A MODEL OF COLLABORATIVE WRITING

Jan Sefranek, Milos Kravcik,
Faculty of Mathematics and Physics, Comenius University, Bratislava, Slovakia

Abstract

The paper presents a cognitive model of writing. The model is based on a concept of a heterogeneous multi-layered semantic network. The network forms the knowledge base part of the model. The other basic component of the model is a text base. We discuss the operations on the network. Reading and writing, the central interests of our model, are considered as complex processes interconnecting the knowledge base and the text base. A special concern is the aspect of collaborative writing. The basic features of the cognitive model are preserved in a conceptual model (its position is an intermediate one between the cognitive model and a specification of implementation). Its main role is to provide some data structures (a data model) and a control structure appropriate for the design of the system. Our aim is to preserve the basic features of our cognitive model in terms of both the data structures and the control structure. The document concept plays the role of the basic data structure. We use it in a very general sense. The control structures of the conceptual model are supposed to be components of an environment called MONSTER.

1 Introduction

This paper presents a model of collaborative reading and writing. The model is intended as a platform for design, specification and development of a software system supporting creation of documents in a group of people. Computer and software are used in the applications of this kind as a tool mediating human-human interaction.

The computational support of collaborative work and a computer-based human-human interaction are examples of an important current trend: Rapidly evolving impact of information technologies on almost all areas of human and social life.

There are very good reasons to be interested in a system that in a natural way supports some ordinary (intellectual) activities of individuals and groups to provide a powerful and comfortable working environment. But the problem of computational support of (intellectual) collaborative work is not a trivial one for at least two reasons: iterequires an interdisciplinary approach (a good understanding of the requires an interdisciplinary approach (a good understanding of the complex application domain), and an integration of several advanced and complex software (and hardware) components.

Because of the complexity of the problem we restrict our attention to one fundamental aspect. The main goal of the paper is a description of a cognitive model of collaborative writing. We are especially interested in expository writing, which is a typical real-world task of which the essence lies in concept construction and knowledge expression.

In proposing our model we try to understand how ideas of a group can result in a finished text. To avoid misunderstanding we state what is *not* our aim.

We do not want to focus on automation of the document creation process or even on automation of discovery. We assume that this process is fully left to human intellectual activities. Of course, analysis and comprehension of the process can help in specifying, designing, and developing a suitable environment and tools – a software system supporting creation of documents in a group. An intermediate level of support is assumed – our ambition is neither automation/semi-automation of texts generating, nor a simple editor.

The proposed model is not a psychological theory, as we do not focus on psychological adequacy, but rather on a useful abstraction of document processing, motivated by the nature of human cognitive processes.

Now we proceed to a characterization of the main ideas of the paper. The basic mechanism used by human intellect is an associative memory and an ability to abstract – to stress and hide some domain features. It would be useful if both can be supported by tools and by a suitable system of work. The proposed model is intended as a step in the search for such a system.

The kernel of our model is (the conception of) a heterogeneous multilayered semantic network. The processes of reading and writing are based on the operations on the network. Similarly, collaborative writing is explained over a structured and evolving complex of private and common subnetworks. Important facilities provided by the network are: an ability to focus attention, to change the level of abstraction and detail, and a crucial ability of distinguishing relevant (irrelevant) ideas in a knowledge base.

The paper consists of four sections. In Section 2 we provide an overview of several basic cognitive models of writing and an outline of our approach. In Section 3 a description of our cognitive model follows. A conceptual model of collaborative writing is introduced in Section 4. Finally, in Section 5 we mention several key features of a proposed implementation.

2 Overview of Related Models

In this part we introduce a brief overview of several basic cognitive models and architectures for writing.

Expository writing, or knowledge construction, belongs to long-term and ill-defined problems, so their solution requires complex strategies or interactions with other subjects. At present, in the attempt to solve these problems, one can formulate only general rules. It is not possible to determine exactly when the task is complete. These are the reasons why there are no rigorous and consistent models for problems defined in this way.

From the models presented here the simplest one is that by [13]. Besides a model for writing, he also introduces a model of reading. In this model mental representation of meaning takes the form of *propositions* or *relationships*. A reader links propositions with one another and creates *local coherence* across sentence boundaries in short-term memory. By establishing coherence, information can be moved from short-term memory to long-term memory. Further *hypotheses* about intentions of the text are generated, refined, and propositions are combined into larger structures. The reader tries to activate the largest relevant structure from memory – the *superstructure*. In describing the reading model, Rada also considers functions. A *reading control system* translates the locally coherent information into globally coherent information. It uses *spreading activation* to obtain further propositions from new and neighboring propositions. Rada's writing model corresponds to Piaget's conception that knowledge and understanding proceed in three dominant steps:

- knowledge of elementary facts, recognition of basic elements,
- discovery of relations (among elements, facts),
- global understanding.

Rada's writing model assumes that the final form of a text is determined by goal and audience. According to him writing has three phases:

- exploring,
- organizing,
- encoding.

These three psychological processes express creation of unstructured notes, their hierarchical ordering, and writing the target document. Rada also mentions an extended goal-oriented writing model, which is a simplified model of Hayes and Flower [8], [6]. Their cognitive model for writing consists of three basic components:

- task environment – formed by a writing assignment and a preliminary text,
- writer's long term memory – including knowledge of topic and of audience as well as writing plans,
- processor – containing three high-level processes together with their subprocesses and monitor, providing overall control of the writing system and defined as a sequence of production rules.

The three high-level processes of the processor are:

- planning – consisting of three subprocesses – generating (ideas), organizing (document scheme) and a goal-setting component to control movement among these processes,
- translating – encoding the ideas into continuous prose,
- reviewing – reading the produced text and its editing.

In comparison to Rada's model, exploring is here called generating, and together with organizing they constitute 'higher-level planning'. Encoding is divided into translating and reviewing. This analytical and conceptual model has brought many new details compared with its predecessors, but it was still not enough to formalize the model into a form suitable for simulation, or for the construction of a predictive program.

Smith [14] has introduced a framework for the description of complex real-world knowledge construction tasks on one hand, and on the other to help in the development of a theory-based computer system supporting these tasks. It is based on a set of cognitive modes which are used by individuals to perform a task, and on strategies exploited for moving between these modes. A *cognitive mode* is considered as a particular way of thinking used for a particular purpose. It is determined by four factors – goals (abstract), products (concrete), processes, and constraints.

The authors distinguish seven cognitive modes of expository writing:

- exploration,
- situational analysis,
- organizing,
- writing,
- editing – global organization,
- editing – coherence relations,
- editing – expression.

When we compare this approach with the model of Hayes and Flower generating corresponds to exploration, translating to writing, and reviewing to editing, which is divided into three parts – global organization, coherence relations (between sentences and paragraphs), and expression (linguistic analysis). Part of the model is situational analysis, which includes an analysis of objectives and audiences, as well as prioritization. Thus it covers goal-setting, but it is even more general.

This approach was also oriented more towards analysis and description than to prediction. In contrast to the previous model, these models are formal, that have directly led to design of a computer based system. The models have been expressed as grammars which after being implemented as parsing programs can help to study users' cognitive behaviors.

Figure 1: Comparison of models

These models do not differ from one another in principle, but only in levels of abstraction or degrees of detail they employ. Our model also corresponds to them in a certain sense. If we use the terminology of Hayes and Flower, the memory of a writer corresponds to our knowledge base.

The task environment is formed by a text base. The processor in our considerations oscillates between development of a document content and its structure, using functions as generating, elaborating, and reviewing.

The main objective of expository writing usually is presentation of a certain view on a problem domain where the level of knowledge of a potential audience is taken into account. Such a domain is often complex and ill-structured, which leads to the danger of its misrepresentation or oversimplification when it is being transformed into a linear form. Thus on the writer's side the ability to spontaneously restructure his or her knowledge in adaptive response to changes in situational demands is important. This ability is called *cognitive flexibility* by Spiro [15] and plays a key role in our model. It involves that we are focused on such problems as choosing a scope of the domain, a point of view, a level of abstraction, a set of concepts, and realizing or discovering relationships. These are the functions that should be included in our model. They help to activate relevant knowledge in the writer's long-term memory and this knowledge becomes the content of his or her working memory. Additionally our model should be formal enough to serve as a base for a programming system supporting collaborative writing.

3 Cognitive Model of Writing

In this section we introduce our cognitive model of writing. We consider two basic components:

- a knowledge base,
- a text base.

Reading and writing, the central interest of our model, interconnect these two bases. First we focus on the knowledge base to get a platform for the investigation of reading and writing. The knowledge base is an important part of our cognitive model. To avoid misunderstanding we stress what it is not to be:

- psychologically satisfactory theory of human mind,
- representation of natural language semantics,
- knowledge base for automatic discovery or text generation,
- knowledge representation language.

Its aims are more modest: it is to be an abstraction of human knowledge in the task of reading and writing. It can be localized in one mind, but also in more than one, or in some external sources. The primary objective of this abstraction is purposefulness, the psychological adequacy is so far not relevant. Our model must be viewed as an excercise of informatics (computer science) and logic, rather than of psychology.

3.1 Semantic Network

In this subsection we discuss the knowledge-base part of our model in more detail. The first point to notice is that the knowledge base is defined as a semantic network (Endnote 1). We now introduce some basic definitions.

Let us consider a set. The members of the set are said to be *nodes*. An *oriented edge* is an ordered pair of nodes. If an edge is oriented from the node u to the node v then the former node (u) is called the *source* and the later (v) is called the *target* (of the edge). Each node and edge has a name called the *label*. If there is a label of a node (or an edge) we call the node (the edge) *labeled*.

An *oriented graph* is a set of nodes and a set of oriented edges. A *labeled* oriented graph is an oriented graph such that at least some of its nodes and/or edges are labeled. (Endnote 2)

We can distinguish distinct types of nodes. Here we are not interested in a detailed specification of node types. In the next example some possible types are shown: *individual concepts, generic concepts, verbs*.

Example 1:

Figure 2: A simple semantic network

A fragment of a simple semantic network is presented in Figure 2. The nodes of the graph are intended to represent concepts. Some of the nodes represent individual concepts. Examples: *Joe* or *Milka*. Other nodes represent generic concepts, for example *mammal, thief, cow*.

An edge labeled *is* expresses that its source is an individual concept and its target a generic concept. Intuitively, an individual is a member of a class (an individual concept is an instance of a general concept).

An edge labeled *is_a* represents a relation of a subclass – cows are a subclass of mammals (in other words, the general concept "cow" is a specialization of a more general concept "mammal").

There are other possible labels of edges, they represent some relations between concepts (for example synonymity, similarity, affinity). We will not take a more detailed account of these possibilities.

The presented fragment of a semantic network also contains a specification of the meaning of the verb *steal* using a *case frame* [5, 1]. The case frames specify the semantic roles that a noun phrase may play within a verb phrase. The semantic roles (cases) are either obligatory or optional. Edges incident with the given verb-node are labeled by a label from a small set (of cases). The verb *steal* from our example is characterized by two obligatory cases – it has an *agens* and an *object*. (Endnote 3)

The labels on the targets of edges with a verb-concept node express restrictions on the type of arguments: the agens has to be a *Person* and the object has to be a *PhysObj*.

Apart from the specification of the meaning of the verb *steal*, our example represents a fact: that *Joe steals Milka*.

Our next goal is a definition of a (generalized) network. First we need the hyperedge concept. Let a graph G be given. Then a nonempty set of its nodes G is called an *hyperedge*. We will assign a label to each hyperedge. (Endnote 4)

Example 2: Let a set of nodes (representing some books) be given. Let the label of each node be the title of the corresponding book. The hyperedge labeled as *genetics* links all books which are devoted to genetics.

Up to this moment with each node (edge or hyperedge) we associated exactly one label. Now a more general convention is accepted: To each node, oriented edge, and hyperedge a set of attributes is assigned. Some of the attributes may have a value that is a set, i.e. a value of the attribute is a set of labels (associated with given node, edge, and hyperedge).

Example 3: The attributes of the books from Example 2 can be: author(s), title, publishing house, year of issue, number of pages, ISBN, number of edition, keywords (etc.). Some attributes can have several values (for a given node, edge, hyperedge). For example, a value of the attribute *key words* of a given node can be the set *genetics, evolution, undulate*. (Endnote 5)

The nodes, oriented edges and hyperedges associated with a set of attributes are said to be *multiply labeled*. From now on we assume that each node, oriented edge, and hyperedge is multiply labeled (Endnote 6).

Now we give a recursive definition of concept of *network*:

- a *network* is composed of: a set of nodes, a set of oriented edges and a set of hyperedges,
- let a *component* be either a node or a network, a generalized (oriented) edge be an (ordered) pair of components (Endnote 7), and a generalized (Endnote 8) hyperedge be a set of components. Then a *network* is composed of a set of components, a set of oriented edges and a set of hyperedges.

The concept of network concept just introduced enables switching of attention and changing a measure of detail and a level of abstraction in knowledge (and meaning) representation in a natural way.

Example 4: Let us imagine that police records contain a case called *The Milka theft*.

Figure 3: Levels of abstraction

The network on Figure 3 is interesting from an important point of view – it illustrates changing levels of abstraction. The network presented in Figure 3 focuses on who is the responsible police clerk who is investigating the case, what the status of the case is, and that the case is an instance of theft.

The inner part of The Milka theft component is hidden. From another viewpoint, may be, a distinct level of abstraction is relevant. In this case a part of the network from Example 1 can be shown.

Sometimes we will use the notion *heterogeneous multi-layered semantic network* in the sequel (instead of the notion "network"). The reason for the adjective *heterogeneous* is that it enables to connect constructs of distinct character – for example unanalyzable nodes with the relative rich structured networks. The reason for multi-layered is that the objects in a network (components, edges, hyperedges) are – in general – labeled by a set of various attributes. The variety of labels represents different levels the network objects can be approached from.

A *subnetwork* of a network N is a proper subset M of network's N components together with the edges and hyperedges from N which are incident with the components from M.

The private and common parts of the network can be determined by using the subnetwork concept and having it marked by a proper labeling. Subnetworks and multiple attributes enable the expression of degrees of credibility, alternatives, multi-agent knowledge, epistemic operators, meta-knowledge, and auto-reflection. For example, in a natural way (see Figure 4) it is possible to represent the knowledge of the type *the agent x knows, that the agent y knows....*

Figure 4: Epistemic operators

The text base is not an interesting problem for us in this section (a naive conception of a store of texts is sufficient). However, we are more interested in the connections between the knowledge base and the text base. Let us begin with the elementary operations on semantic networks. Then special attention will be devoted to writing (by means of an analysis of reading as warming up).

3.2 Operations on Semantic Networks

We describe the elementary operations on semantic networks and then we discuss the problem of inference over the semantic networks.

The *elementary update operations* are insertion of a new node, deletion of a node, insertion of a new (hyper)edge, deletion of an (hyper)edge, insertion of a new label, deletion of a label (Endnote 9).

Our conception of networks is a dynamic one. The time dimension of a network is important, equally important are versions of various components and alternative subnetworks. The network is evolving in time. New nodes, edges or hyperedges are added, sometimes they are destroyed. The views of the network are changing – thanks to the mechanisms of abstraction and attention focusing (see below).

The *elementary retrieval* operations: finding a node (satisfying a condition – Endnote 10), finding a set of nodes, passing from one node to another through a specified edge, selection of a value of a specified attribute (of a given node/(hyper)edge).

Although the problem of inference on semantic networks is a topic of intensive studies, it is obscured by many unclear aspects, with an evident source – the concept of semantic network is used in very different ways, with various ambitions and goals. This is the reason why until now the task of clarifying the inference on semantic networks is not solved in a satisfactory way.

We can represent the inference by an operator Cn, which assigns to a set X of sentences of a language L (Endnote 11) a set of its consequences $Cn(X)$.

Usually some basic properties of the inference operator Cn are defined. We will discuss a variety of rather different kinds of inference, therefore we will not assume some special properties of Cn.

First, we note that answers to queries can be considered as results of an inference (over the given database or knowledge base). The canonical ability of inference in relational databases is specified as a relational completeness of a query language and as a benchmark is chosen the relational algebra (or relational calculus). We call this type of inference specification the specification by a benchmark ([17] – Endnote 12).

More complicated types of inference were studied in artificial intelligence. If we represent a semantic network in a language of logic (enriched by an index mechanism), we can study the problem of inference in a more precise way, see [17]. The problem of inheritance was studied in this way. A well-understood problem is also inheritance in non-monotonic hierarchies with exceptions and with multiple sources of inheritance, see for example [16].

Induction, learning, and analogy (over semantic networks) are among the interesting cases of inference too.

Until now we have mentioned only some traditional forms of inference with a parallel in logic, artificial intelligence, or databases. If we are working with large semantic networks it is necessary to study some non-standard types of inference, for example *activation*.

While working with a network it is important to focus attention on some parts of it. If we explicitly define a subnetwork P of a network S, it is said that we have *bactivated*b the subnetwork P. (A subnetwork could be represented as a set of sentences and the activation of the subnetwork is a product of an operator selecting a subset from a set of sentences.) Sometimes it is useful if the activation is limited in time (we activate a subnetwork only for a given amount of time). In this case we are able to model forgetting or transfer of attention. A *deactivation* of a given subnetwork may be called an explicit deletion of its definition (for example interpreted as a transfer of attention).

If a subnetwork is specified, it can be useful to consider the whole subnetwork as a node. This task is accomplished by a useful operation – *naming*.

Subnetwork activation and deactivation, and naming are the operations behind the above-mentioned mechanism of abstraction and focus-shifting. This mechanism is used when the structure of a complex object is not interesting or if the given object is yet not well understood.

Moreover, operations of activation (deactivation) and naming serve as a means for distinguishing relevant (irrelevant) ideas in a knowledge base.

The knowledge base of our model accounts for the knowledge expressed both explicitly (recorded in the semantic network), and implicitly (derived, inferred from the knowledge recorded in the semantic network). From this point of view, the basic operations on semantic networks and inference over semantic networks are important aspects of the knowledge base.

A proper segmentation of a knowledge base and useful macro-constructions used in the knowledge-representation formalisms are of a decisive importance from the point of view of their expressive power [4]. This is the reason why we once more accent that the concept of heterogeneous multi-layered semantic network and the mechanisms of abstraction and subnetwork activation are key aspects of our model.

We will discuss reading and writing, two complex actions composed from more basic operations on semantic networks, in the next two sections.

3.3 Reading

Our model considers reading as a (complex) process with a text (being read) as input and a modification of a semantic network as output. (Endnote 13)

Reading is considered as a process consisting of four subprocesses:

- initialization,
- details analysis (local analysis),
- reflection (global analysis),
- finalization.

Our model is an *holistic* one. *Initialization* creates the first, preliminary, but overall image of the text, the first approximation of coherence, and also a decision whether the text will be read (just this decision completes initialization). The overall image of the (new) text is represented by creation of a new node in the network, respectively by interconnection of the node to an activated subnetwork.

An *iteration* of *details analysis* and *reflection* is the core of the reading process.

Intuitively, the basic goal is in putting individual sentences of the text into relationship to the knowledge base (performed operations are e.g. creation of new edges, hyperedges, or nodes, labels modification, activation of subnetworks). After a certain number of sentences we move from the level of sentences to the level of wholes – the subject of reflection is to specify meaningful and relatively closed parts of text (Endnote 14) and to incorporate them into the context of the knowledge base (operations can include naming, modification of labels, creation of new edges and hyperedges). The important parts of reflection are also summarization, annotation, evaluation, commentary on the text (it is implemented by some corresponding modifications of the network, using mainly modifications of labels). We can say that during iteration the portion of reflection gradually increases.

We often read in such a way that some parts are analyzed more deeply and other parts more superficially. This can be expressed in a semantic network by representing a superficially scanned passage only by one node with some labels, e.g. the attribute "preliminary summary" can be used. On the other hand, an equally large passage, but more carefully studied, can be represented by a very large network.

Human reading is not only iterative, but also an evolving and dynamic process: comprehension of text (especially if it is difficult) increases gradually in precision and depth. If we consider reading only as a process which leads to an extension of the network, to creation of new nodes, edges, and hyperedges, we would ignore an important aspect: the appearance of "meta-levels" that reflect the text itself, various versions of subnetworks, alternatives, etc. Sometimes inadequate parts of the network are removed. Iterative, repeated reading leads to revisions (gradual refinement and modification) of the semantic network. In the reading process some passages can be interpreted alternatively. Each of these alternative interpretations is included in the semantic network.

Finalization encloses reading, completes incorporation of the (content of the) text into the network, usually accompanied by a summarizing (meta)commentary. It does not exclude returning to the text, and further reading in the future. Therefore, it is appropriate to speak about a preliminary finalization (and about a preliminary incorporation of the content into the network).

3.4 Writing

Writing is a very multifarious type of activity, we are interested first of all in writing as a task (Endnote 15), an organized process (in a group), which has the following key components well defined:

- intention,
- form, structure of the final product (outline),
- working plan,
- distribution of work, competencies, responsibility.

Of course, the portion of these individual components on the overall writing process is variable and may also change within the context of one project.

The above-mentioned "meta-level" characteristics (intention, overview, plan, organization) influence selection of relevant data, ideas from the knowledge base, and in this way also the whole writing process. Writing is usually interwoven with reading.

The objective of meaningful (creative) writing is either to give a realistic and, for the time being, missing (for a certain group of readers) characterization of a certain knowledge area, or to search for answers to questions which have not yet been answered.

For the first of the above-mentioned cases activation of a relevant subnetwork is crucial. Of course, during writing the activated (specified) subnetwork may be changing according to the links followed (edges and hyperedges in the network).

For the second case it is crucial that the system has the ability to formulate queries addressed to the network, and to find answers either directly or through inference. If we do not find the answer in this way, missing knowledge should be supplemented according to observation of reality (Endnote 16).

Concluding: during writing we gradually refine the network, enrich it with more and more edges (relationships among network elements), also with new nodes and newly activated (respectively also named) subnetworks. The knowledge base, is "better organized" after writing has finished. (Endnote 17)

We shall model the process of writing similarly to the process of reading (Endnote 18). It consists of three phases:

- initialization,
- iteration,
- finalization.

For *initialization* an *intention* is indispensable and usually there are available some preliminary texts (outlines, notes, early versions) as a base.

Iteration is an evolving and dynamic process. Formulations of local structures alternate with the design of a global text construction. This phase can proceed according to various strategies between two extremes:

- Top-Down – from structure to content,
- Bottom-Up – from content to structure.

Writing, of course, goes through multiple versions of text or its parts. Revisions are the key aspect of iteration between tuning of local structures and development of the global text. During the process of writing also the concept of relevancy is renovated and revised, thus subnetworks not relevant in the previous phases of writing are activated, and some are deactivated. Similarly, texts without interest in the previous phases are being read.

In parallel with changes in the text also changes of ideas occur. The changes are mirrored in the semantic network. With a help of the heterogeneous multilevel semantic network, the written text is put into relationship with related, connected ideas, but also with other texts already created, notes, outlines, etc. (Endnote 20)

Finalization produces a text which responds to (formal) standards of written text.

3.5 Collaborative Writing

One approach towards development of a *theory of collective intelligence* is presented by [14], where the author tries to define this concept as a form of computer-mediated collaboration.

In most organizations and companies there usually is a strict hierarchy of relationships between people. The people at lower levels of the hierarchy do not necessarily have a complete overview of the whole project. There are also working groups in certain environments where the relationships are less strictly hierarchical, and awareness of the global context is higher. According to Smith we can distinguish between *cooperation*, when the different individuals or groups are not required to know what goes on in the other parts of the project, and *collaboration*, where people communicate with each other and are aware of the work of their colleagues.

In cooperating teams persons responsible for the concrete tasks coordinate them and people performing the tasks do not need to communicate with each other very much as they can get instructions from the coordinating persons if they need. On the other hand, in collaborating groups communication dominates and coordination is more shared, depending on the problem to be solved.

Like Smith [14], we can distinguish three major forms of collaborative work – individual work on the artifact, collective work on the artifact, and conversations/discussions. Both synchronous and asynchronous actions are needed. The purpose of synchronous activities usually is to speed up solution of some (complex) problems or to solve some sensitive issues (where also non-verbal interactive communication can be employed).

If there is a strict hierarchical organization in a group where a person at an higher level instructs persons at a lower level, who do not communicate among each other, but just solve their tasks (cooperation), there is no new and interesting problem from our point of view. Such a situation can be modeled as one author with enhanced working capacity.

Our primary interest is in situations where a group of collaborating people solve a non-routine intellectual task and where problems to be solved are too large and too complex to be fully managed or understood by a (any) single person. Individuals in such a team can be experts coming from different areas. But their collaboration can be motivated by the importance complementary knowledge, but also by complementary skills. For example, one person can be a source of innovations and another may be good in performing functions. Such a collaboration oscillates between individual and collective behavior and uses both private and shared knowledge.

Smith [14] presents an attempt to sketch an image of collective intelligence. He intends to describe from the information-processing perspective describes a single abstract system consisting of both a human and a technological part. The primary components of this system are collective memory, collective processor, collective strategy, as well as collective awareness and control.

Collective memory as plausible analog or extrapolation of human memory, includes long-term and working memory. It contains both tangible (maintained in a computer system) and intangible (carried in the heads of the group members) knowledge.

A *collective processor* performs small-grained operations (develop, access, maintain information) in the group's memory. It is considered as a loosely coupled distributed system including multiple independent processors, joined by communications and social networks.

Collective strategy considers processes in relation to one another, in purposeful sequences. It enables coherence, because if the output is to be coherent, the group must work coherently.

Collective awareness and control focus on reaching a high level of integrity and consistency in a group's work if the task cannot be understood by one person. Development of partial, overlapping areas of knowledge can lead to a collective, distributed awareness. Also control has to be distributed and balances hierarchical authority with mechanisms that distribute responsibility.

There is an obvious correspondence between these components and our model of writing. The knowledge base and the text base can be viewed as two parts of memory, containing intangible and tangible knowledge. Operations on the knowledge base belong to the processes in the memory. Reading and writing use higher-level strategies composed from these operations. Concrete strategies depend on the collaborating authors and their objectives – the topic to be presented and the intended readers. The group determines its awareness and control by the employed style of work, ways of communication, distribution of responsibility, and comprehension of the topic.

We construct the common semantic network on the basis of the overlapping subnetworks. Each of the subnetworks corresponds to an individual semantic network. The knowledge base is in our model represented by a common semantic network where individual semantic networks overlap each other. The knowledge in the common semantic network represents the group's long-term memory.

A natural question is how individual semantic networks relate to each other, or, in other words, what is awareness of the group's long-term memory. Like Smith (1994) we can suppose that it includes a detailed awareness of particular segments of the knowledge base, a less detailed, but substantial, awareness of adjacent parts of the base of ideas, and a basic awareness of the whole knowledge base. This can be reached by the introduction of a *slow build/fast review* cycle, which is based on the assumption that the level of awareness and expertise required to generate a segment is significantly greater than that required for another person to understand it. The collaborative process based on these cycles is in our model reflected by enlargement of individual semantic networks and overlapping areas, which means deeper awareness of particular segments as well as greater awareness of adjacent parts.

An essential feature of collaborative expository writing is the objective to present new ideas to readers. The authors are aware of the knowledge level of potential readers, so it can be represented by another semantic network in our model. The aim of the authors is to write such a text that its reading will continually enhance the reader's semantic network in the way that in the end it includes the new knowledge.

If the knowledge base represented by the common semantic network consists of the intersection of individual semantic networks, there rises the severe problem how to maintain consistency, coherence, and integrity of the knowledge base. The basic requirement is that of consistency, so the question is how to solve inconsistencies between individual semantic networks, if they appear. We have already mentioned that our ambition is not to perform human intellectual activities, like decision or generation of ideas, by a machine. Our aim is to support these activities by suitable means. If a conflict in a collaborating group occurs, argumentation and social mediation should be the best approach to resolve it.

If final documents created by a group of collaborating people are to be elegant, the authors have to guard their coherence and integrity. The lesser individual semantic networks are related to each other, the more difficult the task. For example, two mathematicians, though from different areas of mathematics, are in general able to produce a coherent article more easily than a knowledge engineer and a physician. The reason is that in the first case the individual semantic networks more strongly overlap.

There is a problem of the equivalence of subnetworks. If two private semantic networks are equivalent in depth, it is easier to reach consistency and coherence by joining them, even if they are superficially different. The opposite case, if the private networks are superficially similar, but different in depth, is the most complicated one for establishing a consensus. In any case the persons have to communicate with each other. Each of them evaluates the text produced by the partner from his or her own point of view until a consensus is reached.

3.6 Conclusion

The kernel of our model is the idea of heterogeneous multi-layered semantic network. The processes of reading and writing are composed of the operations on the network. Similarly, collaborative writing is explained over a structured and evolving complex of private and common subnetworks. Important facilities provided by the network are: an ability to focus attention, to change the level of abstraction and detail, and a crucial ability of distinguishing relevant (irrelevant) ideas in a knowledge base.

In the next section we proceed to a lower level of abstraction. We want to build a conceptual environment in which the cognitive model can be transformed into a computer-based system supporting collaborative writing. The aim of the system is not an automatization of writing, but the provision of some tools to support natural human writing.

We characterize the level of the next chapter as a conceptual model (Endnote 21). Its main role is to provide some data structures (a data model) and a control structure appropriate for the design of the system. Our aim is to preserve the basic features of our cognitive model in terms of both the data structures and the control structure.

4 Conceptual Model

In the previous section the basic features of our cognitive model were described. Now our exposition will proceed in terms that are closer to the implementation level, although we shall not describe implementation itself. The chosen concepts and level are useful from the viewpoint of a more detailed and precise description of the conceptual model.

A *document* concept plays the role of the basic data structure. We use it in a very general sense. Each object we will consider is a document.

The document consists of a head and a body. The head consists of some attributes and their values. The body is a text. Therefore, objects of the knowledge base and of the text base can be represented as documents.

The complexity of the head, and the size of the body vary from one document type to another. We note that relationships among documents can be modeled by using documents of a specified type, the point lies in the appropriate choice of attributes. A broad class of activities can be modeled as document processing.

Effective and flexible organization of documents determines the representation of a user's mental model, as well as the way members of a working group can respond to each other. To understand meaning it is necessary to understand coherence, as expressed by relationships between (documents. This is why their organization is so important. The system has to capture both relationships between documents, as well as their internal structure.

The main goal of this section is to describe in detail a way to specify relationships which form the basis of the semantic representation introduced in Section 3. Among the significant properties of semantic networks useful for our model were:

- multiple viewpoints,
- varying degrees of detail,
- an approximation of relevancy.

Synchronous and asynchronous communication were distinguished in the discussion of collaborative writing. We restrict our attention to asynchronous communication in this section. Nevertheless the synchronous communication is an important topic for the future development of our conceptual model (and its implementation).

4.1 Documents

Each document is characterized by a set of attribute values. We distinguish between obligatory and optional attributes. The values of obligatory attributes are written automatically by the system called MONSTER (with one exception, see below). The basic features of MONSTER are described below in Subsection 4.2. The values of optional attributes are written by a user. The *obligatory* attributes are:

- document identifier,
- date (of the document creation),
- owner,
- time stamp – a pair of the form *(date:time, identifier of the last action on the document)*,
- history (values are sequences of time stamps),
- context priority (it will play a role in management of relevancy – Endnote 22),

- document content – a text of unlimited length. (Endnote 23)

The *optional* attributes are either attributes defined (offered) by MONSTER (we shall call them *system* attributes) or *user-defined* attributes (each individual can define new attributes of her/his documents according to her/his needs and wishes).

The attribute values may be either atomic or complex (composed of values of more elementary attributes). System attributes:

- document name,
- author(s) – a set of values is expected in the general case (the value of this attribute may be different from the *owner* of the document),
- abstract,
- references,
- author address(es) (values are complex objects structured in a standard way – phone, fax and e-mail included),
- document structure (a tree of values – sections and sub...subsections of the document),
- access rights (a general value is a list of lists of the form *(Read,subgroup), (Write,subgroup), (Delete,subgroup))*,
- version – a member of a partially ordered set of values is expected (it is possible to create own domains of the version attribute, the decision of what types of versions are distinguished is delegated to the user),
- priority – a value from a partially ordered set (the importance of the document from the users point of view),
- message target (values are complex objects with the usual structure *(e-mail-address,subject,cc))*,
- type (the domain of this attribute is a dynamic one; a flexible, open set of possible values is considered; some examples of possible values are listed in the next section),
- key words,
- identifiers of alternative documents (in the case of alternative documents a pointer to the alternatives is maintained; this attribute is useful e.g. for groups involved in design),
- state (possible values: finished, under development).

Examples of possible values of the *type* atribute:

- first idea,
- note,
- draft,
- outline,
- working material,

- work,
- paper (some types are subtypes of other types – a paper is a special kind of a work),
- book,
- message (the attribute "target" must not have null values within this type of document; in general, for some types of documents it is necessary that some of the optional attributes should not have null values),
- publication,
- quotation,
- remark,
- overview,
- plan,
- calendar (there are calendars for individuals, for subgroups, and for the whole group – they contain the results of the monitoring of actions; values are sequences of tuples of the form *(date:time, identifier of the document, identifier of the individual, identifier of the current subgroup(s), identifier of the last action on the document)*),
- database,
- query,
- directory,
- bibliography,
- thesaurus,
- vocabulary,
- program,
- graph,
- list,
- map,
- contents,
- index,
- ...

When an individual creates a document she/he can choose a subset of the system attributes according to her/his intention and need. It is possible to distinguish various groups of documents. One way is based on the values of the attribute *type*. The other is based on a classification of documents according to the values of the attributes defined for the document. From this point of view the more general document (and the more simple, default kind of document) is one with no optional attributes. Over the default type (we call it *first idea*) it is possible to build (in an obvious way) a partially ordered hierarchy of document types with defined sets of optional attributes. When an individual records his first idea about some topic, no special work is needed except of expressing the idea (if she/he uses the default kind of document). In this case the only task is to create the content of the document (to record a value of the attribute *content*). With increasing distance from (in the above-mentioned hierarchy) from the default document (first idea), more work is needed on the attributes of the document. The benefits of this supplementary work are clear: an individual gains an increased power in creating (implicit) relations among documents (based on the equal values – Endnote 24 – of the attributes) and therefore also in accessing relevant documents when needed. Moreover, a rich structure of document attributes allows users to define useful standards of documents and to create a support for a systematic way of work. This feature can positively influence discipline and productivity while working with MONSTER.

4.2 MONSTER

Document, the only data structure used in the conceptual model, corresponds to the objects of the knowledge base and the text base of the cognitive model. Now, we should discuss in more detail the operations on semantic networks, interconnections between the knowledge base and the text base, management of relevancy, and of asynchronous communication. These aspects of the conceptual model are realized by MONSTER. Intuitively, the purpose of MONSTER is to prepare the environment for human thinking (for effective, comfortable, and rational use of documents). It should play the role of an integrating component of the modeled environment. Maybe, the reader will associate MONSTER as it is described in the following paragraphs with an computational environment. But this is not the only interpretation. We prefer a more general approach treating MONSTER as a tool (of an arbitrary kind) useful for dealing with a large number of documents. In order to understand the role of MONSTER it is important to emphasize the dynamic point of view. Until now our attention was oriented to a local description: a document with a given set of attribute values. If we take a global point of view we can register an evolution of documents. Some of the documents are output documents. A group (or an individual) presents them as the result of its activity. From the MONSTER point of view the path to the output document is important: An individual (or a group of individuals) reads and writes various documents. Some of them are in a strict sense input (read-only) documents. Other documents can be denoted as running documents – they are stepwise modified, their status is changing (we distinguish between a local modification and a version change), a part of their content is moved into another document, the documents are made accessible to other individuals, more individuals can write and read the same document. Some documents are devoted to others (commentaries, summaries, alternatives, messages etc.). The task of MONSTER is in a certain sense to support a stepwise evolution (revision) of proposals, brain-waves, notions, and ideas. The ideology behind this formulation is as follows: If we take no account of brilliant ideas, the quality of intellectual work is dependent on careful and concentrated revision of initial proposals. An important task performed by MONSTER is the support of individuals in their work with documents as to effectivity, comfort, and rationality. One of the main goals of MONSTER consists in increasing the efficiency of the work with documents via support of access to the relevant documents. The main idea of MONSTER is that it is not limited to straightforward help in reading and writing, but that it also helps in deciding what is relevant to read, and where it is relevant to write. At each moment MONSTER shows which (relevant) documents an individual can see. Direct access is for example to those documents with which the individual has worked in the last session (it means that the working atmosphere of the last session is immediately recalled and a fluent continuation of the work is supported), to the personal plan or to the plan of the group. At a given moment the relevant documents (Endnote 25) are the documents with maximal context priority. The context priorities for a given individual are dynamic. They are changed by MONSTER depending on the context, more precisely on

- plan,
- calendar,
- current actions of an individual and its partners,
- user model,
- priority of the document (user defined),
- answer to the last query.

However, our model does not provide simple instructions for the computation of context priorities (for example of the form: *if plan and calendar are in state S, then the context priority is P*).

The computation is rather a result of activities of agents (demons), permanently active processes that can react if and when required. The agents respond to some events and contribute in a certain way to a change of the context priorities. There is a given set of observed event types and one agent watches one event type. The values of context priorities are from a partially ordered set with the top element. We can realize them as n-tuples, where the number of components (n) is determined by the number of observed event types. We suppose that the component values are linearly ordered (with a maximal element).

Let $\phi = (x_1, ..., x_n)$ and $\psi = (y_1, ..., y_n)$ be n-tuples. Then $\phi \leq \psi$ iff $x_i < y_i$ for all components where does not hold $x_i = y_i$. We suppose that the construction of a specialized agent (Endnote 26) analyzing (and reacting to) events is simpler, and above all, more modifiable (when unsatisfactory results are reached in experiments) than the eventual construction of a procedure which on the basis of a synthesis of all events computes a context priority.

We suppose further that results of various agents can be mutually conflicting, other specialized agents are then responsible for handling the conflicts. Some examples of agents' behaviour (Endnote 27):

- agent $A0$ (supervisor): sets the top value in the case of some events (see below); provides a strategy for choosing (promising) documents, whenever there is no preference between maximal context priorities,
- agent $A1$: increases the first component of the value to its maximum and sends a message to the supervisor, whenever according to the plan a document should be finished today or earlier in case it is not yet finished now; or if a document should be finished soon (Endnote 28).
- agent $A2$: sets the maximal value of the second component and sends a message to the supervisor, whenever according to the calendar the last session of the current individual was devoted to the document D;...
- further agents watch further components which influence the context priority as mentioned above

5 Conclusion

The implementation of the proposed system is intended to take account of the following aspects. A *gradual formalization of knowledge*. Hypertext can be used as a mediating "semi-formal" representation [9], enabling transformation of informal information (typically in the form of text) into formal structures (semantic networks, frames), which can be processed by a machine to generate new knowledge and to provide sophisticated search facilities. *Integration of browsing and searching techniques*. It should provide efficient access to relevant information [11]. Combination of these two techniques is quite natural for us. Users often do not know exactly what they are looking for or they are not able to specify their requirements precisely. In these situations iterative use of searching and browsing can improve effectiveness of the access to information. Our main challenge is development of a system that would be able to adapt the content, structure, and presentation of information to the current context, as well as user needs and level of knowledge. Systems of this type are called *adaptive hypertext systems* [2]. We plan to use the HyperWave (former Hyper-G) system [12] as the platform for such a system.

Further implementation details, like considered document types and basic architectural features, can be found in our previous paper [10]. Their main objectives were processing of complex inter-object relationships and efficient access to relevant information. To achieve this, an integration of several information technologies is needed, mainly hypertext, information retrieval, knowledge-based systems, and groupware.

6 Endnotes

1. We do not propose the semantic network as a structure used by a software system, but we do also not exclude such a use. The

described model provides only a general framework for representing the ideas of a person or of a group. The main intended application of the ideas is writing and reading. Special interest is devoted to the revisions of the knowledge base caused by reading and writing. The instantiations of the proposed abstract structure are context- and situation dependent.

2. An efficient mechanism for formal diagrammatic representation of semantic networks are *higraphs* [7], which integrate two types of relationships in topological diagrammatic formalisms. These are binary relations represented in graphs and structural relationships among sets known from Venn diagrams. Additionally higraphs can also represent the Cartesian product. A drawback of this formalism is its inability to specify both set inclusion and set membership. Application of higraphs can be especially useful in specification and design of reactive systems, that are event driven. A major problem of these systems is in the difficulty to describe reactive behavior in a way that is clear and realistic and also formal and rigorous. Enhancement of standard state diagrams based on higraphs is called *statecharts* and brings an opportunity to represent depth, orthogonality (unordered Cartesian product), and broadcast communication (actions on transitions). These properties are of interest to us because they enable the modelling of spreading activation when new knowledge enters the system. This is also supposed to be the way the contents of working memory can be activated in the long term memory of a writer.

3. Among the optional cases there may be, for example: place, time, way, the robbed person etc. An important remark: we are not aiming at the linguistic adequacy here.

4. The *hypergraph* concept is defined by Berge, see [3].

5. Readers familiar with the idea of frame representation will see that the set of attributes associated with a node, edge, or hyperedge can be expressed by a frame. We do not consider it as a deficiency of our representation. A goal of the concept of semantic network as defined here, is its generality and flexibility. Therefore, the capability of frame representation is a welcome property for our representation formalism.

   Similarly, we can imagine a semantic network connecting the nodes associated with a person and with a book by an edge labeled *author*. Nothing prevents us to introduce such a choice. We again emphasize that this only is useful because it leads to a greater flexibility and generality, as the result of the possibility to assign a set of attributes to each construct used in the semantic network.

6. With extreme cases of zero or one label.

7. It is assumed that components are labeled, too.

8. The adjective "generalized" will be usually dropped out.

9. The two last mentioned operations suggest that attributes are subject of updates, too.
10. A language for the expression of conditions is not a subject of our interest now.
11. We assume that the notion of a sentence in the language of semantic networks is defined – we consider such a task to be realizable.
12. It is possible to specify the elementary retrieval operations in a query language. We will study specification of inference over heterogeneous multi-layered semantic networks by using a query language in a future paper.
13. Some remarks to avoid eventual misunderstandings: We pay attention to reading and writing of explanatory (not e.g. artistic) texts.

    We do not focus on lexical and syntactic aspects of reading, neither on natural-language understanding. The domain of our model is the process following comprehension of a linguistic object (we realize that the border is not sharp and one can distinguish various levels of comprehension, understanding). But we assume that we are able to diagnoste the reason for text misunderstanding – we can decide whether it is unfamiliarity with the language, or with the area being described. So, our approach is based on indisputable intuitions: the better knowledge is read (and concepts are used), in correspondence with the knowledge and conceptual equipment of a reader, the better is comprehension of the text being read.
14. To understand coherence between sentences, and the global sense of the text.
15. At the beginning several basic notions. The following formulations are not to be definitions, just contexts, in which the introduced terms appear in a significant manner. We consider these terms as satisfactorily intuitive, their definition could end up in the way that is common in similar situations, when people try to define well-known terms: relatively clear is defined by less clear. The *task* is writing which was initialized also on the basis of some author's (authors') commitments with respect to the outside environment. *Project* is a complex of activities performed by a group of people to fulfill a task. We assume that the notions as *intention* and *structure* (of a text) do not need comment.
16. The objective of this model is not to explain creativity and discovery. The ambition is – while talking about writing – just to catch the transformation of a knowledge base state into a new text. The modification of the knowledge base can lead to creative act or discovery as an auxillary benefit.
17. From the psychological point of view the case is of interest when this order or organization is obtained by getting an insight, a bird's eye view, a new hypothesis, which surprisingly brings order into what previously seemed chaos.

18. Of course, there are many other possibilities, but we choose for structural homogeneity.
19. The author(s) judge that the overall scheme and prepared details fir well together if the facts are confirmed, one thing falls in with another, the whole makes sense.
20. In this context we may not forget that there is no straightforward and linear coherence between written text and a knowledge base. Text is a cultural creation and depends on various incidental influences. Sometimes there is no idea behind a text. Cliches and often-used formulations may enter into a text.
21. We speak about conceptual model in order to emphasize an intermediate position between the cognitive model and the specification of an implementation.
22. The context priority can have the null value. The role of this attribute will become clear later.
23. The values are written by the user.
24. A network of connections is created over the attribute values.
25. The way MONSTER determines the relevance of documents is a basis for the user interface of MONSTER. The user interface is conceived as a medium for rationality, and relevance is one of the most important aspects of rationality.
26. We can call the corresponding user interface also a 'user interface based on a society of agents'.
27. At present our attention is not devoted to the details of this problem.
28. The nature of the rules presented is clear, they are rules of thumb. Of course, the only way to improve them is by gradually changing and choosing of the rules, and testing the result.

References

[1] Allen, J. (1987), *Natural Language Understanding*, The Benjamin/Cummings Publishing Company, Inc.

[2] Beaumont, I., Brusilovsky, P. (1995), *Adaptive Educational Hypermedia: From Ideas to Real Systems*, in Proceedings of ED-MEDIA 95 – World Conference on Educational Multimedia and Hypermedia, 93-98.

[3] Berge, C. (1989), *Hypergraphs: Combinatorics of Finite Sets*, North-Holland.

[4] Brachman, R.J. (1979), *On the Epistemological Status of Semantic Networks, in Associative Networks: Representation and Use of Knowledge by Computers*, 3-50, ed. by N.V.Findler, New York, Academic Press.

[5] Fillmore, C. (1968), *The Case for Case*, in Universals in Linguistic Theory, E.Bach and R.Harms (eds.), Holt, Rinerhart and Winston, New York.

[6] Flower, L.S., Hayes, J.R. (1984), *Images, plans, and prose: The representation of meaning in writing*, Written Communication, 1, 120-160.

[7] Harel, D. (1988), *On Visual Formalisms*, Communications of the ACM, 31/5, 514-530.

[8] Hayes, J.R., Flower, L.S. (1980), *Identifying the Organization of Writing Processes*, in L.W. Gregg, E.R. Steinberg (Eds.), Cognitive processes in writing (pp. 3-30). Hillsdale, NJ: Lawrence Erlbaum Ass.

[9] Kaindl, H., Snaprud, M. (1991), *Hypertext and Structured Object Representation: a Unifying View*, in Hypertext '91 Proceedings, 345-358.

[10] Kirchheimova, H., Kravcik, M., Sefranek, J. (1994), *MONSTER: An Environment for Cooperative Reading and Writing*, in Proceedings of the First Interdisciplinary Workshop on Cognitive Modelling & User Interface Development, 49-64.

[11] Lucarella, D. (1990), *A Model for Hypertext Based Information Retrieval*, in Proceedings of the ECHT'90 European Conference on Hypertext, 81-94.

[12] Maurer, H. (1996), *HyperWave: The next generation web solution*, Addison-Wesley, 1996.

[13] Rada, R. (1991), *Hypertext: from Text to Expertext*, McGraw-Hill.

[14] Smith, J.B. (1994), *Collective Intelligence in Computer Based Collaboration*, Lawrence Erlbaum Associates, 1994.

[15] Spiro, R.J., Jehng, J.-Ch. (1990), *Cognitive Flexibility and Hypertext: Theory and Technology for the Nonlinear and Mutlidimensional Traversal of Complex Subject Matter*, in D. Nix, R. Spiro (Eds.), Cognition, Education, and Multimedia: Exploring Ideas in High Technology (pp. 163-204). Hillsdale, NJ:Lawrence Erlbaum Associates.

[16] Stein, L.A. (1992), *Resolving Ambiguity in Non-Monotonic Inheritance Hierarchies*, Artificial Intelligence 55, 259-310.

[17] Sefranek, J. (1996), *Knowledge Representation and Inference*, to appear.