Legal Knowledge Based Systems

JURIX'96

Foundations of legal knowledge systems

The Foundation for Legal Knowledge Systems

Editors:
R.W. van Kralingen
H.J. van den Herik
J.E.J. Prins
M. Sergot
J. Zeleznikow

Paper version: Tilburg University Press, ISBN 90-361-9657-4, NUGI 699

CONTENTS

SALOMON: AUTOMATIC ABSTRACTING OF LEGAL CASES FOR EFFECTIVE ACCESS TO COURT DECISIONS

Caroline Uyttendaele, Marie-Francine Moens, Jos Dumortier
Interdisciplinary Centre for Law and IT (ICRI),
K.U. Leuven, Tiensestraat 41, B-3000 Belgium
email: {caroline.uyttendaele, marie-france.moens, jos.dumortier}@law.kuleuven.ac.be

Abstract

The SALOMON project is a contribution to the automatic processing of legal texts. Its aim is to automatically summarise Belgian criminal cases in order to improve access to the large number of existing and future cases. Therefore, techniques are developed for identifying and extracting relevant information from the cases. A broader application of these techniques could considerably simplify the work of the legal profession.

A double methodology was used when developing SALOMON: the cases are processed by employing additional knowledge to interpret structural patterns and features on the one hand and by way of occurrence statistics of index terms on the other. As a result, SALOMON performs an initial categorisation and structuring of the cases and subsequently extracts the most relevant text units of the alleged offences and of the opinion of the court. The SALOMON techniques do not themselves solve any legal questions, but they do guide the user effectively towards relevant texts.

1 Introduction

The SALOMON project is a *contribution to the automatic processing of legal texual information*. SALOMON stands for "Summary and Analysis of Legal texts fOr Managing On-line Needs". As a research project, SALOMON is navigating between at least three scientific domains: legal informatics, legal linguistics and information retrieval. The project is realised by an interdisciplinary team, composed of a computer scientist, a computer linguist and a lawyer.

1.1 Purposes

The main purpose of the project is to test and develop several techniques to make a vast corpus of criminal cases (written in Dutch) easily *accessible*. More in particular, SALOMON automatically extracts relevant information out of the full text of a case, and uses it to compose a summary of each decision. Each criminal case will be represented by a separate index card allowing the user to dispose at a single glance of the most important information in the case: the name of the court that issued the decision, the date of the decision, the offences charged, the relevant statutory provisions and the most important legal principles applied. Such a case summary facilitates the rapid determination of the relevance of the case. The summary components can easily be integrated in a database. In a preparatory survey we identified the growing need for text extraction and summarisation systems in the legal field.

Secondly, SALOMON wants to contribute to the study of more general methods for classification, extraction and summarisation of different types of text, and to increase the understanding thereof.

1.2 Legal-social relevance

The SALOMON techniques are of immediate interest for at least two important user groups: the users of the system itself (institutions and companies storing and processing

legal information) and the end users who will be using the information made available by the system (practising lawyers, students and legal scholars).

It is clear that the results of the SALOMON project are important for legal practice: they should *improve the access to criminal jurisprudence*. This evolution is very noteworthy for editors of existing legal information systems still being filled up manually today, which is a slow, labour-intensive and expensive operation. Currently, the selection of published jurisprudence is greatly arbitrary. Judges and lawyers forward certain of their decisions to legal editors. A complete overview of Belgian jurisprudence in a certain field is non-existing at this moment. SALOMON techniques should help overcoming this difficulty by automatically drawing up an index card for every single criminal case in order to collect them in a central database. An important user emancipation is the result thereof: the user can select cases of interest to him/herself, whereas before this selection was made by editorial staff. The disposition of a full overview of criminal jurisprudence has great additional value for policy purposes. It allows an immediate evaluation - and if necessary - re-orientation - by the government of its criminal policy.

1.3 Background

Part of the SALOMON research concerns automatic abstracting of text. Abstracting is a form of information capture. Document abstracts, generated automatically, generally belong to two types. Firstly, the abstract is constructed for easy and fast determination of relevance: it indicates at a single glance whether or not the complete text version is of interest (*indicative abstract*). Secondly, the abstract is a document surrogate expressing the main contents of the document: its components may be used for text search and linking (*informative abstract*). In this way abstracting has some *relation with indexing*. A very brief summary may serve as a complex structured index description (Sparck Jones, 1993). The components of the case summary (e. g. words, phrases and sentences) can be used as indices or keys for accessing the information of the text of the case. At present the majority of automatically generated abstracts are document extracts. It has been shown that document extracts consisting of 20% or less of the original may be as informative as the full text (Kupiec, Pedersen and Chen, 1995).

The automatic generation of document abstracts has early been recognised as a potential area for automation (Luhn, 1958). In early times, automatic abstracting and information retrieval (especially text indexing) were strongly related (Luhn, 1957, 1958; Baxendale, 1958; Earl, 1970). Since then, automatic abstracting has never received special attention, apart from the application of artificial intelligence techniques in restricted text domains (for an overview see Jacobs (1992)). With the current information overload, automation of text summarisation received renewed interest. An example of the automatic generation of case summaries in the legal field and their use for information retrieval is FLEXICON (Fast Legal Expert Information CONsultant) (Gelbart and Smith, 1991; Gelbart and Smith, 1994). According to Sparck Jones (1993) progress in automatic abstracting may be realised along two directions. First, *text structure* is important when accessing the content of a text. For modelling the text structure of the different text types and for relying on it for text processing tasks such as text generation, abstracting and retrieval, we may build on realisations in *natural language processing*. Secondly, the progress made in *information retrieval*, especially the current refinement and sophistication of *statistical techniques* developed for the identification of index terms and text structure, may be fertile for automatic abstracting of texts of unrestricted domains. Parallel, in the information retrieval field there is growing interest in complex indices for document access. It is along the two directions, suggested by Sparck Jones (1993) that we developed the SALOMON project.

2 Methods

It is useful to consider the manual process of abstracting legal cases, not only for defining the desired output, but also in finding appropriate techniques, which may be automated for automatic abstracting. The intended output for SALOMON is inspired on the abstracts actually preceding every publication of a legal case in magazines or retrieval systems. These abstracts are drawn up manually by specialised staff. They consist of several *keywords* (describing the legal question treated in the case) and a short *summary* of the case (reflecting the legal principles applied by the court).

The drawing of the abstract mostly happens according to the following technique: the summary is composed first by extracting one or more interesting paragraphs from the decision. Consequently, the appropriate keywords are selected, either from a fixed list (related to the classification of the case), either they are copied from the text of the case.

Some of the *recommendations for manual abstracting* (Kintsch and van Dijk, 1978; Pinto Molina, 1995) have a *potential in automatic abstracting*. These recommendations concern the recognition of fundamental characteristics of the document as form, class, and structure of the information, the deletion of insignificant and redundant information, and the selection of thematically important sentences.

The general process of automatic abstracting can be described as the transformation of an abstract representation of the source text, containing the necessary attributes for summarisation into a summary representation embodying the organised content of the summary. It is critical to define the *attributes* of the source text representation. These attributes contain information directly *supplied by the input texts* or include information *supplied from knowledge sources* that support the information supplied by the input texts. Sparck Jones (1993) distinguishes two strategies in automatic abstracting. A first strategy relies on the surface structure of the text and is called *shallow processing of the text*. Although, in this strategy text processing relies on some heuristics, the knowledge about the text is very restricted. The text is processed based upon occurrence statistics of index terms and locational cues. A second strategy employs additional knowledge to interpret the surface features found in the text and is called *deep processing of the text*. It entails performing a *detailed semantic analysis of the source text based upon a semantic representation of the text type under consideration*. The summary relies on common, "expected" structures in the text, which form the basis for the summary. The representations can capture linguistic information, domain world information, or communicative information. Knowledge intensive methods are successful in restricted domains. Handling dynamic changes in the input texts is difficult.

Both strategies are applied in the SALOMON system. This was necessary because part of the text to be summarised is predictable (logical structure and category of the case, irrelevant paragraphs of the offences and the opinion of the court, irrelevant legal foundations), while other text parts treat unrestricted subject matter (delict descriptions of the alleged offences and the argumentation in the opinion of the court). The former is processed based upon a text grammar (deep source representation), the latter is summarised based upon shallow statistical techniques.

SALOMON employs *deep text processing* to automatically categorise the cases and to identify their logical structure (Moens and Uyttendaele, 1996a). Additionally, irrelevant paragraphs of the alleged offences and the opinion of the court, and irrelevant legal foundations are identified based on indicator phrases. The knowledge involved is implemented as a text grammar. A text grammar may be defined as a system of text features such as text structure and word arrangements, which deals with the functions and relations of these features in the text. The text grammar is formally represented by a semantic network of frames. The skeleton of the representation is a network of frames which have a hierarchical (*has a*), sequential (*precedes*) or conditional (*if ... then)* relation between them. Each frame represents a segment or subsegment of the text. Segments may be optional. Segments belong to one of the following segment types: text

block limited by word pattern(s), paragraph, sentence, or phrase. Each type may be characterised by specific word patterns or a logical combination of these patterns. Patterns consisting of one or several words are the most important features that delimit or classify text. Word patterns with a same delimiting or classifying function are grouped in a semantic class. A pattern and its class are represented by frames. A pattern class frame is connected with a delimiting or classifying relation to the segment frame. A parser was implemented to process the case based upon the text grammar.

In SALOMON *shallow text processing* is employed to eliminate redundant information in the alleged offences, to group the paragraphs of the opinion of the court into thematically coherent units, and to identify thematically important text units and key terms (Moens and Uyttendaele, 1996b). Shallow processing techniques are needed because crimes concern every aspect of society and the linguistic context of the information is not predictable particularly because cases may cover delicts not previously seen as a result of new legislation.

When automatically grouping paragraphs of the alleged offences and opinion of the court, SALOMON builds upon current research in information retrieval. Full text retrieval of long texts may benefit by the structuring of the text according to topics and subtopics. In this way the user may efficiently query portions of the text (Brown, Foote, Jones, Sparck Jones and Young 1995). The research of Salton and his co-researchers (Salton *et al.*, 1994; Salton *et al.* , 1996) focuses in *finding subparts of a large document that are very similar in context*. In this way linear text is transformed into structured text representations and relevant text excerpts may be identified. Small text units (e. g. sentences, paragraphs) are represented as vectors of weighted index terms.[1] Similarities between text vectors are calculated as the inner product or the cosinus coefficient, both two well known similarity coefficients for comparing text vectors.[2] Text units are thematically grouped, when the similarity between them exceeds a preset value. Hearst and Plaunt (1993) also used patterns of lexical connectivity between text units to identify the subtopics of a text. Here, only similarity values between adjacent text units are computed and placed in a graph. Ruptures in the topic structure of the text are identified as valleys in the graph.

SALOMON not only thematically groups the paragraphs of the alleged offences and the opinion of the court, it also identifies significant paragraphs and key terms to include them in the index card. Each paragraph is represented as a vector of weighted index terms. The index terms of the alleged offences are weighted with the in-paragraph frequency, the index terms of the opinion of the court with the inverse document frequency. In different experiments paragraphs are compared with the inner product and cosinus coefficient. A variant of the *k-medoid clustering method* (Kaufman and Rousseeuw, p. 68 ff.) is used to group the paragraphs of the opinion of the court. The k-medoid method searches the best possible clustering in *k*-groups of a set of objects. A set of objects is automatically divided in *k*-groups such that the average similarity between paragraphs of the same cluster and their medoid is maximised. The number of paragraphs to be clustered is limited, so we can test different *k*-values and select the best possible solution. The best number of clusters is the *k*-value for which the similarities between objects of a cluster are maximised and for which the similarity of the objects with their second choice medoid is minimised.

We employ the *covering clustering algorithm* (Kaufman and Rousseeuw, 1990, p. 111) to eliminate redundant paragraphs of the alleged offences. In this algorithm the number of clusters is not fixed, but each object must at least have a given similarity (threshold) with the representative object of its cluster. The objective is to minimise the number of representative objects.

The medoid of each cluster forms a representative description of each crime/topic treated in the alleged offences/opinion of the court. The medoid of the cluster is the object of the

[1] An overview of index term weighting functions is given by Salton and Buckley (1988).

[2] An overview of similarity functions used in text-based systems is given by Jones and Furnas (1987).

cluster that has a maximum average similarity with all other objects of the cluster. Each cluster of opinion of the court paragraphs is represented by its most important keywords.

At present we limit ourselves to the extraction of relevant information from the case, which form the fields of the index card. No attempt is made to re-edit this information.

3 Results

3.1 Corpus analysis

The choice for a corpus of criminal decisions as a research object, was no coincidence. Only criminal cases are available in machine readable format for the time being. Moreover, criminal law is clearly structured and criminal decisions have a fixed, recurring composition. For the SALOMON project, a corpus was used consisting of all the decisions that the correctional Court of Leuven pronounced between January 1992 and June 1994. These are more than 3000 documents altogether, containing more than 5000 offences charged.

Basically, the cases can be classified into 7 main categories, distinguishing general decisions from particular ones. The latter are concerned with appeal procedures, civil interests, refusals to witness, false translations by interpreters, infringements by foreigners or the internment of people.

Although belonging to different categories, all criminal cases have a typical structure. They are made up of 9 elements, some of which are optional (cf. Figure 2):

- *superscription*, containing the name of the court, the date and the registration numbers of the court administration and of the prosecutor
- identification of the *victim*
- identification of the *accused*
- *alleged offences*
- *transition formulation*, marking the transition to the grounds of the case
- *opinion of the court*
- *legal foundations*, containing statutory provisions applied by the court
- *verdict*
- *conclusion*

The SALOMON techniques were developed in order to extract and summarise the most relevant parts of the cases: the alleged offences, the opinion of the court and the legal foundations.

The *alleged offences* give an exact description of the crimes a person is accused of. At least 50 percent of the cases studied judge more than one offence. The accused may have committed two or more offences, or there may be several accused involved in the same case. The *opinion of the court* allows to distinguish three types of cases within the studied corpus: *routine cases* (containing only routine/unimportant grounds in their opinion), *non-routine cases* (containing other than routine-grounds) and *leading cases* (containing more than 5 “principle grounds”). Principle grounds are the paragraphs of the opinion in which the court gives general, abstract information about the application and the interpretation of some statutes. The leading cases only represent 3 to 5% of the total corpus. Finally, the *legal foundations* consist of a complete enumeration of legal texts and articles applied by the court. Several of these foundations (*routine foundations*) are cited in each case; they have no relevance for the user. The user is only interested in the foundations concerning the essence of the case.

3.2 Architecture of the demonstrator (Figure 1)

The initial processing of the case by SALOMON identifies the general category and the logical structure of the case with the help of a text grammar. Additionally, irrelevant portions of the alleged offences and the opinion of the court are recognised. The result is a case tagged in SGML-syntax.[3] From the tagged case general information about the case such as date, name of the court and relevant legal foundations are easily extracted and placed on the index card. The relevant parts of the alleged offences and opinion of the court need further processing. Key paragraphs and terms are extracted using the above described clustering methods. The index terms, needed for the vector representation, are selected with the help of a thesaurus with index term weights and/or with the help of a language processing module.[4]

3.3 Initial categorisation and structuring of the cases (Figure 2)

SALOMON (Moens and Uyttendaele, 1996a) realises an *essential categorisation of the correctional cases*: general decisions are distinguished from the particular ones. Also the structuring of the correctional case in segments and subsegments is accomplished. Additionally, insignificant portions of the alleged offences and the opinion of the court are identified.

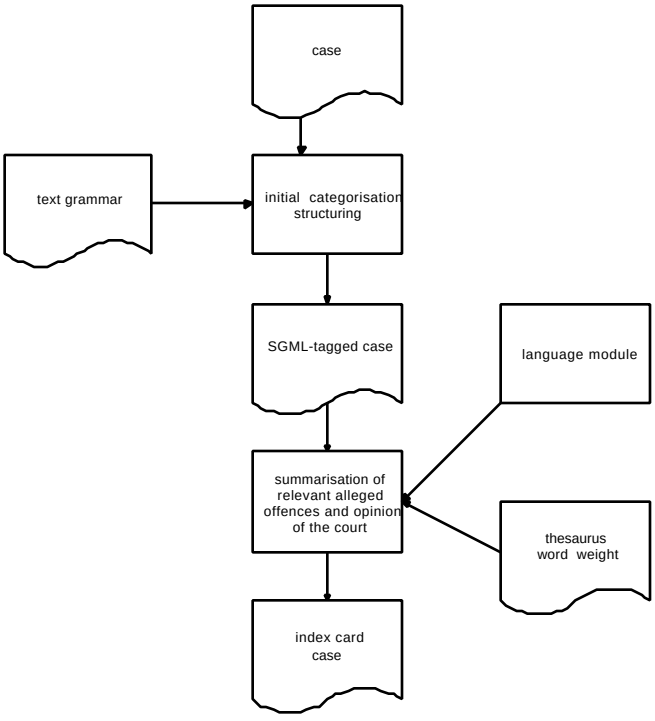

Figure 1: Architecture of the demonstrator

[3] In the future courts may tag case category and logical structure during text generation.
[4] The language module is being developed by Dr. R. Gebruers. After it is linked to SALOMON, we should be able to select nouns and verbs as index terms, as well as word stems (Uyttendaele *et al*., 1995).

A sample of 1000 correctional cases, composed of 882 general and 118 specific decisions, was drawn from the original corpus. The results were manually verified by a student finishing law school. The effectiveness measures most widely used in information retrieval are recall and precision. For each case and segment category recall and precision were computed respectively as the proportion of correct assignments to the category upon the real existing number of this category in the case base, and as the proportion of correct assignments to this category upon the number of assignments to this category. For the case category an average recall and precision of respectively 0.95 and 0.99 is achieved. For the case segments an average recall and precision of respectively 0.88 and 0.93 for general decisions and respectively 0.66 and 0.88 for special decisions is obtained. In general precision is higher than recall. Recall errors are usually the result of lack of knowledge (especially the case for specific decisions), whereas precision errors may be due to ambiguities in the knowledge. A substantial number of errors were caused by typing errors.

<appeal>
<superscription> *Griffie Nr.: ...*
<court> *Correctionele rechtbank te Leuven* </court> ...
<date> *20 januari 1993* </date> ...
In de zaak van het Openbaar Ministerie en van:
</superscription>
<victim> ...
</victim>
<accused> *Tegen ...*
Eiser op verzet ...
</accused>
<alleged_offences>
<routine_paragraph> ...*Beklaagd te: ...* </routine_paragraph> ...
<routine_paragraph> ...*Uit hoofde van ...* </routine_paragraph> ...
</alleged_offences>
<transition_formulation> *Gezien de stukken van het onderzoek ...*
Gehoord het openbaar ministerie in zijn vordering
</transition_formulation>
< opinion_of_the_court> *Overwegende dat ...*
<routine_paragraph> ...*inbreuk ... vaststaat...* </routine_paragraph> ...
<routine_paragraph> *Gezien de beschikking...* </routine_paragraph>...
</opinion_of_the_court>
<legal_foundations> *Op deze gronden en met toepassing van de artikelen ...*
<routine_foundations> ...*Wetboek van Strafvordering...* </routine_foundations>
</legal_foundations>
<verdict> *DE RECHTBANK ...*
</verdict>
<conclusion> *Aldus gedaan en uitgesproken ...*
</conclusion>
</appeal>

Figure 2: SGML-tagged case

3.4 Summarisation of the alleged offences and opinion of the court (Figure 3)[5]

A delict description in the alleged offences is disclosed in a separate paragraph of the text. Such a description contains the specific facts of the delict, integrated in the text of

[5] General evaluation of this process is under way and the results will be available at the time of the conference.

the description. The alleged offences may contain several delict descriptions. Often, some of them are referring to the same crime, but to different facts (in case of multiple accused) or are cited in an abbreviated form. SALOMON discriminates distinct delict descriptions (key paragraphs) and eliminates redundant descriptions. SALOMON recognises non-routine cases from routine cases. In the non-routine cases, where the opinion of the court is a long and elaborated text, the system groups the paragraphs of the opinion around the subjects treated and extracts a key paragraph and key terms of each group (Moens and Uyttendaele, 1996b).

4 Discussion

Text structure is especially prominent in Belgian legal cases. The use of this structure for automatic abstracting fits the current research interest in using text structure for abstracting and indexing purposes. An substantial part of the text structure is identified based on knowledge about the text type. This knowledge is organised as a text grammar, incorporating not only the attributes of the text type, but also the relations between them. In this way a more elaborated semantic model of the text type is created and a refined identification of relevant information in the cases is possible, which may be more advantageous than the use of simple thesauri of indicator phrases as in FLEXICON (*supra*). According to Paice (1991) "superstructural" schemes or grammars are promising for automatically abstracting text. Document grammars have a well-known potential for modelling multi-media documents.

The topic structure of elaborated offences and opinions of the court is automatically recognised building on techniques, recently developed in the domain of information retrieval. In this way redundant information is deleted from the delict descriptions and thematically coherent text pieces of the argumentation of the judge are identified.

The use of cluster algorithms based on the *selection of representative objects* is new in the context of text-based systems. These algorithms provide the possibility to identify highly informative text units that through their lexical patterns are linked to other text units (*cf.* Prikhod'ko and Skorokhod'ko, 1982). We assume that these that these highly informative text units are relevant to include in the case summary. FLEXICON (*supra)* extracts relevant text units based on techniques developed by Luhn (1958), Baxendale (1958), Edmundson (1969) and Earl (1970) such as locational heuristics, frequency occurrences of index terms, and the use of indicator phrases. In order to obtain a balanced summary of the opinion of the court that contains a representative paragraph and key terms regarding each topic treated, a cluster algorithm, which produces a natural clustering, is employed in stead of using threshold values in determining the cluster structure.

In SALOMON we started from the manual practice of abstracting legal cases. Part of this process can be automatically simulated. This includes the identification of the case type, the structure of the information, deletion of redundant and insignificant information, and selection of thematically relevant text units and key terms. In this way we obtain a summary of the case, which is about 20% of the size of the full text of the case (Figure 3).

However, part of the manual process seems out of reach (Paice, 1990; Pinto Molina, 1995). It would be wrong to overestimate the possibilities of legal text extraction systems, they are still far from ideal. The legal field is not straightforward in the way that there is only one unique solution possible to a problem. This subjectivity of the law causes severe problems in designing legal extraction systems such as SALOMON, mainly due to the use of knowledge bases on the one hand and of statistical techniques on the other.

Knowledge bases inevitably reflect a certain *interpretation* of the cases, a problem the SALOMON-team was confronted with when implementing the knowledge of the text grammar. In order to avoid too much subjectivity in identifying the irrelevant

paragraphs of the alleged offences and of the opinion of the court, the knowledge base contains a limited amount of patterns indicating the irrelevancy of the respective paragraph. A more elaborated knowledge base would increase the risk of subjective interpretations. Different knowledge engineers have different ways of selecting, representing and processing knowledge. Other interpretations may be perfectly valid as well. This is exactly the reason why even manual abstracts of legal cases are no more than the extraction of relevant text parts. There is no need to go that far as to re-edit the text, given the danger of misinterpreting or misrepresenting the case. After all, it is not up to the abstractor to make the law.

NAME OF CASE = /users/sien/motivs3/D
DATE = 2 juni 1993.
COURT = CORRECTIONELE RECHTBANK TE LEUVEN
REPRESENTATIVE PARAGRAPHS OF THE OFFENCES=
A. In overtreding van art. 1, 4, 5 en 6 van de wet van 24 oktober 1902, gewijzigd door de wet van 19 april 1963, op welke plaats en onder welke vorm ook, voordeel getrokken te hebben van kansspelen, hetzij dat ze zelven of door hunne gelastigden daaraan deelnamen, en te hunnen bate voorwaarden stelden welke de kansen ongelijk maken, hetzij dat ze van de personen wien het toegelaten werd daaraan deel te nemen een loon in geld ontvingen of iets afhielden van de inzet, hetzij dat zij zich rechtstreeks of onrechtstreeks enig ander voordeel verschaften door middel deze spelen, namelijk het banken georganiseerd te hebben.
B. In overtreding van artikelen 2, 4, 5 en 6 van de wet van 24 oktober 1902, gewijzigd door de wet van 19 april 1963, zelfs wanneer zij hoegenaamd geen toegangsprijs heffen noch enig ander voordeelaanbrengend feit bedrijven, een voor het publiek toegankelijk lokaal houdende er willens en wetens en gewoonlijk spelen toegelaten te hebben die aanleiding geven tot overdreven inzetten of weddenschappen.
REPRESENTATIVE PARAGRAPHS OF THE OPINION OF THE COURT=
Dat de verdediging zich eveneens ten onrechte steunt op het feit dat elders -buiten België- niet wordt opgetreden tegen hanengevechten en of dat er talrijke andere praktijken zijn van dierenmishandeling.
Overwegende dat de beklaagden ten onrechte beweren dat er geen bewijs is van een verboden kansspel.
Overwegende dat ook de strafwaardigheid van de louter deelname aan het spel betwist wordt.
Dat de verbeurdverklaring overeenkomstig artikel 42 van het Strafwetboek, 43 wet 14.8.1986 en 6 wet 24.10.1902 niet alleen nuttig is maar tevens noodzakelijk is althans voor alle zaken die gediend hebben of bestemd waren tot het plegen van het misdrijf en zoals hierna nader zal blijken.
REPRESENTATIVE KEY TERMS OF THE OPINION OF THE COURT=
hanengevechten spel
REPRESENTATIVE LEGAL FOUNDATIONS =
OP DEZE GRONDEN en met toepassing van de artikelen 38-40-42-43-65 van het Strafwetboek; 1-2 en 6 van de wet van 24 oktober 1902, gewijzigd door de wet van 19 april 1963;

Figure 3: Summary of a correctional case

Human abstracting always involves interpretation (Pinto Molina, 1995). Here apart from the objectivity of textual content, certain extra-textual factors intervene, among them the base knowledge of the abstractor, the broad context of the text and the abstracting objectives. Since these aspects of human abstracting are still out of reach in automatic summarisation, SALOMON cannot for the time being automatically select leading cases

by way of statistical techniques[6]. The identification of principle grounds in the opinion of the court is typically a subjective operation, not only because of the interpretation involved, but also because of the need for *contextual information*, to be found within as well as beyond the text of the case: other statutory provisions, legal principles, and multiple social customs and norms. It is up to the user himself - with the help of the full-text of the decision- to situate the SALOMON summary in a general contextual framework.
It is perfectly possible to draw up manually more than one good summary of a case. Hence, the SALOMON output cannot be matched to the one and only correct summary of the case. The accuracy of the output can only be evaluated properly by a college of experts or by the user himself (Dabney, 1986).

Systems like SALOMON can simplify the lawyer's job a great deal. Unable to provide the user with ready-made answers to complicated legal cases, they can at least direct him towards documents where the answer must be found, and even represent them on the screen (Moles and Dayal, 1992). The legal text extraction system is no more than a lawyer's tool, like a book or a library, telling him what the law is in a certain case and where to find it (Zeleznikow and Hunter, 1992, 1994,). It is not intended to perform the interpretation of cases or legislation itself, only to assist the user in his own interpretation process. The purposes of the system should be clearly specified in order to make it immediately clear for which type of users the system was intended, and what they can reasonably expect of it (Susskind, 1986). The intervention of domain experts is indispensable, methods for building the knowledge base should be well-considered (Susskind, 1986; Wang and Ng, 1992).

The SALOMON summaries have features of indicative as well as of informative abstracts. They enable the user to judge the relevancy of a case within seconds, without having to read the full text of it. In the mean time components of the summary can be useful as index terms in a search engine.

5 Conclusion

A growing amount of electronically available legal cases enlarges the need for effective access to these documents. The automatic generation of case abstracts is one way to ensure the accessibility of the cases. These abstracts may be consulted as an initial screening of the case text or may be employed for text search.

SALOMON automates part of the manual abstracting practice and yields relevant extracts of the case that are indicative and informative about the content of the case. In a first step it employs deep processing (knowledge-based) techniques to identify the category and logical structure of the case. It has been shown that a knowledge representation as a text grammar is very useful. In the future a text grammar may already be employed during text generation. Shallow (statistical) techniques are employed to structure the full text of the alleged offences and opinion of the court and to extract relevant text units from them. The SALOMON research proposes solutions based on clustering algorithms not previously employed in text-based systems and hereby contributes to the current research of automatic theme generation and text abstracting. Because of the variety of text types, it is hard to obtain complete domain independence when automatically abstracting text. However, the techniques SALOMON employs, are portable other legal texts, possibly written in other languages.

[6] Complete results will be available at the time of the conference.

Acknowledgements

We are grateful to Dr. R. Gebruers for the precious time spent on SALOMON. We thank Tine Bouwen for the verification of the results. We are grateful to Dr. C. Belmans and Dr. J. Huens for technical support.

References

Baxendale, P. B. (1958). Machine-made index for technical literature - an experiment. *IBM Journal of Research and Development*, 2, 4, pp. 354-361.

Brown, M. G., Foote, J. T., Jones, G. J. F. , Sparck Jones, K., and Young, S. J. (1995). Automatic content-based retrieval of broadcast news. *Proceedings ACM Multimedia '95* , pp. 35-43.

Dabney, D.P. (1986). The curse of Thamus: An analysis of full-text legal document retrieval. *Law Library Journal*, 78, pp. 5-40.

Earl, L. L. (1970). Experiments in automatic abstracting and indexing. *Information Storage and Retrieval*, 6, 6, pp. 313-334.

Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the Association for Computing Machinery*, 16, 2, pp. 264-285.

Gelbart, D, and Smith, J. C.(1991). Beyond boolean search: FLEXICON, A legal text-based intelligent system. *Third International Conference on Artificial Intelligence & Law. Proceedings of the conference*, pp. 225-234. ACM, New York.

Gelbart, D., and Smith, J. C. (1994). Automating the process of abstracting legal cases. *International Journal of Law and Information Technology*, 1, 3, pp. 332-334.

Hearst, M. A., and Plaunt, C. (1993). Subtopic structuring for full-length document access. *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (eds. Korfhage, R., Rasmussen, E. and Willett, P.), pp. 59-68. ACM, New York.

Jacobs, P. S. (ed.) (1992). Text-based intelligent systems: Current research and practice in information extraction and retrieval. Lawrence Erlbaum Associates, Hillsdale, NJ.

Jones, W. P., and Furnas, G. W. (1987). Pictures of relevance: A geometric analysis of similarity measures. *Journal of the American Society for Information Science*, 38, 6, pp. 420-442.

Kaufman, L., and Rousseeuw, P. J. (1990). *Finding groups in data. An introduction to cluster analysis.* John Wiley & Sons, Inc., New York.

Kintsch, W., and van Dijk, T. (1978). Toward a model of text comprehension and production. *Psychological Review*, 85, 5, pp. 363-394.

Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (eds. Fox, E. A., Ingwersen, P.and Fidel, R.), pp. 68-73. ACM, New York, 1995.

Luhn, H. P. (1957). A statistical approach to the mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1, 4, October, 1957, pp. 309-317.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2, 1958, pp. 159-165.

Moens, M. -F., and Uyttendaele, C. (1996a). Automatic text structuring and categorization as a first step in summarizing legal cases. Submitted for publication in *Information Processing & Management*.

Moens, M. -F., and Uyttendaele, C. (1996b). Automatic text structuring and abstracting: Building clusters based on the selection of representative objects (in preparation).

Moles, R.N., and Dayal, S. (1992). There is more to life than logic. *Journal of Law and Information Science*, pp. 188-218.

Paice, C. D. (1990). Constructing literature abstracts by computer: techniques and prospects. *Information Processing & Management*, 26, 1, pp. 171-186.

Paice, C. D. (1991). The rhetorical structure of expository text. *Informatics 11. The structuring of information (* ed. Jones, K. P.), pp. 1-25. Aslib, London.

Pinto Molina, M. (1995). Documentary abstracting: Toward a methodological model. *Journal of the American Society for Information Science,* 46, 3, pp. 225-234.

Prikhod'ko, S. M., and Skorokhod'ko, E. F. (1982). Automatic abstracting from analysis of links between phrases. *Nauchno-Tekhnicheskaya Informatsiya, Seriya* 2, 16, 1, pp. 27-32.

Salton, G., Allan, J., Buckley, C., and Singhal, A. (1994). Automatic analysis, theme generation, and summarization of machine-readable texts. *Science,* 264, pp. 1421-1426.

Salton, G., Allan, J., and Singhal, A. (1996). Automatic text decomposition and structuring. *Information Processing & Management,* 32, 2, pp. 127-138.

Salton, G., and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management,* 24, 5, pp. 513-523.

Sparck Jones, K. (1993). What might be in a summary. *Information Retrieval 93: Von der Modulierung zum Anwending* (eds. Knorz, G., Krause, J. and Womser-Hacker, C.), pp. 9-26.

Susskind, R.E. (1986). Expert Systems in Law: a Jurisprudential Approach to Artificial Intelligence and Legal Reasoning. *The modern law review*, 49, pp. 168-194.

Uyttendaele, C., Gebruers, R., Moens, M.-F. (1995). SALOMON: an exercise in legal information extraction. Legal, linguistic and information technology aspects. Notes of workshop *Information Technology and Law*. K. U. Leuven.

Wang, J. T. L., and Ng, P.(1992). Textpros: An intelligent processing document system. *International Journal of Software Engineering and Knowledge Engineering*, 2, 2, pp. 171- 196.

Zeleznikow, J., and Hunter, D. (1992). Rationales for the continued development of legal expert systems. *Journal of Law and Information Science,* pp. 94-110.

Zeleznikow, J., and Hunter, D. (1994). *Building intelligent legal information systems*. Kluwer Law and Taxation Publishers, Deventer-Boston.