SimSum: Simulation of summarizing

Brigitte Endres-Niggemeyer
Polytechnic of Hannover
Department of Information and Communication
Hanomagstr 8
D-30449 Hannover, Germany
phone +49 511 92 96 606
fax +49 511 92 96 610
ben@iks ik fh-hannover de

Abstract

SimSum (Simulation of Summarizing) simulates 20 real-world working steps of expert summarizers It presents an empirically founded cognitive model of summarizing that operationalizes the discourse processing model developed by van Dijk and Kintsch (1983) The observed strategies of expert summarizers have given rise to cooperating object-oriented agents communicating through dedicated blackboards Each agent is implemented as a CLOS object with an assigned actor at the multimedia user interface The interface is realized with Macromedia Director Communication between CLOS and Macromedia Director is mediated by Apple Events

1 Introduction

The SimSum (Simulation of Summarizing) system does what its name promises it simulates summarizing of human experts and thus produces a computational cognitive model of their processing The model concentrates on the specific features of summarizing It presupposes "normal" text understanding and text production The simulation serves scientific and presentational purposes

- As usual, the computer model serves to explain and check the empirical cognitive model which is its foundation
- It prepares a cognitively grounded approach to automatic summarizing, something like agents running through the net and in response to a user's query, bringing home a reasonably short statement (a summary) of the knowledge available
- To its users of today, SimSum shows in a movie-like style how expert summarizers perform real-world working processes, thus complementing a textbook about summarizing The advantage of the simulation ressembles that of a flight simulator As pilotes steer through possibly difficult situations in the physical world, summarizers work their way through a flood of information Both activities are cognitively demanding People understand them better if they are presented with them in realistic settings

Simulation approaches to summarizing are few and far between, but one can point to the SUSY system (Fum et al , 1982, 1984 and 1985) as an ancestor of SimSum SUSY aimed at following human performance in a limited way, though keeping at a distance from real simulation SimSum represents progress with respect to SUSY, because it is empirically founded, it does a real simulation, and it is implemented Furthermore, SimSum innovates through its multimedia user interface

For practical reasons, the SimSum simulation is restricted to 20 working steps involving 79 agents They were chosen from an empirical cognitive model (a "grounded theory" - Glaser & Strauss, 1980, see also Lincoln & Guba, 1985, for the way to implementation refer to Schreiber et al 1993) of summarizing which comprises an intellectual toolbox of 552 strategies, knowledge about the process organization and a set of interpreted summarizing steps Its basis are 54 summarizing processes of 6 experts from the USA and Germany The summarization processes were recorded by thinking-aloud protocols (Ericsson & Simon 1980, 1984) and analyzed under the scientific umbrella of the discourse comprehension model proposed by van Dijk and Kintsch (1983) The experts being professionals working in the context of information systems, three forms of summa-

rizing occur abstracting, indexing and classifying

A simulation system such as SimSum is bound to empirical validity, giving a reverse engineering of a cognitive process Such a reconstruction of human cognitive activities is possible because human experts subdivide long cognitive efforts like summarizing into modules, called here working steps In the thinking-aloud record they are separated by boundary signals such as pauses or interjections It is these working steps that are reconstructed Put in sequence, they yield the model of the process

Since the sequences in the SimSum system are short, there is almost no chance for for seriously dealing with metacognition (Flavell 1981) in the system Hence metacognitive knowledge is simply hard-coded in the form of working plans etc.

In the following, SimSum is explained first at the macro level of system architecture and system components Then the description narrows down to the micro level of processing After a demonstration of the text representation, two exemplary relevance agents are discussed

2 System overview

SimSum currently runs on Macintoshes with System 7 5, a CD-drive, a 17" monitor and some additional RAM as is usual for multimedia applications It is implemented as an object-oriented blackboard system in CLOS and Macromedia Director (see figs 1 and 2)

Cognitive strategies are represented by object-oriented agents grouped around their respective blackboards The agents are equipped with specialized knowledge, e g an indicator phrase lexicon or a basic representation of SGML codes They process text structure in an SGML-like coding and text meaning in a propositional representation The fact knowledge referenced in texts is defined in document-specific ontologies On the screen the agents appear as animated beasts The CLOS objects simulate the cognitive strategies They send out Apple Events to make "their" animals on the stage perform according to the simulation

An access system accommodates user interaction in a movie-like style the user chooses a working sequence and can interrupt at any time to get further explanations about what the cognitive agents do, how they are motivated empirically and how they are implemented

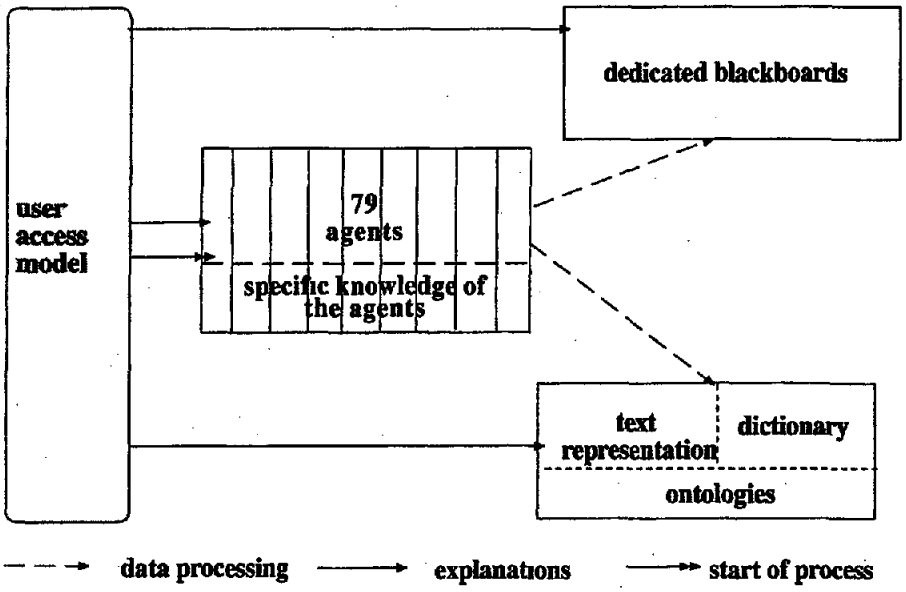

Figure 1 System architecture

Figure 2 gives a screenshot of the SimSum multimedia interface, presenting the relevance assessment agents at work The agent *relevant-texthint* (a ladybird) is putting its candidate statements on the relevance blackboard, while the relevance agents *hold* and *relevant-unit* are sitting on the bench, together with the suspended control agents *explore* for document exploration and understanding (the bee) and *construct* for target text production (the spider) Below, can we see the document blackboard with the representation of the source text, showing its meaning panel, the scheme representation stating the document organization, and the theme representation storing the theme, i e the top of the macrostructure as far as known to the summarizer At the bottom, a comment explains what is currently happening on the screen

The central system components have been derived from observation

- ***Cooperating agents***

Experts use recurring goal-oriented procedures, corresponding to the strategies sketched by van Dijk and Kintsch (1983) These procedures or strategies were operationalized into intelligent agents of the computerized system Agents consist of a script that defines how they deal with their task, they have a general communication component that allows them to exchange messages with other agents and to access global knowledge sources, they may possess private task-oriented knowledge, and they are equipped with task-oriented data views for input and output Control agents (responsible agents for a blackboard - see below) are in addition assigned a little scheduler They activate their subordinates by direct message passing Data exchange between agents takes place via the blackboards The agents keep to the communication rules Strategies / agents cooperate in concrete tasks such as deciding about relevance or setting up a target summary statement Agents may use products of other agents, but since they have limited tasks, they have no sophisticated communication behaviour such as bargaining or discussing

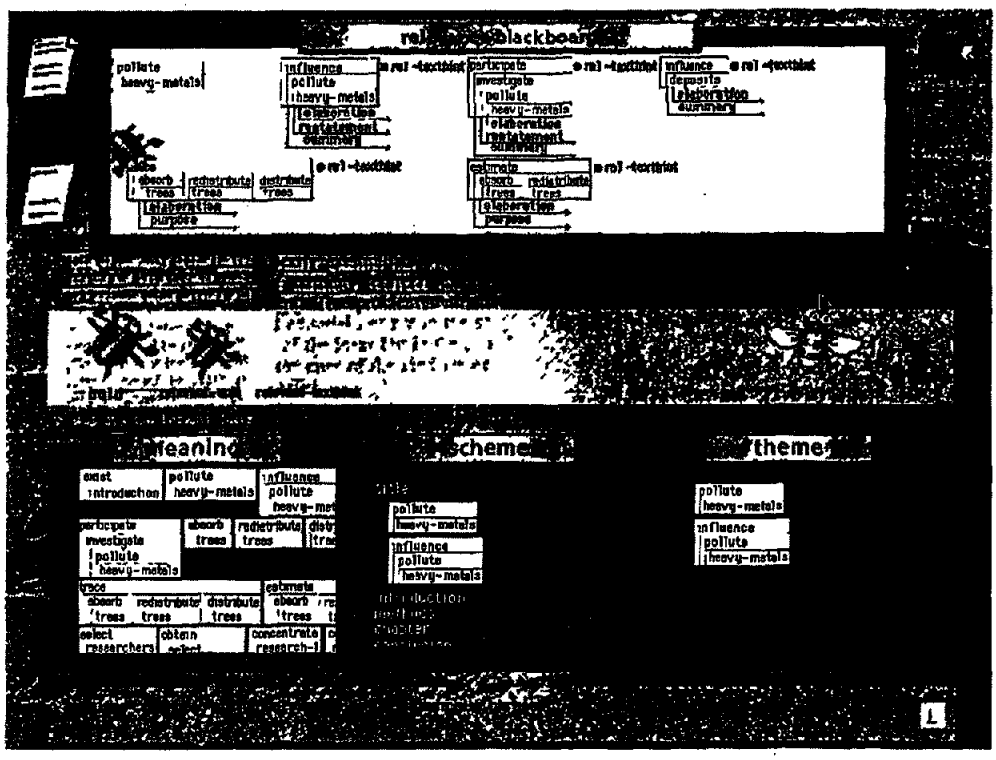

Figure 2 A screenshot of the SimSum user interface relevance assessment agents (ladybirds) are busy while exploration (done by bees) and target text production (by spiders) are suspended

- ***Blackboards***

Agents need communication areas as a medium of cooperation Functionally speaking, these are blackboards (Selfridge, 1959, Carver & Lesser, 1994, Engelmore & Morgan, 1988) SimSum blackboards are dedicated They are used for reception, storage of the input text representation, relevance assessment, target summary construction and so on Central is the document blackboard that stores and organizes all knowledge acquired from the source document (cf fig 2) Since in the case of professional summarizing cognitive processing is modular, the agents work in task-specific groups using a dedicated blackboard For instance, the relevance assessment agents use the relevance blackboard to put the relevance judgement together Every blackboard has a control specialist It organizes the work of the group, sums up what they have achieved, executes the group opinion and delivers the result to the next blackboard

- ***Knowledge base***

The SimSum knowledge base is a common knowledge store comprising a text representation which holds all texts in the system and an ontology of the concepts which are needed to deal with them

3 Computer-oriented discourse representation

Since summarizing is a text and information processing task, we have to represent those surface text passages and text meaning units in the system which are really worked upon, concentrating on semantic and pragmatic structures The representation must support pragmatic text handling and deal with holistic text structures as well as with local microstructures and layout features, because document structure knowledge is a core item of a professional summarizer's competence

- The practical coding of the visible document architecture follows SGML conventions SGML tags like "<h1 1> <h1 1t> "Introduction" </h1 1t> "assign a layout feature derived from content structure In the example in table 1, the section beginning is indicated by <h1 1> Its title is included by the tag pair <h1 1t> and </h1 1t>
- The passages that are really read in the simulated working steps are furthermore coded in first-order-logic-like propositional form (see table 2) During text coding we deliberately chose fitting predicates and standardized presentation (e g ordering of arguments, matching semantically nearly equivalent words in one concept) Domain predicates are distinguished from predicates that describe an interaction between the authors and their readers

<h1 1> <h1 1t> 1 Introduction </h1 1t>

<body1 1> This study forms part of the project "Atmogenous and geogenous components in the heavy metal balance of forest trees" The goal of this project is, on the basis of the distribution within the tree, to trace paths of heavy metal absorption and the regularities of their internal redistribution Furthermore, it is aimed to estimate absorption and redistribution rates In order to obtain as clear results as possible, the majority of trees analyzed were located in areas with atmogenous or geogenous pollution In continuation of the previous studies, which concentrated on trees in contaminated dead ore areas and Black Forest locations with low atmogenous pollution, the following reports about trees influenced by high atmogenous deposits in the district of Stolberg </body1 1> </h1 1>

Table 1 Text representation, SGML style coding of an introduction

```
3  domain_exist (introduction)
4  domain_pollute (heavy_metals, forest_trees, , , , [geogenous, atmogenous])
5  domain_investigate (project, 4)
6  domain_participate (study_this, 5)
7  domain_absorb (trees, heavy_metals, , paths)
8  domain_redistribute (trees, heavy_metals, , internally, , regularity)
9  domain_distribute (trees, heavy_metals, , internally)
10 domain_trace (project, [7, 8], 9, , , , aim)
11 domain_estimate (project, [7, 8], , , , aim))
```

Table 2 Text representation, beginning of the introduction (propositional coding)

- To account for discourse level document structures, SimSum uses text-type specific superstructures (Kintsch & van Dijk, 1983) From a practical point of view, superstructures consist of semantic components which are linked by discourse relations In SimSum, these are RST relations (RST Rhetorical Structure Theory - Mann & Thompson, 1987, Hovy, 1993) While the SGML and the propositional representations are precoded, the discourse level document structures are reconstructed during summarizing The cognitive agents install the respective RST relations Only a few of the most necessary and most simple RST relations have been implemented ELABORATION, RESTATEMENT, PURPOSE, CAUSE/RESULT, EXAMPLE

A small parsimonious ontology has been coded for every document, where the used concepts are organized in a small and very flat hierarchy The ontology is divided into two parts according to Penman (1989) The upper model is domain independent and therefore used for all texts in the system, whereas the lower model is domain specific, so that one is modelled for each document The agents do some basic inferencing such as comparing text units with knowledge base entries and installing relations from a fixed set between text units

4 Agents

The core of the SimSum simulation are object-oriented agents As representatives of the empirically found cognitive strategies they manage the reduction of a large document to a short summary Agents differ in the representations they work upon Some of them are sensitive for SGML tags, others need the propositional presentation to run their methods

In the SimSum system, 39 agents are modelled in great detail They are involved in the central information reduction task of summarizing, e g the relevance agents Reading and writing strategies are realized carefully only in so far as they are specific for professional summarization, otherwise they remain black box agents About half of the agents are "real" agents and the rest are "pseudo"-agents For instance, the *explore* agent is a black box agent of understanding It fakes text comprehension by assigning input passages a precoded propositional representation The *reorganize* agent is a black box agent as well It is presumed to impose English grammar and spelling which is not a specific subtask of professional summarizing Therefore the agent functions more or less as a placeholder

The agents fall into the following functional classes planning and control, exploration, relevance assessment, target text construction, quality enhancement, formulation, and general knowledge processing In addition, there are minor agents such as readers and writers

To make the agents more concrete, we discuss in the following two "real" relevance agents that happen to be good old acquaintances of everybody in automatic summarizing *relevant-texthint* (realizing the indicator phrase method, see, e g , Paice, 1990 and Borko 1968) and *relevant-call*, which assesses the importance of an entity by measuring its distance from the theme (principle used in Jacobs & Rau, 1990, McKeown, 1985, Trabasso & Sperry, 1985) More about agents is found in Endres-Niggemeyer et al (1995) and in Endres-Niggemeyer (1997)

Relevance agents work under the control of *hold*, the responsible agent for the relevance blackboard (cf fig 2) Since the skilled reduction of document meaning to the most relevant items is central to professional summarization, *hold* is in charge of the core of the whole summarizing process

- ***Relevant-texthint***

The *relevant-texthint* agent implements the "indicator phrase method" known since the early days of automatic abstracting It exploits cue phrases by which authors qualify their statements, assuming that the qualification applies to the scope of the indicator phrase By its mere presence, a (positive) indicator phrase expresses the author's emphasis and suggests the relevance of the statement in its scope In addition, cue phrases often explain what the author announces, e g a new finding or the content of the conclusion, and its role in the document

Relevant-texthint reads the propositions on the meaning panel of the document blackboard (see fig 2) To make out relevant propositions, it uses a private dictionary, where the potential indicator predicates (cf table 3) are listed Since the dictionary entries are annotated with interpretations, the agent can draw the attention of other agents to these propositions by passing them parts of its private knowledge

Relevant-texthint recognizes the indicator predicates by simple pattern matching as containing an indicator phrase, matching its dictionary entry with a proposition such as proposition 5 in table 2 Consequently, the agent annotates proposition 4 as describing the project theme and therefore as important and puts it together with others on the relevance blackboard (see fig 2 and table 4)

- ***Relevant-call***

Relevant-call recognizes a text meaning item as relevant because it links it to the document theme (see figure 3) The agent needs the thematic structure and, as a candidate for linkage to the document topic, a text proposition The agent checks whether an open RST-type link of the document theme is able to attach the candidate If so, the proposition is regarded as relevant and added to the document theme

Theme-of-document	**Methods-of-research**
domain_investigate (project, X)	domain_select (researchers, X)
domain_participate (study_this, X)	domain_obtain (researchers,
domain_estimate (project, X, aim)	results_clear, X, aim)
interaction_report (author, X)	**Research-background**
domain_continue (researchers, Y, X)	domain_concentrate (research, X, past)

Table 3 Some propositions from the indicator phrase dictionary

4 domain_pollute (heavy_metals, forest_trees, , , , [geogenous, atmogenous]) theme-of-document
7 domain_absorb (trees, heavy_metals, , paths) theme-of-document
8 domain_redistribute (trees, heavy_metals, , internally, , regularity) theme-of-document
9 domain_distribute (trees, heavy_metals, , internally) theme-of-document
12 domain_select (researchers, [trees/locations_polluted_atmogenous_high/, trees/locations_polluted_geogenous_high/]) methods-of-research
16 domain_influence (deposits/atmogenous_high/, trees, , stolberg_district) theme-of-document

Table 4 Choice of what *relevant-texthint* judges relevant

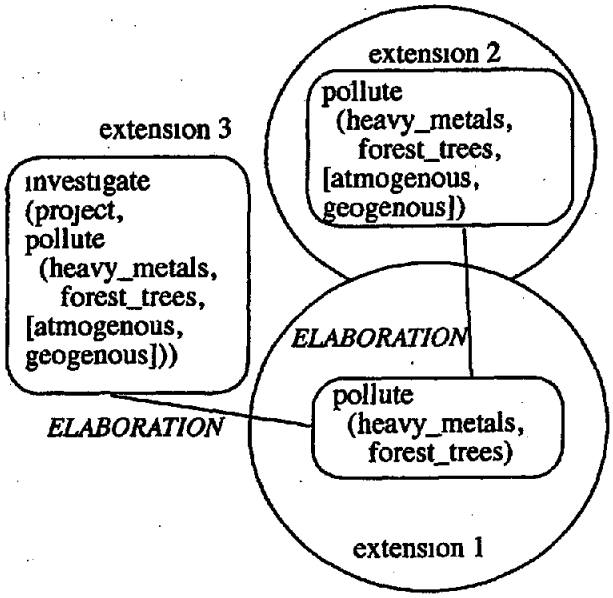

Figure 3 *Relevant-call* expands the document theme

To find the document theme, *relevant-call* accesses the theme panel of the document blackboard The agent tries to attach propositions discovered by other (data-oriented) agents For instance it picks up proposition 4 recommended by *relevant-texthint* because it states what the research is about ("This investigation forms part of a project " - cf table 4) *Relevant-call* tries all available RST-relations in order to link proposition 4 to the document theme (in extension 1) It is easy to see what happens proposition 4 rephrases the theme, the concepts "pollute", "heavy_metal", and "forest_trees" of the theme are repeated The theme and the text proposition unify, but proposition 4 brings some additional information about the ([geogenous, atmogenous]) components of contamination This corresponds to an elaboration of the theme Consequently, the propositon. is attached by an ELABORATION link The new hypothesis of a topic structure is given in figure 3 At that moment, two new propositions have been attached to the theme, so that the theme has three extensions

5 Conclusion

Advancing the scientific frontiers of text summarization presupposes more knowledge about the way summarization works The main fruit of the empirical investigation behind SimSum is an image of the summarization process which is detailed enough to lay the foundations for a simulation Since the resulting summarization model incorporates the know-how of human experts, it has good prospects of presenting powerful techniques Summarizing by cooperating cognitive agents seems to be such a principle

The researchers have reached their aim to show that an observationally founded implementation of summarizing processes is possible However, SimSum is a system in-the-small It suffices to demonstrate how the summarization agents work in their cognitive environment To meet practical challenges such as text summarizing in the WWW, a much more comprehensive system must be realized This means in particular

- providing knowledge bases of real-world size, be they private ones of agents or public resources of the whole system
- choosing the most useful strategies or agents and making them flexible to deal with any legitimate data
- using text understanders or information extraction components as well as generation systems provided by colleagues

6 Acknowledgements

The SimSum development has been funded under grant F 916 00 by the German Federal Ministery of Education and Research

7 References

Borko, H (ed) (1968) *Automated language processing* New York Wiley

Carver, N , & Lesser, V (1994) Evolution of blackboard control architectures *Expert Systems with Applications 7*, 1-30

Endres-Niggemeyer, B (1997) *Summarizing text* (forthcoming)

Endres-Niggemeyer, B , Maier, E , & Sigel, A (1995) How to implement a naturalistic model of abstracting four core working steps of an expert abstractor *Information Processing & Management 31(5)*, 631-674

Engelmore, R , & Morgan, T (Eds) (1988) *Blackboard systems* Wokingham Addison Wesley

Ericsson, K A , & Simon, H A (1980) Verbal reports as data *Psychological Review 87*, 215-251

Ericsson, K A , & Simon, H A (1984) *Protocol analysis Verbal reports as data* Cambridge MA MIT Press

Flavell, J H (1981) Cognitive monitoring In W P Dickson (Ed), *Children's oral communication skills* (pp 35-60) New York Academic Press

Fum, D , Guida, G , & Tasso, C (1982) Forward and backward reasoning in automatic abstracting In *COLING Proceedings of the 9th International Conference on Computational Linguistics* (pp 83-88) Prague

Fum, D , Guida, G , & Tasso, C (1984) A propositional language for text representation In B G Bara & G Guida (Eds), *Computational models of natural language processing* (pp 121-150) Amsterdam North-Holland

Fum, D , Guida, G , & Tasso, C (1985) Evaluating importance A step towards text summarization In *IJCAI Proceedings of the 9th International Joint Conference on Artificial Intelligence* (pp 840-844) Los Altos CA Kaufmann

Glaser, B G , & Strauss, A L (1980) *The discovery of grounded theory Strategies for qualitative research* (11th ed) New York Aldine Atherton

Hovy, E (1993) Automated discourse generation using discourse structure relations *Artificial Intelligence 63*, 341-385

Jacobs, P S , & Rau, L F (1990) SCISOR Extracting information from on-line news *Communications of the ACM 33* (11), 88-97

Kintsch, W , & van Dijk, T A (1983) *Strategies of discourse comprehension* Orlando FLA Academic Press

Lincoln, Y S , & Guba, E G (1985) *Naturalistic inquiry* Beverly Hills CA Sage

Mann, W C & Thompson S A (1987) Rhetorical Structure Theory A Theory of Text Organization In L Polany (Ed) *The Structure of Discourse* Norwood, N J Ablex

McClelland, J L , & Rumelhart, D E (1981) An interactive activation model of context effects in letter perception Part 1 An account of basic findings *Psychological Review 88*, 375-407

McKeown, K R (1985) *Text generation Using discourse strategies and focus constraints to generate natural language text* Cambridge Cambridge Univ Press

Norman, D A (1983) Some observations on mental models In D Gentner & A L Stevens (Eds), *Mental models* (pp 7-14) Hillsdale NJ Erlbaum

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

Penman Project (1989) *PENMAN documentation the primer, the user guide, the reference manual and the Nigel manual* Technical Report USC/Information Sciences Institute, Marina del Rey, California

Schreiber, G , Wielinga, B , & Breuker, J (1993) *KADS A principled approach to knowledge based system development* London Academic Press

Selfridge, O (1959) Pandemonium A paradigm for learning In *Symposium on the mechanization of thought processes* London HMSO

Trabasso, T & Sperry, L (1985) Causal relatedness and importance of story events *Journal of Memory and Language* 24, 595-611