipsis, complex semantic structure of natural language.

Let us consider such relatively simple queries as:
Q1. Supercomputer market data
Q2. Digital users group in England
Q3. Parallelization tools
It is very difficult to find documents relevant to these queries using traditional keyword search. Even more sophisticated approaches based on different thesauruses do not allow matching semantic relations between different concepts in queries and texts.

In order to solve the above-mentioned search problems we designed an NLP-based search system REASON. This system understands free-text documents, stores their semantic content in the knowledge base and performs semantic matching during the processing of queries. This paper briefly describes the main principles of the System.

General Principles and System Architecture

REASON is an NLP- based information search system which is capable of:

- understanding the content of natural language documents extracted from large information collections, such as the WWW;
- understanding the content of the users' query;
- building a knowledge base in which the extracted information contained both in the documents and the query receives an adequate semantic representation;
- performing information search which consists in matching the content of the query against the content of the documents ;
- finding relevant documents or giving another kind of response (e.g. answering questions).

REASON does not use the keyword matching method. Instead, the System applies more sophisticated matching techniques which involve a number of reasoning operations (cf. Haase 1995):

- establishing equivalence of words and constructions,
- expansion of the query on the basis of associations between concepts (definitions, "general-specific" relations and so on),
- phrase transformations (based on "cause-result" connections, temporal and local relations, replacement by paradigmatically related words),
- logical inferences (based on cross-references in the text, restoration of implied words).

The main constituent parts of REASON are: the Language Model, the Dictionary, the Knowledge Base and the Software. The present version of REASON processes texts in the English language. The System software is written in C and runs on different platforms. The interface to the Web is implemented in Java.

Language Model

The needs of natural language processing call for a consistent, unambiguous and computer-oriented model of linguistic description. A natural language, unlike artificial languages, abounds in ambiguities, irregularities, redundancy, ellipsis. In constructing the language model we proceeded from current linguistic theories, in particular, semantic syntax (Tesniere 1959), generative semantics (Fillmore 1969), structural linguistics (Chomsky 1982, Apresian et al. 1989), Meaning-Text Model (Melchuk & Pertsov 1987). We also make use of the experience of the "Speech Statistics" Research Group headed by Prof. Piotrowski (Piotrowski 1988).

Our language model includes four levels: morphological, lexical, syntactic, discourse. At each level we distinguish two planes: the plane of expression (forms) and the plane of content (meanings, concepts). The latter is considered the semantic component of language.

Morphological Level

In describing English morphology we applied the technique suggested by J.Apresian. For compact presentation of morphological information we used charts of standard paradigms of morphological forms and masks for non-standard stems. Every word stem is ascribed a certain morphological class on the basis of three charts for verbs and two charts for nouns and adjectives.

We elaborated rules for revealing deep structure categories for verbs. Thus, we have learned to determine the real time and aspect meanings guided by the co-occurrence of adverbials of time (certain features are ascribed to each group of adverbs and prepositions of time), grammatical features of verbs (stative verbs, verbs of motion). This approach allows to transfer to utterances in other languages, where the expression of tense-aspect relations is different.

Our Model includes about 80 patterns which establish correspondence between form and content of verbal tense-aspect forms.

Lexical Level

All the words are divided into two kinds: major and minor classes. The former include conceptual (notional verbs and nouns) and attributive classes (e.g. adjectives and adverbs of manner, denoting the properties of concepts). Major word classes are open groups of words, while minor classes are closed groups consisting of a limited number of words. Major and minor classes are treated differently in text processing. We have singled out about 200 conceptual word classes for verbs and 220 - for nouns. Each word class is characterized by some common semantic or pragmatic feature, e.g. Word class 187 - "Sphere of human activity" (science, business), 204 - "Cessation of Possession" (lose). We have singled out about 150 attributive classes, e.g. Word class 407 - "Color" (white), 480 - "Character of action according to intention" (deliberate).

We differentiate between the names of properties (color, size, depth, price) and the value of these properties (red, big, deep, cheap). The names of properties (nouns), their values (adjectives) and units of measurement (nouns) are linked in terms of lexical functions (Melchuk & Pertsov 1987), which enables the System to paraphrase different statements and bring them to a canonical form.

Minor word classes play an auxiliary role. They include functional words, supra-phrasal classes, words expressing modality, probability, the speaker's attitude, and so on. Supra-phrasal word classes comprise words indicating reference to the information contained in other parts of the sentence (also, neither), sentence connectors and sentence organizers: e.g. "to begin with" marks the starting point of the sentence, "by the way", "on the other hand" develop the theme, "therefore" marks the conclusion, and so on. Special place is occupied by intensifiers of properties (too big, quite well), actions(liked very much), etc.

Our Linguistic Model reflects logical, extralingual connections and associations between words. Due to these relations, it becomes possible to join words into groupings on the paradigmatic axis (Miller 1990).

We distinguish several types of paradigmatic groupings:

1. Quasi-synonymic groups
2. "General-specific" groupings
3. "Whole-part" groupings
4. "Cause-result" groupings
5. Conversives
6. Definitions
7. Lexical paradigms - "Nests".

Quasi-synonymic groups include words which are close in meaning and can be interchangeable in speech or, at least, in documents pertaining to a certain domain of science or technology. As an example, we could mention such Quasi-synonymic groups as Qv3 (show, demonstrate, present, describe, expose), Qn10 (developer, author, inventor).

"General-specific" groupings. Such a grouping consists of a word with a generic meaning and words with more specific meanings. For instance, "market" is a generalized term for more specific "purchase", "sale", "orders", "supply".

"Whole-part" groupings. The name of the whole object can be used instead of the names of its components. For example, "America" can be used instead of USA, Canada; "processing" can be used instead of "analysis", "synthesis", and so on.

"Cause-result" groupings. Many concepts, especially actions and states, are connected by the "Cause-result" relation. We have revealed a number of such connections, e.g.:

"produce" → "sell" → "order"

(If a company produces something, it can sell this product and a customer can order it.)

Special rules are designed in order to reveal the role of the participants of the situation.

Conversives. Many verbs are associated on the basis of conversion. They form pairs of concepts (take - give, sell - buy, have - belong). In real speech the statements organized by these verbs are interchangeable if the first and the second (or third) actants of the predicate are reversed:

I gave him a book → He took a book from me.

They have a laboratory → A laboratory belongs to them.

The company sold us a computer → We bought a computer from the company.

There are rules stating correspondences between actants .

Definitions. Paradigmatic associations between concepts, based on the above-mentioned groupings, are most efficiently realized in definitions. Every definition states a certain equivalence of concepts. If definitions are loaded into the knowledge base, the associations between the components of the definition help identify statements expressing the same idea. We singled out a number of types of definitions:

1. Incomplete definition of the type Ns is Ng (specific is generic): DECUS is a worldwide organization.
2. Complete definition of the type Ns is Ng that (which)... : DECUS is a worldwide organization of Information Technology professionals that is interested in the products, services, and technologies of Digital Equipment Corporation and related vendors.
3. Incomplete definition of the type Attribute + Ng is Ng: A parallel computer is a supercomputer.
4. Complete definition of the type Attribute Ng is Ng that... : A parallel computer is a supercomputer that is based on parallelization of sequential programs.
5. A list of junior concepts - Ng is N1s, N2s, N3s... : Market is sale,purchase, orders, supply ...
6. N1 is the same as N2: Identifier is the same as symbol.
7. N1 is abbreviation of N2: COBOL is the abbreviation of Common Business-Oriented Language.

Nests. One of the most ingenious and useful ways of grouping words is their organization into lexical paradigms which we call "Nests". Every Nest contains words of different parts of speech, but united around one basic concept (e.g. invest, investment, make (an investment), investor, etc.). Members of the Nest may be derived from different root morphemes, e.g. buy, buyer, customer, purchase, shopping. The main criterion is that the members of the Nest stand in certain semantic relations to the head of the Nest. Every Nest is viewed as a frame and its members occupy certain slots in it. Each slot corresponds to a specific semantic relation with the head of the Nest or sometimes with other slot-fillers. The relations between the Nest members can be described in terms of lexical functions - LF (according to I.Melchuk). We have developed and adjusted the system of LF to the peculiarities of our Language Model. We designed several types of Nests. Transitive verbs of action, for example, form a frame which includes 25 slots. In each particular case some of the slots may remain empty.

Below we illustrate the Nest formed by the verb "produce":

Slot 1 - the verb of action: produce

Slot 2 - nomina actionis: production, manufacture (LF S0)

Slot 3 - perform the action of Slot 2: carry out (LF Oper1)

Slot 4 - the performer of the action: producer, manufacturer

Slot 7 - the generalized name of actant 2: product

Slot 13 - able to perform the action of Slot 1: productive (LF A0).

Semantic Features. The mechanism of semantic features (SF) plays an important role in the semantic and syntactic description of language. Within our Model every lexeme (hence, a concept) is assigned a specific set of SF which characterize their semantic content and often determine their syntactic behavior. The co-occurrence of words is often determined by their semantic agreement.

We distinguish about 200 SF for nouns, 180 SF for verbs, 50 SF for adjectives.

Some examples would suffice.

SF for nouns: "Relative" (son), "Economic organization" (company), "Inanimate" (house, book), "Transparent" (a kind of, a part of), etc.

SF for verbs: "Causative" (compel), "Possession" (own), "Movement" (go), etc.

SF for adjectives: "Measurable property" (cold), "Relation to time" (past), etc.

SF are used in government models to specify the semantic restrictions on the choice of actants. They help identify the actants and their semantic roles, resolve ambiguity, reveal second-level models commonly formed by a limited circle of predicates and so on.

Here is one example to illustrate the assertion:

The homonyms "raise" (grow plants), "raise03" (bring up), "raise01" (lift), "raise02 (gather money), "raise04" (produce an action) are differentiated only due to the SF of their first and second actants.

Syntactic Level

The basic syntactic unit is a predicative construction. The predicate governs the dependents as the head of the construction. The main form of the predicative construction is a simple sentence, but the predicate can govern, in its turn, other predicative constructions. Among dependent structures we distinguish subordinate clauses of all kinds and predicative groups formed by verbals. So it is possible to grade predicative constructions by the rank of their dependence (first, second or third levels).

At the deep structure level relations between words are viewed in terms of semantic roles, when a dependent ("a Servant") plays a definite semantic role in relation to the head ("the Master"). Thus a predicative construction is represented as a role relation (Allen 1995). In our Language Model the Syntactic Level distinguishes about 600 semantic roles of the first and second levels (including subroles).

Here are some examples of roles.

#1	Addressee
#9	Instrument
#35.1	Agent of the action
#35.2	Experiencer

##24.6.a Content of complete information at the 2nd level (that-clause), etc.

Other parts of speech can also form role relations. One of the most frequent heads of a predicative construction is a predicative adjective ("He is afraid of progress"). But we treat such adjectives as verbs with auxiliary "be", and in the course of analysis the combination "be + adjective" is reduced to a one-member verb (Lakoff 1970). Besides, many nouns can have servants linked to them by certain semantic roles. A special place among potential masters is occupied by prepositions and conjunctions.

If the servants can be predicted, the combinability of the master is described in terms of models. We have singled out about 260 patterns of government, or Government models. Every model is a frame. The slot-fillers are actants, each of which is linked to the master by a certain semantic role. In typical government models the actants are described in terms of word-classes, but in local models, characteristic of concrete predicates, the SF of actants are usually specified. The number of actants varies from 1 to 7. For example, the verb "buy" has five actants. In the following sentence all of them are realized:

"The customer bought a computer for his son from IBM for 2000 dollars".

Act 1 is linked by role #35.1 (Agent). Required SF N13 (Legal person).

Act 2 is linked by role #60.1 (Object of Action). Required SF N4 (Inanimate non-abstract thing), N3 (Animals), N87 (Services).

Act 3 is linked by role #2 (Beneficiary). Required SF N13, N3.

Act 4 is linked by role #55 (Anti-Addressee). Required SF N13.

Act 5 is linked by role #12 (Measure). Required SF N51 (Money).

The master can have servants the occurrence and selection of which is not determined by the predicate. Their positioning, structural and semantic features are less fixed than those of the actants. They are considered non-actant servants. Lexemes functioning as non-actant servants are ascribed reverse models which indicate in what relation they stand in reference to their master.

Second-level government models are also classified into two types: actant and non-actant. For instance, the verbs "tell" and "inform", besides first-level models, can form models on the second level, e.g.:

1) an object clause linked by "that" (He told us that he lived there).

2) an object clause introduced by relative conjunctive words (He told us how he worked there).

Discourse

An important component of syntactic and lexical analysis is processing textual units larger than a sentence (Allen 1995). This is the level of discourse.

One of the most salient features that characterize text is cohesion. Our System deals with different types of sentence cohesion. They include the use of substitutes, correlative elements, all types of connection signals (like word order, functional words, sentence organizers of the

"yet", "besides", "to begin with" types). Semantic interdependence between sentences involves naming identical referents (objects, actions, facts, properties) in different ways. Thus, parts of a broad context are knit by various kinds of cross-reference. For instance, pronouns and their antecedents found in the preceding parts of the passage co-refer. Our Language Model has special rules for identifying such cases of co-reference.

Among other means of co-reference we take into account the following:

a) the repetition of the same noun with different articles or other determiners;

b) the use of synonyms or other semantically close words (company - firm, parallelizer - parallelization tool);

c) the use of generic terms instead of more specific in the same context (vessel - ship, warship, freighter);

d) the use of common and proper nouns for denoting the same referent (Washington - the capital of the USA);

e) the use of semantically vague terms instead of more concrete denominations (company - Cray Research);

f) the occurrence of words with relative meanings (sister, murderer, friend), the full understanding of which is possible only if discourse analysis helps to find their actual relationships (say, whose sister, the muderer of whom, a friend to whom).

The semantic representation of a sentence is often impossible without taking into consideration the representation of other parts of the context. In particular, it appears necessary to consider the syntactic structure and role relations of preceding sentences. Special rules help to restore ellipsis, incomplete role relations with missing obligatory or essentially important actants (key actants), to identify semantically close syntactic structures. Proper discourse analysis is an important prerequisite for natural text comprehension and information search.

Dictionary

The main components of the Language Model are mirrored in the Dictionary entries. Our dictionary currently consists of about 20,000 items. Further expansion of the Dictionary can be automated. New lexemes can be added by analogy if their semantic and morphological characteristics coincide with those of the previously introduced entries. An appropriate mechanism for expanding the Dictionary has been elaborated and tested.

Every entry contains exhaustive lexicographical information about the word: its semantic, morphological, syntactic, collocational, phraseological, paradigmatic characteristics. Each entry has 20 fields. Here is part of the entry "sell":

f1. The stem: sell

f2. The code of the word-class: 204

f3. The code of the word within the word-class: 7

f4. The code of the typical government model: 4.3.1

f5. Morphological class: 42

f7. Syntactic features: G1, G9, G18

f8. Semantic features: V6, V86, V19, V53

f9. Grammatical features: Gl2; Tv3

f15. Lexical functions: Lf60 = "sell$"

The verb "sell" has two stems: "sell" and "sold"; both of them share the same fields - 2 and 3 and constitute one lexeme. Some other fields may differ. Thus, in the entry "sold" we see:

f5. 48

f9. Gl2; Tv3; Pass1

In the entry "inform" we see:

f10. Paradigmatic connections: Qv13; Lv23

In the entry "abolish" we see:

f13. Style: St3, St4

For the verb "delete":

f14. Domain of usage: Cf 45

Some comments on the meaning of the symbols in the fields are presented below:

G1: possible to use absolutely, without any objects;

G18: possible to use with a direct object;

Gl2: used with prepositionless direct object;

G9: used only with adverbials of place, and not those of direction;

Tv3: verb of a dual character (can be either resultative or durative);

Lf60: "sell$: the verb forms a Nest of type 1;

Pass1: the stem forms Passive of type 1, when actant 2 becomes the subject (the left-hand dependent);

Qv13: the lexeme belongs to a quasi-synonymic group No13 (alongside with "say", "report", etc);

Lv23: the verb can govern a second-level model of a certain type;

St3: literary style;

St4: style of official documents;

CF 45: the domain of usage "Computers".

Lack of space does not permit us to dwell on many other subtle word characteristics contained in the entry.

There are some more points concerning dictionary entries. Special entries are compiled for set expressions. Fixed and changeable set expressions are processed differently (compare "of course", "by bus", on the one hand, and "vanity bag", "bring up", on the other).

The dictionary entry also contains full information about typical and local government models.

Some entries are formed not for concrete words, but for representatives of groups of words: e.g. conceptual word classes or quasi-synonymic groups.

Knowledge Base

The System's Knowledge Base (KB) serves for representing the meaning of natural language texts - both documents and queries. In our System the term Knowledge base is used in a slightly different sence than in the theory of expert systems. The KB structure is specially oriented towards extracting from the text a maximum of knowledge that is necessary for further search of documents on the basis of the query.

Simple Role Relation. The basic construction of the KB is a so-called simple Role Relation (simple RR). A simple RR is a KB representation of a simple sentence in the form of a tree. The root of the tree is the predicate of the sentence. All nodes of the tree are classified into two types - 0-rank nodes which represent concept instances (objects, actions, states) and 1st-rank nodes which represent the values of the properties of these concept instances. An arc from a 0-rank parent node to a 0-rank child node represents a role the latter performs in relation to the former. An arc from a 0-rank parent node to a 1st-rank child node represents an instance-property link which is not treated as a role.

To illustrate a simple RR, let us take a natural text sentence:

S1: The Company provides customers with the newest technology.

For describing Role Relations, we use our own metalanguage in which the RR of S1 looks as follows:

provide => Company < 1act, #35.1 >, customers < 2act, #2 >, technology (newest) < 3act, #60.1>

This is interpreted in the following way:

The predicate "provide" - Master - has the following Servants:

"Company" as the 1st actant linked by role #35.1 ("Agent"),

"customer" as the 2nd actant linked by role #2 ("Beneficiary"),

"technology" as the 3rd actant linked by role #60.1 ("Object of Action");

"technology" is specified by a property having the value "new".

Characteristics of the Elements of a Simple Role Relation. Each node - an element of the RR - is represented in the KB by a set of characteristics derived from the text in the course of processing. Thus, for each concept instance, there is a KB correspondence - a 0-rank node which is an RR element having about 20 characteristics, e.g.:

- the class and the ordinal number,
- the morpho-syntactic characteristics,
- the reference to the properties of the element,
- the role of the given element in reference to the master.

Likewise, for each property of the concept instance, there is a KB correspondence - a 1st-rank node which is an RR element having about 30 characteristics, e.g.:

- the value of the property (can be expressed by a number, a word from the Dictionary or a word unknown to the Dictionary),
- the name of the property ("price"),
- measurement units associated with the property (as, for example, "dollars" for "price"),
- the intensity and degree of intensity for the property value,
- the numerical coefficient of fuzziness used for processing fuzzy values.

Complex Role Relation. Simple RRs serve as building material for three more complicated types of knowledge structures which represent any natural language sentence in the KB, that is, for System of Facts, Task, and Rule.

These three types reflect the following communicative types of utterances.

System of Facts corresponds to narrative sentences that denote facts, events, situations, etc and cannot be defined as Tasks or Rules.

Tasks refer to sentences with hortatory meaning (e.g. requests, orders, commands) that impel man or machine to perform a certain action (Directive) under certain Starting Conditions. Directive is expressed by means of the Imperative mood, different forms of modality, questions, and so on (cf. "Help me!", "I'd like you to help me", "Could you help me, please?"; "What did you see at the exhibition?" ~ "Tell me what you saw at the exhibition").

Rules reflect various types of relationships, dependencies, regularities, objective laws, etc that do not impel man or system to action. Rules are mainly expressed by "if- then", "in order to" and similar constructions.

For formal differentiation between Systems of Facts, Tasks and Rules, a number of linguistic means are involved, such as mood, tense, modality, evaluative and emotional expressions, interrogative forms, and others. Specific combinations of these factors also predetermine quite definite semantic roles characteristic of each type.

The following sentences are examples of the three types of knowledge structures respectively:

S2: Convex announced the total investment program which preserves customer investments. (System of Facts)

S3: As soon as a new issue of the "Computer Magazine" appears, find me articles on supercomputers. (Task)

S4: If a company produces any product, it evidently receives orders for this product. (Rule)

Each structure is presented in the form of a complex RR which is a tree of simple RRs.

A System of Facts has the following KB characteristics: the moment of registration; the author or source of information; the reliability coefficient of the source of information; the references to the simple RRs, each of them presenting one separate fact of the system of facts.

In sentence S2, for example, the simple RR1 represents the main clause of the complex sentence, while the simple RR2 represents its subordinate clause manifesting inter-predicate (2nd-level) role ##80.1.1 ("Defining Relative Clause"):RR1 =>RR2 <##80.1.1>.

Natural language documents to be searched are, as a rule, systems of facts.

Note: A simple sentence corresponds to a system of facts containing only one fact.

A Task includes a Directive and Starting Conditions. The System's operation, on the whole, consists in monitoring the starting conditions and, on their fulfillment,in executing the corresponding directives.

A Task has the following KB characteristics: the moment of registration; the author or source of information; the references to the RRs, each of them presenting one separate directive or starting condition of the task.

In sentence S3, for example, the simple RR1 represents the main clause of the complex sentence, while the simple RR2 represents its subordinate clause manifesting 2nd-level role #4.1.2.2.1.3.1.b ("Just after"): RR1 => RR2 < ##4.1.2.2.1.3.1.b >.

Natural language questions and queries are particular cases of Task, with starting conditions often omitted, which means "Now" by default (e.g.: Tell me about the supercomputer exhibition in London).

A Rule has the following KB characteristics: the moment of registration; the author or source of information; the reliability coefficient of the source of information; the type of the rule: inference rule, identity, cause-effect relationship, condition, etc; the probability of the antecedent-consequent relation; the references to the antecedent and consequent of the rule, each of them presented by a RR.

In sentence S4, for example, the consequent is represented by the simple RR1, while the antecedent is represented by the simple RR2 with role ##30.1.a ("Condition"):
RR1 =>RR2 < ##30.1.a >.

Thus, any natural language sentence is presented, in a general case, by a complex RR - a tree of simple RRs, where the root represents the main clause, the other nodes represent subordinate clauses, and the arcs represent the corresponding roles.

Note: We treat coordination of clauses as a special case of subordination.

Special Search-Oriented KB Constructions. Alongside with the registration of the above-mentioned knowledge constructions, the System also registers special search-oriented constructions in the KB. They are files of concepts and files of concept instances.

In the file of concepts, each concept has the following characteristics: the class of the concept; the list of the elements of RRs which include the instances of the given concept, i.e. the inverted list for the concept; the list of definitions for the concepts and the concept instances which provide a powerful tool for the search of documents.

Main Modules of the System

The System's software has a modular structure. The Modules run asynchronously, exchanging messages, which allows to parallelize their operations on a multiprocessor.

The System consists of the following main modules:

(a) the module of graphical user interface,

(b) the module of analysis which forms the KB from natural language texts,

(c) the module of generation (synthesis) which produces natural language texts from the KB,

(d) the KB-access module which provides access to the KB for analyzing Facts, Tasks and Rules,

(e) the module of logical inference which handles document searching on the basis of a set of equivalent transformations over the KB.

Modules (b) and (e) are the most complicated in the System and play a most important role in the document search.

Module of Analysis. The process of analysis consists of a number of steps, each of them providing knowledge for the final KB representation. The process begins with morphological analysis, after which semantic-syntactic parsing is performed. The parsing falls into two stages.

The first stage accesses the input text in its surface form. This stage consists of the steps which carry out, in the appropriate order, the processing of words and word groupings (set expressions, articles, personal pronouns, noun constructions (NN and the like), negations, etc) and attach 0- and 1-st rank servants to their masters - verbs, predicative adjectives, nouns. The searching of 0-rank servants is conducted according to the master's Government model (for the actant servants) and by means of special Prepositional models (for the non-actant prepositional servants).

There are special rules that limit the area of searching.

The System also makes a canonical reduction of certain constructions on the basis of equivalent transformations of different types.

The first stage results in the draft variant of parsing at the level of simple RRs.

The second stage of parsing, which deals with RRs only, produces final variants of the simple RRs, establishes 2nd-level relations (conjunctional and non-conjunctional), forms the final complex RR, computes the deep tense-aspect characteristics, and registers all the knowledge obtained in the KB.

In the process of analysis, the System detects various types of conflict situations, e.g.: a servant is linked to more than one master; a servant is ambiguous; a predicate is ambiguous,etc. To resolve the conflicts the System exploits a special inventory of rules and the mechanism of penalties and awards. If a conflict cannot be resolved, the System registers more than one variant.

It should be noted that word-sense disambiguation is carried out at various steps including morphological analysis, set expression and article processing and especially when using such powerful filters as Government and Prepositional models with their strong semantic restrictions. In addition, word-order rules and domain filters are applied.

Module of Logical Inference for Document Search. The module of logical inference, together with the KB- access module, makes all special transformations over the KB which are necessary for query matching on the basis of the KB representations of the document and the query. The technique of keyword matching is not used at all. Thus, the one-sentence document "The developers of this supercomputer market a variety of electronic devices" will be recognized by our System as non-relevant to the query "Supercomputer market".

The module possesses the full inventory of the System's search techniques, bringing them into play when necessary. The use of these techniques may be both sequential and parallel.

The search process consists of two main stages:

Stage 1. The expansion of the input query (i.e. its KB representation) by use of quasi-synonyms, definitions, "General-specific" relations and conceptual Nest relations (cf. Vorhees 1994).

The expansion of the query enables the System to take into account not only the explicit occurrences of the query elements in the RRs, but also the implicit, expanded ones. Due to the query expansion, the document D1, for example, will be recognized as relevant to the query Q1, given the set of definitions E1:

E1: Sale is a particular case of market.
An Exemplar system is a supercomputer.

D1: The sales of high-performance Exemplar systems increased threefold in the last 3 years, to $6 billion in 1995.

Q1: Supercomputer market.

Stage 2. The use of the rules of logical inference. They are put into effect if stage 1 is not sufficient for making the final decision. The logical inference rules allow to match the query with separate pieces of knowledge which are often scattered over the document.

Some techniques of this stage can be illustrated by the following search microsituation, where E2 presents "encyclopedic" world knowledge (definitions and other natural language statements entered into the KB beforehand), D2 is the document, Q2 is the query.

E2: Cray produces supercomputers.
Sale is a particular case of market.

D2: Cray products are widely used for high-performance computing. In the last quarter, the sales totaled $266 million.

Q2: Supercomputer market.

The elements of the query (i.e. its RR) are checked for their co-occurrence (explicit or implicit) in one of the document's sentences (i.e. in the corresponding RR). (It should be noted that here we mean such co-occurrence of the query elements that preserves the structure of the query Role Relation).

There aren't such sentences in D2, so the System selects the sentence containing only the predicate of the query - "market" (rendered by its implicit occurence "sale"). Then the previous sentences of D2 are searched through for finding the object of "sale". For this, special object-searching rules are applied. The object is found - it is "products".

"Products" is not the second element of the query ("supercomputer"), but it belongs to the group of the so-called non-terminal words which suggest further reasoning. For the non-terminal words, the conceptual Nest rules are attached which activate the conceptual Nest relations of "products". This leads to generating a new inner subquery in the form of the sentence "Cray produces X1", and, through the use of the encyclopedic fact "Cray produces supercomputers", results in recognizing the relevance of D2 to the query Q2.

Besides the rules mentioned, the System uses a wide range of knowledge-based search techniques, such as

- various inference rules based on different paradigmatic word groupings,
- mechanisms for searching antecedents, including not only antecedents for pronouns, words of vague meaning, etc., but also for text lacunas,
- local and temporal transformations,
- "cause-result" transfer, etc.

These techniques allow, in particular, to realize query matching for the following search microsituation:

E3: Order is a particular case of market.
Cray sells supercomputers.
France is in Europe.

D3: Cray Reports the Results of the Fourth Quarter and the Full Year. In France, orders totaled $66 million in the quarter compared with $47 million for the third quarter of 1995.

Q3: Supercomputer market in Europe.

The query matching process uses, among others, the inference rules

"produce" → "sell" → "order"

described in the Language Model section.

Conclusion

At the present stage our System handles documents containing commonly used, non-specialized vocabulary. Besides, we process documents belonging to computer science and related domains. Further expansion of the Dictionary will enable the System to process documents related to other domains. Experiments with the System show its efficiency in refining results of the search performed by traditional search engines on the WWW. The Web-resident version of the System is currently under development. REASON possesses great potentialities for various practical applications involving man-machine communication and artificial intelligence (e.g. expert systems, machine translation, document summarization and filtering, information extraction, educational software, control of industrial processes).

References

Allen, J. 1995. Natural Language Understanding. USA/Canada: The Benjamin/Cummings Publishing Company, Inc.

Apresian, J.; Boguslavski, I.; Iomdin, L.; Lazurski, A.; Pertsov, N.; Sannikov, V.; and Zinman, L. 1989. Linguistic Support of ETAP-2. Moscow: Nauka.

Chomsky, N. 1982. Some Concepts and Consequences of the Theory of Government and Binding. Cambridge: MIT Press.

Fillmore, Ch. J. 1969. Toward a Modern Theory of Case. *Modern Studies in English*. N.Y.: Prentice Hall: 361-375.

Haase, K. B. 1995. Mapping Texts for Information Extraction. *Proceedings of SIGIR '95*.

Lakoff, G. 1970. Irregularity in Syntax. N.Y.: Holt.

Melchuk, I.; and Pertsov, N. 1987. Surface Syntax of English. Amsterdam/ Philadelphia.

Miller, G. 1990. Wordnet: An On-Line Lexical Database. *International Journal of Lexicography* 3(4).

Piotrowski, P. 1986. Text Processing in the Leningrad Research Group "Speech Statistics" - Theory, Results, Outlook. *Literary and Linguistic Computing: Journal of ALLC* 1(1).

Tesniere, L. 1959. Elements de syntaxe structurale. Paris.

Vorhees, E. M. 1994. Query Expansion Using Lexical-Semantic Relations. *Proceedings of SIGIR '94*.