Information Processing & Management Vol. 27, Nos. 2/3, pp. 219-227, 1991
Printed in Great Britain.

0306-4573/91 $3.00 + .00

CONCEPTUAL PROCESSING OF TESTS IN PALAEOGRAPHY

G. CAUTIERO*, M.I. SESSA*, M. SESSA**, and M. VACCA*
*Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi (SA), Italy
**Sovrintendenza Archivistica per la Campania, Napoli, Italy

(*Received* 15 *March* 1990; *accepted in final form* 8 *October* 1990)

Abstract—The design and the implementation of a system supporting Palaeographic analysis of documents is presented. The proposed approach relies on Conceptual Processing methodologies. Some experimental results are also provided.

1. INTRODUCTION

In this paper the first release of a research project carried out in collaboration with the "Archivist Superintendence for Campania" of the Italian Ministry for Cultural Property is presented [1,2]. The aim of this project is to design and realize suitable automatic tools to support the Palaeographic analysis of XII century Southern Italian documents.

Many factors make such analysis very difficult and give rise to the usefulness of the informatic tools application. Indeed, the Palaeographic study is generally founded more on palaeographer experience than on a systematic technique application.

Transcription is the main problem in the Palaeographic analysis. It concerns the decoding of shortened forms of written words used by notaries; the lack of complete and general rules makes this task very difficult.

Our approach relies on Conceptual Processing methodologies that can be useful to design systems that embody and process knowledge the human expert uses to solve real existing problems. Broadly speaking, conceptual processing of text is the translation process from a natural language text into a given scheme for representing knowledge.

It is worth noticing that our project is among the first attempts to solve a real existing problem by means of techniques developed in the Natural Language area, and it allows some evaluations about the possible applications of such a theory.

In Sections 2 and 3 a short account of Palaeographic analysis and of Conceptual Dependency applications is given; in Section 4 our CD-based approach to the Palaeographic analysis problem is described.

2. THE PALAEOGRAPHIC ANALYSIS

In Palaeography, the products of the writing process (books, documents, inscriptions, etc.) are analyzed in order to study the writing history. Moreover, such analysis makes the texts readable, and then it provides essential preprocessing to recover written information.

Transcription concerns the decoding of shortened forms of written words used by notaries for economy and usage matter. A shortened form is the removal of some letters in the word, and it is marked by the writing notary with a sign (*signum*). Such a *signum* can be a full stop or an overwritten short dash (general *signum*-shortened form), or else a special symbol containing information about the removed letters (particular *signum*-shortened form). In what follows, the underlined words denote shortened forms.

Because the last ones can be decoded by analyzing their form, special purpose tools are needed. Therefore, we will deal only with the general *signum*-shortened forms that can be divided into two classes. There is a cutting if the removed letters are the last ones (for

This work was supported in part by the Italian Ministry of Education, Project: "Algoritmi e Sistemi di Calcolo."

instance: *dom* = *domini*), whereas there is a contraction if the removed letters are in the middle of the word (for instance: *dni* = *domini*).

Generally, there is a many-to-many correspondence between shortened forms and words (i.e., a unique shortend form can represent different words and different shortened forms can be used as abbreviations of a unique word). For example, the shortend form *fi*, in different contexts, can represent the word *filius* or *finis*, while both *sep* and *septrio* can represent the word *septemtrionis*.

In order to decode such shortened forms some general rules can be applied, but many changes occur depending on the place, the time, and the notary related to the document. For instance, the following two documents show different shortened forms and grammatical rules used by the notaries (the deleted letters are displayed in parentheses):

1. *In no(mine) Domini. Vicesimo quarto an(no) princ(ipatus) d(o)m(n)i Gisulfi glo(riosi) princ(ipis), men(se) septe(m)bri quarta indic(tione).* [3 (Salerno, 1096)].
2. *In n(omine) Domini. Vicesimo quinto anno prin(ci)b(atus) d(o)m(n)i Gisulfi glorioso prin(cebs), m(ensi)s agustus quinta ind(ictione).* [3 (Cilento, 1066)].

The corruption of the Latin language (characterizing such age) that discourages an exclusive syntactic approach can also be noticed.

Moreover, the written witnesses are themselves very different; then, in order to realize a feasible and efficient automatic analysis, it is mandatory to consider texts coming from the same region (production center) and written in a fixed time. Indeed, such elements have a great influence on the text feature.

We restrict our attention to the private documents coming from the South of Italy and written during the XII Century. A private document states the matching of intentions between members of the public, and generally is drawn up by a notary. The main features of a document are: its written form, its legal subject-matter, and its formal drawing up. Moreover, in any document there are always the two parties involved in the legal transaction (sale, acquisition, deed of gift, etc.) and the writer. In many cases, there is also a judge giving impressiveness and legality to the document itself.

A recurrent structure dividing the document into three sections can be singled out. The first one (*Protocollo*) contains the invocation to God and the date. The second one (*text*) is the more complex and larger section containing: the names of the parties (*Intitulatio*); the exposition of the transaction; the sanctions if someone does not comply with the agreements fixed in the exposition (*Sanctio*); the request, made by the parties, of drawing up the deed (*Rogatio*). The last section (*Escatocollo*) contains the signatures of the parties, the notary, and the judge if present.

It is worth noticing that the second section has variable contents and form whereas the first and the last ones show structured and recurrent sentences.

3. THE CONCEPTUAL DEPENDENCY (CD) THEORY APPLICATIONS

The Conceptual Dependency Theory [2,4–10] concerns the representation problem of the sentence meaning. The basic axiom is: Two sentences with identical meaning (regardless of the language) have the same conceptual representation. A very important corollary of the above axiom is: Any implicit information in a sentence must be explicit in the representation of the meaning of the sentence. This corollary is important because it influences the design (and hence the behavior) of CD based systems, and it is a peculiarity of the theory.

The meaning propositions underlying language are called *conceptualizations*. Broadly speaking, in CD theory a conceptualization is a network of concepts and conceptual relations, where concepts are defined by means of six conceptual categories (or types) and conceptual relations belonging to a predefined set of conceptual syntax rules.

MARGIE [4] (Meaning Analysis, Response Generation and Inference on English) was the first CD based system. It understands simple English sentences and also embodies the Riesbeck theory of free-form inferences [4]. CD provides, hence, a representation of the

sentence meaning, but this representation is not enough to represent the meaning of a text. Indeed, it is more than the sum of its sentence meanings.

In order to connect the text contents with the causality relation, Schank and Abelson introduced high level structures, called *scripts*[5–6], useful to represent a predetermined and stereotyped sequence of events that define a well-known situation, such as eating in a restaurant. The introduction of these structures allows one to limit the inferences explosion phenomenon.

SAM [5] (Script Applier Mechanism) is a program that understands a script-based story. It points out different problems related to the use of scripts in the story understanding process. Other high level CD-based structures, such as *plans* and *goals* [5], allow one to overcome some of the above problems. A plan represents the set of possible choices that an actor (in a story) can perform to achieve a given goal. PAM [5] (Plan Applier Mechanism) is the first plan-based understanding system.

The theory of Dynamic Memory [7,8,11,12] is the last trend in the CD area. A Dynamic Memory is a way to organize knowledge represented by MOPs (Memory Organization Packets) structures [11,12]. It must be able to reorganize itself when a new input text arises.

Even though many systems are MOPs-based (IPP [13-14], BORIS [15], CYRUS [16–17], ATLAST [18], DMAP0 [19], REASERCHER [20], MOPTRANST [21], the original version of MOPs structures has never been implemented. Dynamic Memories are a first step toward the realization of understanding and learning text systems, that is, toward actually robust text-processing systems.

4. PALEO: A CD BASED SYSTEM FOR PALAEOGRAPHIC ANALYSIS

We would like to specify that in our approach semantic association is the first and the most important step toward the complete resolution of the decoding problem. For example, we associate the concept of "child," relation among two humans, to the shortened form "*fi*" (of the word "*filius*"). Henceforth, in the sequel, as decoding we mean semantic decoding.

The semantic and syntactic features of the document are the two main knowledge sources involved in our approach.

The contents allow us to restrict the total number of necessary words for processing our texts. Indeed, existing lexicons show that, for example, about 1400 words are necessary to understand 137 documents [3].

In order to obtain the more detailed scheme of document content shown in Fig. 1, juridical notions about indentures have been considered [22]. The content of a single indenture is represented by the path from the start node to a terminal node.

The occurrences in a single node of the network are classified in: **events, descriptions,** and **legal formulas**.

The events, further on classified in enabling Conditions and Actions, represent states or actions involving people entering into the contract. Descriptions are attributes belonging to an a priori fixed set. Legal formulas are standard sequences of words. As regards legal formulas, the decoding problem can be efficiently solved by means of pattern matching techniques, so that they are not conceptualized. Nevertheless, formulas give information about the document structure.

Syntax knowledge is fundamental too. It is possible to restrict the syntactical variability by considering the syntactical forms notaries used to express juridical contents. The syntactical structures of our documents have been carefully analyzed and are represented in a procedural way.

4.1 *Knowledge representation*

Among many examined knowledge representation techniques [4,5,7–10,23–29], Conceptual Dependency seems the one closest to our requirements. Nevertheless, CD has, for our goals, some negative aspects. In particular, less attention has been paid to the state representation than to the action representation problem.

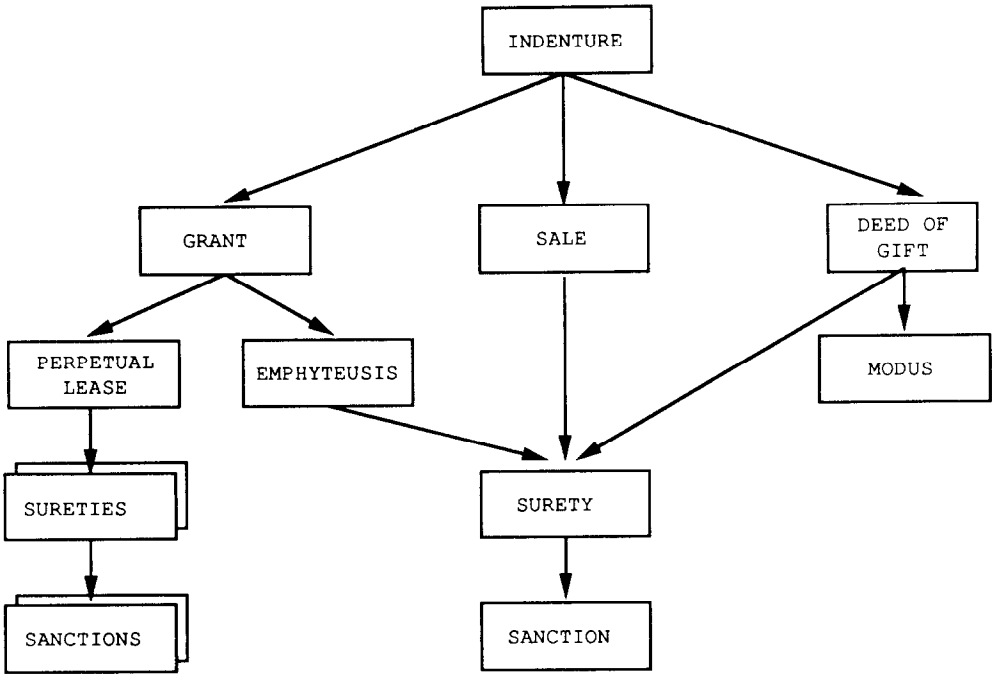

Fig. 1. The indenture content scheme.

In order to overcome such problems and to fit the present application more precisely, we developed a CD-based knowledge representation technique.

EVENTS: Events are conceptual structures involving concepts. In our problem, event representation must meet the following requirements: to be language-free, in order to overcome the morphosyntactic variabilities; to be close enough to the words, in order to simplify the shortened form decoding by means of the translation text-representation.

We define a concept as an ordered couple (t, w), where t is a type and w is a word of medieval Latin language having type t. For example, the couple (MALE *Rottelgrimus*) is a concept, while (PERSON *pecia de terra*) is not, because *Rottelgrimus* is a male name and *pecia de terra* means "a patch of land."

Further modifications are:

- In CD theory any element in the conceptual category PA is in the form: state(value), where value is a number in a fixed range. We define a PA element as an ordered couple of concepts, the first one representing the state, the second one the value.
- Introduction of subcategories. Each CD category is the root of a semantic tree of subcategories having only ISA-link [9]. The introduction of subcategories allows us to define the contents in a more detailed way and it is also useful in decoding.

For example, the event described by the sentence *ipsi germani Petrus et Guaiferius susceperunt ab eodem Desiderio auri tarenos bonos sexaginta* has the conceptualization shown in Fig. 2.

DESCRIPTIONS: Information describing a person (or object) is variable, but it always belongs to an a priori fixed set. We represent such information by means of frame-like structures we called D.structures, filled by state conceptualizations. For example, in Fig. 3 a simplified version of D.MALE structure is shown (the * is a special symbol indicating a generic word).

Such structures accomplish a double task: to define descriptives context and to store information about people or objects described in the already examined part of text (memory of people or objects).

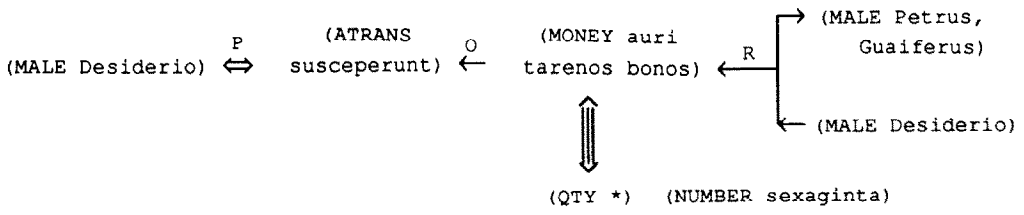

Fig. 2. An example of sentence conceptualization.

NETWORK REPRESENTATION: It is easily seen that many problems arise using structures as *scripts* to represent the whole network: proliferation, difficult instantiation, and memory employment. *Scripts* are, instead, suitable to represent only one type of document.

The structures we propose, even though *script*-like, allow us to overcome the above problems. These structures, which we called S.structures, are constituted by:

- **Roles and Props** to define people (or objects) involved in events of a given node. The definition is made by using D.structures. It is worth observing that unlike *script* theory, our roles and props, must be explicitly mentioned in the currently processing text.
- **Conditions** are state conceptualizations that define enabling conditions of a given juridical action. In documents, conditions do not always precede the actions.
- **Actions** represent juridical actions that people stipulating a contract must perform.
- **Spec.link** and **F.link** (Specification link and Free link) to link the nodes of network. We introduced two kinds of link: specialization and free links. These links are activated when an action (in a given node) holds.

For example, the node 1 (indenture) in Fig. 1 is represented by the S.indenture structure in Fig. 4.

Operations that can be made on the S.structures are:

- Make them active and not active.
- Select them.

To make an S.structure active means to fix the context of the current processing part of the text. Therefore, it is not possible to have more than one S.structure simultaneously active. Indeed, an S.structure must be made not active when it is no longer representative of the current processing part of the text.

To select S.structures means to predict the possible contexts of the part that must be processed immediately afterward. Then, the select S.structures are the candidates to be activated.

In order to define how to perform such operations, two more definitions are needed. A conceptualization, belonging to an S.structure, "holds" if it is similar to the conceptualization of the current processing part of the text. Defining the conceptualization similar-

```
(D.MALE    ((MALE *)    <≡>  (CHILD *) ((PERSON *)),
            (MALE *)    <≡>  (BROTHER-IN-LAW *) ((PERSON *)),
            (MALE (*))  <≡>  (BROTHER *) (),
            (MALE *)    <≡>  (CHARACTERISTIC *) ((TITOLO *)),
            (MALE *)    <≡>  (CHARACTERISTIC *) ((PROFESSION *))))
```

Fig. 3. D.MALE structure.

S.INDENTURE

Roles:
(P1, (PERSON *), (P1 ⟨≡⟩ D.person))
(P2, (PERSON *), (P2 ⟨≡⟩ D.person))
(P3, (PERSON *), (P3 ⟨≡⟩ (CHARACTERISTIC *) (JUDGE *)))

Props:
(0, (GROUND *), (0 ⟨≡⟩ D.ground))

Conditions:
(P1, P2) ⟨≡⟩ (BEFORE *) (P3)
P1 ⟨≡⟩ (OWNER *) (0)
P1 ⟨≡⟩ (FREE-WILL *) ()
P1 ⟨≡⟩ (EXPEDIENCE *) ()
0 ⟨≡⟩ (INTEGRITY *) ()

Actions:
P1 ⇔ (TRANS *) $\overset{o}{\leftarrow}$ 0 $\overset{R}{\leftarrow}$ → P2, ← P1

Spec_link:
(Spec_tipo: (TRANS *) (SALE *) S.sale)
(Spec.tipo: (TRANS *) (GRANT *) S.grant, S.sale)
(Spec_tipo: (TRANS *) (GIFT *) S.deed_of_gift)

Fig. 4. S.INDENTURE structure.

ity we follow the Gentner's principle [11,12]. Moreover, an S.structure is "instanced" if, in the current processing part of the text its actions hold. It is "completely instanced" if its conceptualizations hold.

A node is made active when one of its conceptualizations holds. Otherwise it is made not active. Starting from the active node, the selected nodes are chosen according to the following rule: If the active node is instanced, then select all nodes linked to it by Free-links and by Spec-links whose conditions are satisfied. In our system the process starts with the root as active node and an empty set of selected nodes.

4.2 *The processing model*

The model we propose to process palaeographic texts is a two-step model. In the first step, hypotheses about the associate concept to the shortened form are formulated. The second one is a checking step where we are looking for contradictions between formulated hypothesis and information related to the already examined part of text; when a contradiction arises, it needs to reject the current hypothesis and to formulate a new one.

In order to state a decoding hypothesis, it is mandatory to take into account many kinds of information: lexical typology, syntactic rules, knowledge of juridical contents, memory of already examined part of text, and memory of people and objects among others.

4.2.1 *Modules and their connections.* Our implementation is grounded on the design of several modules strictly related to the different factors involved in the decoding process.

Lexical information is obtained by using a dictionary and a reading module whose task is to associate to each word (resp. shortened form) a concept (resp. a set of concepts). All words in the dictionary are partitioned according to types and concepts, and we associated

to each subset the minimum possible type. For example, the type MALE is associated to the set {*Adelbertus, Marius, Rottelgrimus*, . . . }.

A conceptual parser was designed to realize the translation sequence of concepts into conceptualizations. The parser design is founded on the following rules: There are key concepts (all those having as their own types a subtype of LOC, TIME, ACT, or STATE). When a key concept is found in the text, a new conceptual structure is selected and hence available to be filled; moreover, a conceptual structure is processed until a new key concept is found or until it is completely filled. Finally, a procedure filling conceptual structures provides the representation of syntactic rules.

Processing of juridical knowledge is accomplished by two modules: Network Management Module (NMM) and Description Management Module (DMM). Such modules also play a role in the checking step.

The NMM performs two fundamental tasks: The first one is to predict the right context (i.e., the node that likely represents the contents of currently processing text). Prediction is done on the basis of the conceptualizations yielded by parser and via F.link and Spec.link. The second one is to store conceptualizations associated to the text in the right node and hence to manage memory of the already examined part of text.

DMM was designed to activate descriptive contexts (D.structures). The activation is made by using concepts; for example, the concept (MALE *Desiderio*), associated to the word *Desiderio* in the processing text, allows the activation of the D.MALE structure (Fig. 3).

The Control Module, by using information obtained by the previous modules, provides decoding hypotheses for a given shortened form. The decoding algorithm relies on the main feature of the considered documents (i.e., their standard contents). It allows one to predict the likely future contents on the basis of the past ones. Then, the knowledge of the context of the shortened form makes a feasible decoding possible.

4.2.2 *The Current Processing State of the text.* In Section 4 the content of the document has been represented (Fig. 1) as a path from the starting node to one of the terminal nodes. At any processing time, the part of the previous path, from the starting node to the currently active node, is filled and the likely future contents are known.

The information we use in order to formulate a decoding hypothesis, are synthesized in the Current Processing State of the text, which is characterized by means of the following parameters:

- **CAIC**: currently active juridical context (currently active S.structure);
- **CADC**: currently active descriptive context (currently active D.structure);
- **DC**:descriptive conceptualization (State Conceptualizations);
- **JC**: juridical conceptualization (Conditions; Actions);
- **EXPECTATION**: the set of possible concepts that the parser expects to complete the analysis of the current sentence (i.e., to fill completely the current active conceptualization).

In this way a stratified view of the contents of the document is possible, as shown in Fig. 5. Any level gives a prediction degree that increases with the level order. The first two levels give predictions about S.structures and D.structures related to the processing part of the text. The third level gives predictions about conceptual structures related to phrases. The last level provides concepts related to words.

4.2.3 *The processing.* The output of our system is the conceptual representation (S.structure, D.structure, Conceptualizations, and Concepts) of the input text. When a shortened form is found in the processing text, the Control Module provides the decoding hypotheses. It is essentially a choice of a concept in a subset of all the possible ones related to the shortened form (via dictionary). In this subset there are specializations [10] of the elements belonging to the Current Processing State. If the subset is empty, the process restarts from the last multi-choice decoding step and explores another possible choice. When a different choice does not exist, the process stops. This means either that the dictionary is not complete or that a mistake arose in the conceptualization process. If the subset has

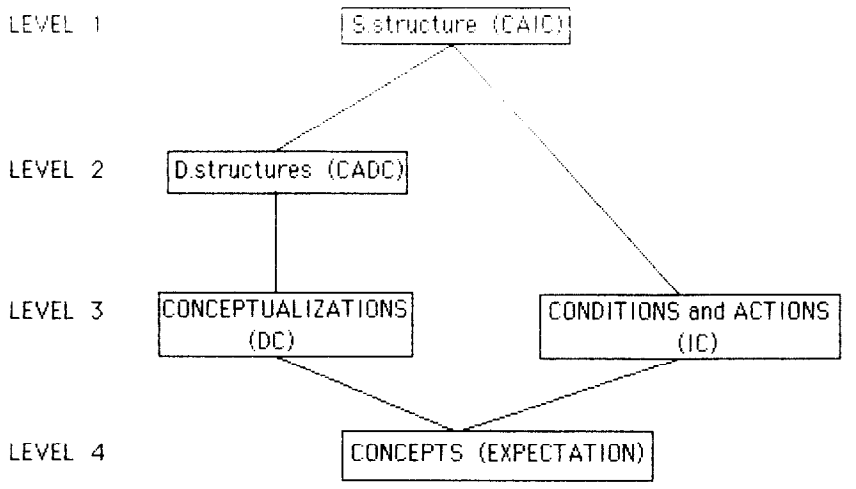

Fig. 5. A stratified view of the contents of the document.

more than one element, the choice is made with respect to priorities fixed according to the prediction degree previously explained. The case of more than one concept with the same priority is solved by means of a "set coercion" operation [10].

The program is written in Franz Lisp language under Unix and it runs on a DEC Vax 785 at University of Salerno. The program is a first release of the final system. The juridical knowledge is related to a simplified purchase deed, so that few syntax forms and a limited dictionary have been taken into account, Twenty-five sample texts and almost all the *Intitulatio* in [3] (about 100) have been processed. The results have been considered very satisfactory by the Palaeographer. No comparative evaluation is possible because it is the first attempt in this field.

The following is an example of the semantic decoding made by the program. A simple Italian language generator was implemented to make the output readable, but for a better understanding, an English translation is given.

INPUT [3 doc. 39]: *ROTTELGRIMUS FI QUONDAM SERGII ET GEMMA FI ROMOLDI QUE EST UXOR IPSI ROTTELGRIMI CONIUNCTI SUNT CUM DESIDERIO COMIT FI QUONDAM PETRI. IPSI ROTTELGRIMUS ET GEMMA DIXERUN SE HABERE TERRAM CUM ARBUSTO: AB OCCI FI VIA ET PASS DECEM; AB SEPTETRIO FINIS SEP ET PASSUS DECEM; AB MERIDIE ET PASSUS DECEM; AB ORIEN FINIS VIA ET MENSURATI PASSUS DECEM. IPSI ROTTELGRIMUS ET GEMMA VENUDEDERUNT IPSO DESIDERIO IPSAM TERRAM. IPSI ROTTELGRIMUS ET GEMMA SUSCEPERUNT AB IPSO DESIDERIO MON DECEM.*

OUTPUT: *Rottelgrimo* is son of *Sergio*. *Sergio* is dead. *Gemma* is the daughter of *Romualdo*. *Gemma* is the wife of *Rottelgrimo*. *Gemma* and *Rottelgrimo* have come together with *Desiderio*. *Desiderio* is an Earl. *Desiderio* is son of *Pietro*. *Pietro* is dead. *Rottelgrimo* and *Gemma* own a land. In the land there is a tree. Boundary lines are the road at west and the hedge at nord where the land measures ten yards. Toward sud the land measures ten yards. Toward east the land measures ten yards and has as border the road. *Rottelgrimo* and *Gemma* sold the land to *Desiderio*. *Rottelgrimo* and *Gemma* received ten coins from *Desiderio*.

REFERENCES

1. Cautiero, G. Una soluzione per il problema dello scioglimento di testi paleografici fondata sulla teoria della dipendenza concettuale, degree thesis, University of Salerno; 1989.
2. Vacca, M. Applicazioni della teoria della dipendenza concettuale al problema della comprensione del linguaggio naturale, degree thesis, University of Salerno; 1989.
3. Leone, S. and Vitolo, G., editors. Codex Diplomaticus Cavensis. Cava dei Tirreni, Italy; 1986.
4. Schank R.C., editor. Conceptual Information Processing. Amsterdam: North-Holland; 1975.

5. Schank, R.C.; Abelson, R.P. Scripts, Plans, Goals and Understanding. Hillsdale, NJ: Lawrence Erlbaum; 1977.
6. Schank, R.C.; De Jong, G. Purposive understanding. In J.E. Hayes, D. Michie, and L.I. Mikulich, editors. Machine Intelligence, 9. New York: Ellis Horwood Limited; 1979.
7. Schank, R.C. Language and memory. Cognitive Science, 4, 243–284; 1980.
8. Schank, R.C. Dynamic Memory. A Theory of Reminding and Learning in Computer and People. New York: Cambridge Univ. Press; 1982.
9. Scragg, G. Semantic net as memory models. In E. Charniak and Y. Wilks, editors. Computational Semantics. Amsterdam: North-Holland; 1976.
10. Sowa, J.F. Conceptual Structures: Information Processing in Mind and Machine. Reading, MA: Addison Wesley; 1983.
11. Gentner, D. Structure-mapping: A theorical framework for analogy, Cognitive Science 7, 155–170; 1983.
12. Gentner, D.; Toupin, C. Systematicity and surface similarity in the development of analogy, Cognitive Science 10, 277–300; 1986.
13. Lebowitz, M. Memory-based parsing. Artificial Intelligence, 21, 363–404; 1983.
14. Lebowitz, M. Generalization from natural language text. Cognitive Science, 7, 1–40; 1983.
15. Lehnert, W.G.; Dyer, N.G.; Johnson, P.N.; Yang, C.J.; Harley, S. Boris–An experiment in in-depth understanding of narratives. Artificial Intelligence 20, 15–62; 1982.
16. Kolodner, J.L. Maintaining organization in a dynamic longterm memory. Cognitive Science, 7, 243–280; 1983.
17. Kolodner, J.L. Reconstructive memory: A computer model. Cognitive Science, 7, 281–328; 1983.
18. Granger, R.H.; Eiselt, K.P.; Holbrook, J.K. Parsing with Parallelism: A spreading-activation model of inference processing during text understanding. In: J.L. Kolodner and C.K. Riesbeck, editors. Experience, Memory and Reasoning, Hillsdale, N.J.: Lawrence Erlbaum; 1986.
19. Riesbeck, C.K.; Martin, C.E. Direct memory access parsing. In J.L. Kolodner and C.K. Riesbeck, editors. Experience, Memory, and Reasoning. Hillsdale, N.J.: Lawrence Erlbaum; 1986.
20. Rich, E. Artificial Intelligence. New York: McGraw Hill; 1983.
21. Lytinen, S.L. A more general approach to word disambiguation. In J.L. Kolodner and C.K. Riesbeck, editors. Experience, Memory and Reasoning. Hillsdale, N.J.: Lawrence Erlbaum; 1986.
22. Rescigno, P., editor. Trattato di Diritto Privato, Vol. 10, Tomo secondo, Obbligazioni e contratti, UTET Torino; 1982.
23. Bertram, B. Case systems for natural language, Artificial Intelligence 6, 327–360; 1975.
24. Charniak, E. A common representation for problem solving and language comprehension information, Artificial Intelligence, 16, 225–255; 1981.
25. Charniak, E. The case-slot identity theory, Cognitive Science 5, 285–292; 1981.
26. Charniak, E. Passing-markers: A theory of contextual influence in language comprehension, Cognitive Science 7, 171–190; 1983.
27. Lehnert, W.G. Plot units and narrative summarization. Cognitive Science, 4, 293–331; 1981.
28. Lebowitz, M. Using memory in text understanding. In J.L. Kolodner and C.K. Riesbeck, editors. Experience, Memory and Reasoning. Hillsdale, N.J.: Lawrence Erlbaum; 1986.
29. Wilensky, R. Knowledge representation–A critique and a proposal. In J.L. Kolodner and C.K. Riesbeck, editors. Experience, Memory and Reasoning. Hillsdale, N.J.: Lawrence Erlbaum; 1986.