# A linguistic and navigational knowledge approach to text navigation

Javier Couto, Jean-Luc Minel

# A linguistic and navigational knowledge approach to text navigation

**Javier Couto**
Facultad de Ingeniería - UdelaR
Julio Herrera y Reissig 565
11300 – Montevideo – Uruguay
jcouto@fing.edu.uy

**Jean-Luc Minel**
MoDyCO, UMR 7114 CNRS – Univ. Paris X
200, avenue de la République
92 001 – Nanterre – France
jminel@u-paris10.fr

## Abstract

We present an approach to text navigation conceived as a cognitive process exploiting linguistic information present in texts. We claim that the navigational knowledge involved in this process can be modeled in a declarative way with the *Sextant* language. Since *Sextant* refers exhaustively to specific linguistic phenomena, we have defined a customized text representation. These different components are implemented in the text navigation system *NaviTexte*. Two applications of *NaviTexte* are described.

## 1 Introduction

Text navigation has several interpretations. Usually, this term refers to hypertext systems, which offer the possibility to activate hyperlinks, moving the reading point from a text unit (*source*) to another one (*target*), this change being intra or intertextual. From our point of view, this conception presents some limitations. First, the hyperlink activation is not assisted. In other words, imprecise, poor or no information is provided to the reader before s/he activates the link. Second, the reader does not know where the movement will be carried out in the text (before or after the reading point or outside the text), which generates the "lost in hyperspace" problem (Edwards and Hardman 1989). Finally, hyperlinks are embedded in the hypertext. Therefore, there is no clearly distinction between text constituents and navigation knowledge. In addition, by not explicitly modeling this knowledge, it is not reusable.

Different solutions have been proposed to address the problems mentioned. Some researchers (Danielson, 2002) have tried to mitigate the lost in hyperspace problem offering global maps where the reading point is clearly identified. Adaptive hypertext (Mathe and Chen, 1994; Brusilovsky, 1996) relying on user model, proposes to modify the way the text is shown on the screen. Dynamic hypertext (Bodner and Chignell, 1999) computes the value of hyperlinks using several criteria such as text similarity or predefined relations. In this approach, a hyperlink is defined as a query returning a text node.

In some way, our conception of text navigation is related to the notion of computed query, but rather than taking into account criteria depending on the reader, the target is computed by exploiting linguistic information in texts. Moreover, the queries are not placed in texts but they are encapsulated as knowledge by a specific language (Sextant), which allows isolating the navigational knowledge to create knowledge bases. Both texts and queries (navigational knowledge) are interpreted by NaviTexte, which manages the interactions with a reader.

The remainder of this paper is organized as follows. In the next section, we discuss our approach to text navigation. The third section describes a navigational knowledge modeling language called Sextant. The fourth section details the text navigation system NaviTexte. The fifth section describes two applications of NaviTexte. Then we address the evaluation aspects of our approach. At last, conclusions are presented.

## 2 Defining text navigation

Our conception of text navigation lies in the hypothesis that navigating through texts is the expression of a cognitive process related to specific knowledge (Minel, 2003; Couto and Minel, 2006). More precisely: we claim that a reader moves through texts applying some knowledge to exploit linguistic information present in texts (*e.g.* discursive markers). Moreover, we claim that this knowledge may be articulated in a declarative way (*cf.* Sec. 3) relying on information in texts coded, on the one hand, by its structure, and, on the other hand, by specific annotations.

The main difference between classic hypertext and our conception of navigation lies on the status of texts and on the definition of navigational knowledge. In the case of hypertext, the visualization of a text is unique and navigational knowledge is encoded (embedded) in the text. In our approach, there are several ways to visualize a text (Couto, 2006), each way called *text view*, and for each view, different navigational knowledge may be defined. As a consequence, the navigation is not guided by the author, compared to hypertext navigation where s/he determines the links, but it is the result of an interpretation process made by the reader, relying on text structure and annotations.

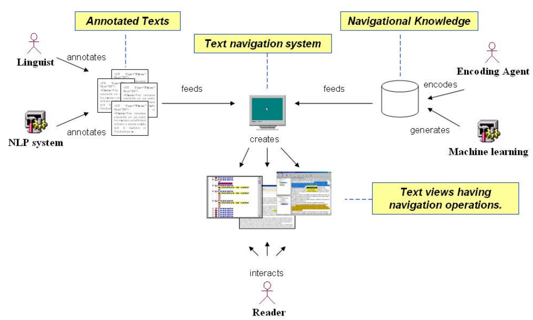


Figure 1. Elements of text navigation.

Our conception of navigation (*cf.* Fig.1) relies on four elements: i) a text representation allowing linguistic specific phenomena and annotations (see Couto, 2006); ii) a language to model navigational knowledge (*cf.* Sec. 3); iii) an agent (an individual or a software) able to encode such knowledge; iv) a system, called NaviTexte, to interpret and apply knowledge to a specific text (*cf.* Sec. 4).

## 3 Modeling navigational knowledge: the Sextant language

To allow the unambiguous isolation of navigational knowledge we need a formal modeling language. We want to model the knowledge applied by a reader to move through texts, claiming that this knowledge exploits linguistic information present in texts. We do not say that this is the only way a reader may move through texts (*e.g.* strolling courses proposed by Géry (2002) is a counterexample), but we say that this is the kind of way than we are going to model. Different views of a text and the fact that each view contains specific indications of the possible reading courses constitute the heart of the language.

### 3.1 Knowledge modules ant text views

A text view may be a *full* view or a *partial* view focused in some specific phenomena present in the text (for example a view of all discourse referents). The constituent elements of a view are formalized in a *view description*, which contains the type of view, its parameters, the creation constraints (*i.e.* conditions to verify by the TU of the view) and the navigation operations (see next section). At present, four types of view have been defined: *plain-text*, *tree*, *graph* and *temporality*. The three firsts types are described in (Couto, 2006). The last one graphically represents temporality in texts and a complete description is in (Battistelli *et al.*, 2006).

Several view descriptions may be gathered by the encoder in an entity called *navigational knowledge module*. The creation of a view may be conceptualized as the application of a view description to a specific text. Thus, the application of a module implies the creation of a set of text views.

### 3.2 The navigation operation

The notion of computed query mentioned in section 1 is formalized in Sextant as a *navigation operation*, which links a *source TU* to a *target TU*. In classic hypertext systems one must explicitly connect the specific source to the specific target. For example, if a reader wants, for all definitions in a scientific paper, to move from one to the following one, several hyperlinks must be defined. In our approach we specify the source and the target using conditions. As a result, we can abstract, for example, the navigational knowledge that states "go from one definition to the following one", being "definition" one of the TU annotations.

We can specify several conditions for the source and the target. We say that a navigation operation is *available* for a TU if this TU verifies the source conditions. A text navigation system should find the TU that verifies the target conditions. As several TU in the text may verify them, we need a way of disambiguation. This is done by the *orientation* parameter, which specifies the direction of the target search by using one of these options: *first*, *last*, *forward(i)*, *backward(i)*. *First* and *last* indicate that the search of the target is absolute: the TU to select will be the first (respectively the last) TU that verify the conditions. *Forward(i)* and *backward(i)* indicate that the search is carried out relatively to the source (before or after) and indexed by the integer *i*. For example, “forward(3)” is interpreted as the third TU, after the source, of all the TU verifying the target conditions.

### 3.3 The conditions language

The conditions language is an important component of Sextant ant it is composed by *basic conditions*, *TU elements existence conditions*, *hierarchical conditions* and *non-hierarchical conditions.*

*Basic conditions* concern TU’s attributes and annotations. For this kind of condition we use a notation close to the pattern notion. We define an operator called TU, having five operands that correspond to the following properties: *type*, *number*, *level*, *annotations* and *string*. With the three first operands and the fifth one, we denote constraints of equality, inequality, order, prefix, suffix and substring occurrence. The fourth operand is used to indicate the existence or non-existence of annotations, whether it is an annotation name, a value or a name-value pair.

For *TU elements existence conditions*, we define operators without operands to verify if a TU has *annotations*, *string*, *title*, *parent* and *children*.

For conditions dealing with *hierarchical relationship* between different TU, a set of unary operators have been defined, taking a basic condition as an argument. For example, the *isAscendant* operator verifies if a TU is the ascendant of another TU specified by a basic condition. The other operators are: *isParent*, *isChild*, *isSibling*, *isDescendant*, *hasInTitle*, *isInTitle*. We would like to draw attention to the fact that these operators allow to move through the hierarchy of TU from a starting TU (Couto, 2006).

*Non-hierarchical conditions* concern constructed units’ attributes and annotations as well as TU constitution.

All conditions may be combined using the classic logic operators OR, AND and NOT. Figure 2 presents an example of a language expression that formulates the following condition: TU of type “NP”, having an annotation of name “discourse referent”, for which it exists, among its descendants, a TU of type “paragraph” not having an annotation of name “semantic label” whose value is “conclusion”.

**TU**(type = NP, *,*,{(discourse referent, *)},*) **AND**
**isDescendant**(**TU**(type = paragraphe,*,*, {¬∃̣(semantic label, conclusion)},*))

Figure 2. Conditions language example.

This condition means: noun phrases being a discourse referent that does not occur in a concluding paragraph.

## 4 NaviTexte: a text navigation system

Several adaptive navigation systems have been proposed (Benyon and Murray, 1993; Kaplan *et al.*, 1993; Boyle and Encarnacion, 1994; Brusilovsky and Pesin, 1994; Brusilovsky *et al.*, 1996). While they are goal specific (learning, tutoring, reading, etc.), NaviTexte (Couto, 2006) is a generic text navigation system implementing our approach. This means that, depending on texts and knowledge modules, NaviTexte may be used, for example, as a learning, tutoring or reading system. Another important difference is that NaviTexte gives the user the liberty to navigate through the text following its own interests (the system propose - the reader chooses), while the mentioned systems try to maintain a user stuck to a given route (the user chooses - the system propose) (Höök and Svensson, 1999).

NaviTexte consists of sub-systems dealing with: *text representation*, *navigational knowledge*, *visual representation* and *user interaction*. The first one builds a text representation in memory from a text annotated manually or by dedicated software (Cunningham *et al.*, 2002; Bilhaut *et al.*, 2003). The second sub-system loads and compiles the knowledge modules. The result of this compilation is a graph of potential navigation courses that in practice is calculated as needed and stored in optimization data structures. The third sub-system calculates and displays different text views and the fourth one manages the user interaction.

The reader has the possibility to load and unload several texts and knowledge modules in the same work session. A complete description of NaviTexte may be found in (Couto, 2006).

## 5 Applications of NaviTexte

Building an application with NaviTexte requires a set of texts and navigational knowledge modules. Both text representation and Sextant language have XML implementations with dedicated editors to use in case of a manual text annotation and a human knowledge encoder, respectively (*cf.* Fig.1).

So far four applications have been developed: *alternative automatic summarization* (Couto and Minel, 2006), *the NaviLire project* (Couto *et al.*, 2005; Lunquist *et al.*, 2006), *re-reading Madame Bovary* (Couto and Minel, 2006) and *temporality in texts* (Battistelli *et al.*, 2006). We present two of them to illustrate NaviTexte's potential.

### 5.1 NaviLire: a text linguistics application

For the past thirty years, text linguistic researchers have worked on describing linguistic markers of textual coherence in order to bring out principles of text structuring (Lundquist, 1980). A set of concepts and models of textual interpretation has been worked out, including for example, anaphora, connectors, mental spaces, etc. In particular, these studies have shown that even for languages apparently close like French and Danish, texts are not organized in the same way (Lundquist, 2005). Consequently, text linguistics has important implications in foreign language teaching, especially from a contrastive point of view, when language pairs are analyzed through texts used in authentic communication situations. It seems that the teaching of text linguistics contributes to sharpen the attention of students towards the building of well-formed texts and to stimulate their own text production. Therefore, a tool that allows the student to perceive text units that contribute to and maintain text coherence and to navigate between them, can be supposed to be an important didactic tool for teaching reading of foreign language texts, as well as producing written texts in the foreign language.

In the reading process, the student has to deal with two basic types of cognitive problems. First, s/he has to identify discourse referents in a text and choose the correct relations between the noun phrases that refer to them. Second, s/he has to identify the function and orientation intended by the sender. In the NaviLire project, navigation operations assisting the student are defined used Sextant and the texts are manually annotated by a text linguistics expert.

### 5.2 Navigation as an alternative to automatic summarization

Many automatic summarization systems have been proposed (Mani, 2001; Minel, 2003). All these systems, based on the principle of phrase, proposition or group extraction, have been confronted to two problems intrinsic to the extraction procedure: i) the rupture of text cohesion, like in cases of anaphora where the corresponding discourse referent is missing; ii) the adaptation of the summary to reader specific needs. Actually, there are no completely satisfying solutions to these problems. An alternative approach is to consider the summarizing process as a reading course belonging to the reader (Minel, 2002; Crispino and Couto, 2004). Thereby, instead of extracting text fragments, we propose specific reading courses, whose specifications are based on propositions of (Endres-Niggermeyer *et al.*, 1995) and on practice observations made in the frame of the automatic summarization system SERAPHIN evaluation (Minel *et al.*, 1997) and the Filtext framework (Minel *et al.*, 2001).

These works showed that professional summarizers are interested by discourse categories that they retrieve by exploiting text organization and lexical markers. They also showed that these professionals navigate through texts using heuristics acquired by experience. For example, they begin by reading the conclusion, and then they continue by looking, in the introduction, for nominal groups that occurred in the conclusion. This is the knowledge we have modeled with Sextant.

A specific reading course specifies, on the one hand, the kind of discursive category searched by a reader (*e.g.* a conclusion, a definition, an argument, a hypothesis, etc.[1]) and on the other hand, the course in which the segments that linguistically enunciate these categories (typically phrases) must be presented to the reader.

To carry out these reading courses, it is necessary to locate the discursive categories involved and mark them in the text. For this purpose, we used ContextO (Minel *et al.*, 2001). A reading course example is presented in Fig. 3. The reading point is positioned over the first TU, a phrase in this case, annotated "Thematic Announcement". When the user clicks over the TU, NaviTexte recommends her/him four navigation operations. The first one suggests bringing her/him to the following "Thematic Announcement". The others ones suggest going to the first "Conclusion", the first "Recapitulation" and the first "Argument". For a given TU, each navigation operation available has three

[1]For more details on different categories or on what empirical basis were these types derived, see (Minel et al., 2001).

possible states (and there is a visual semantic associated to states), depending if it has been executed (bold font and "*" prefix) or not (normal font and no prefix), and if it exists a TU target (clickable menu option) or not (non-clickable menu option).

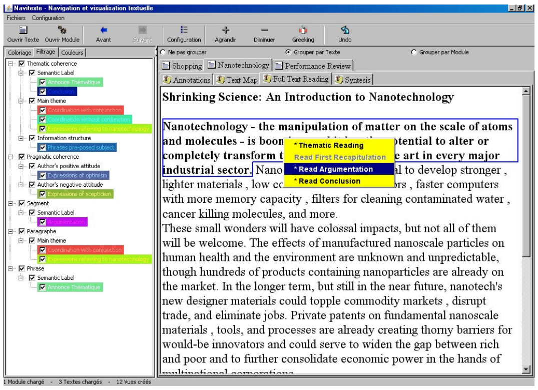


Figure 3. Automatic summarization courses.

These kinds of suggestions (*i.e.* showing available navigation operations for a TU) are made all over the reading process. Consequently, along her/his reading continuum, the reader is assisted by the display of a specific set of signs, and there is no rupture of cohesion because s/he is able to continually see all the text (Battistelli and Minel, 2006).

## 6 Evaluations

There are few studies of adaptive navigation in hypermedia systems and most of them are focused in measures such as the number of visited nodes or the task completion time (Höök and Svensson, 1999). Are we interested in this kind of measures? Being NaviTexte a generic text navigation system, we think that what it has to be evaluated are the different applications. Each application requires pertinent measures. For example, in NaviLire, the number of nodes or the time factor seems less useful that the comprehension of the text analyzed.

So far, NaviLire has been put into practice on a small scale only, viz. in the teaching of French texts and text linguistics to Danish language students in the 4th year of Language and Communication studies at the Copenhagen Business School. A pilot experiment was carried out in order to evaluate the effects of using the program.

The first results are based on forty answers, of which 35 concern questions about the content of the text. These results show that the *navilistes* (people using NaviLire) have a better comprehension performance than the *papiristes* (people using paper and pencil) for 14 questions, an identical performance for 16 other questions, and a poorer performance for 5 questions (*cf.* Table 1).

| | #questions | % |
|---|---|---|
| *Navilistes* better than *Papiristes* | 14 | 40 |
| *Navilistes* the same as *Papiristes* | 16 | 45,7 |
| *Navilistes* worse than *Papiristes* | 5 | 14,3 |
| **Total** | 35 | 100 |

Table 1. Comparison of *navilistes* and *papiristes* (Lundquist *et al.*, 2006)

Evaluations of the alternative automatic summarization approach are ongoing. Our main problem is that automatic summarization evaluations, well known as difficult to carry out, typically compare to summaries made by professional summarizers (Mani, 2001; Minel, 2003). On the one hand, since we do not create a summary, we do not have an *object* to compare. On the other hand, since we have modeled the professional heuristics, we cannot compare the behavior of our system to theirs because it is exactly what it has been modeled.

## 7 Conclusions and future work

We have presented our approach to text navigation conceived as a cognitive process that exploits linguistic information present in texts. We have defined it and explained the main differences with the hypertext navigation approach. The four elements needed to implement our approach are described: a text representation, the navigation knowledge modeling language Sextant, the knowledge encoding agents (via applications) and the NaviTexte system.

Two distinct applications of NaviTexte have been presented, showing the versatility of our approach. The quantitative results of our experimentation with Danish students learning French confirm the improvement obtained by using text navigation.

A long term consequence of modeling navigational knowledge is the creation of knowledge bases exchangeable and reusable. Actual collaborations are reusing the knowledge coming from the NaviLire project into others e-learning projects.

We think that our approach may have a significant impact on the way text is being read when its amount or nature does not allow sequential reading (*e.g.* the Web). Related to last works in Web Wise, we plan to couple our approach to Semantic Web approaches to exploit existing annotations.

## References


Battistelli D. and Minel J.-L. 2006. Les systèmes de résumé automatique: comment assurer une continuité référentielle dans la lecture des textes, *in G. Sabah (Ed.), Compréhension des langues et interaction*, 295-330.

Battistelli D., Minel J-L and Schwer S. 2006, Représentation des expressions calendaires dans les textes: vers une application à la lecture assistée de biographies. *Revue TAL*, 47(3): 1-26.

Benyon D. and Murray D. 1993. Developing Adaptive Systems to Fit Individual Aptitudes, In W.D.Gray, W.E., Helfley and D.Murray (eds.), Proceedings of the 1993 International Workshop on IUI, 115-122. Orlando, FL., New York, ACM Press.

Bilhaut F., Ho-Dac M., Borillo A., Charnois T., Enjalbert P., Le Draoulec A., Mathet Y., Miguet H., Pery-Woodley M.P. and Sarda L. 2003. Indexation discursive pour la navigation. intradocumentaire: cadres temporels et spatiaux dans l'information géographique. Actes de TALN 2003, 315-320.

Bodner R. and Chignell M. 1999. Dynamic hypertext: querying and linking, *ACM Computing Surveys*, 31(4): 120-132.

Boyle C. and Encarnacion A. 1994. MetaDoc: An Adaptive Hypertext Reading System, *UMUAI*, 4: 1-19.

Brusilovsky P. 1996. Methods and techniques of adaptive hypermedia . *UMUAI*, 6(2-3): 87-129.

Brusilovsky P. and Pesin L. 1994. ISIS-Tutor: An adaptive hypertext learning environment. In H.Ueono & V.Stefanuk (eds.), Proceedings of JCKBSE'94.

Brusilovsky P., Schwartz E. and Weber G. 1996. ELM-ART: An Intelligent Tutoring System on World Wide Web, ITS'96, Berlin, Springer, 261-269.

Couto J., Lundquist L., Minel J.-L. 2005. Naviguer pour apprendre. *EIAH 2005*, Montpellier, 45-56.

Couto J. 2006. Modélisation des connaissances pour une navigation textuelle assistée. La plate-forme logicielle NaviTexte. PhD, Université Paris-Sorbonne.

Couto J. and Minel J-L. 2006. Navigation textuelle: Représentation des textes et des connaissances, *Revue TAL*, 47(2): 1-24, Hermès, Paris.

Crispino G. and Couto J. 2004. Construction automatique de résumés. Une approche dynamique. *Revue TAL*, 45(1): 95-120, Hermès, Paris.

Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V. 2002. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, ACL'02, ACM Press, 168-175.

Danielson D.R. 2002. Web navigation and the behavorial effects of constantly visible maps, Interacting with Computers, 14: 601-618.

Edwards D.M. and Hardman L. 1989. Lost in hyperspace: cognitive mapping and navigation in a hypertext environment, in Hypertext: Theory and Practice. Oxford, Intellect Books, 105-125.

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

Géry M. 2002. Un modèle d'hyperdocument en contexte pour la recherche d'information structurée sur le Web. *Revue des Sciences et Technologies de l'Information*, 7/2002, Hermès, Paris, 11-44.

Höök K. and Svensson M. 1999. Evaluating Adaptive Navigation Support. In: Social Navigation of Information Space. Springer Verlag, 240-251.

Kaplan C., Fenwick J. and Chen J. 1993. Adaptive Hypertext Navigation Based On User Goals and Context, *UMUAI*, 3, 193-220.

Lundquist L. 1980. *La cohérence textuelle, syntaxe, sémantique, pragmatique*, Copenhagen, Nordisk Forlag.

Lundquist L. 2005. Noms, verbes et anaphores (in)fidèles. Pourquoi les Danois sont plus fidèles que les Français. Langue française. Vol 145: 73-92.

Lundquist L., Minel J.L. and Couto J. 2006. NaviLire, Teaching French by Navigating in Texts, IPMU'2006, Paris.

Mani I. 2001. *Automatic Summarization*, Amsterdam, John Benjamins Publishing Company.

Mathe N. and Chen J. 1994. A User-Centered Approach to Adaptive Hypertext based on an Information Relevance Model, UM'94, Hyannis, MA, 107-114.

Minel J.-L, Cartier E., Crispino G., Desclés J.P., Ben Hazez S. and Jackiewicz A. 2001. Résumé automatique par filtrage sémantique d'informations dans des textes, Présentation de la plate-forme FilText. Technique et Science Informatiques, n°3: 369-396.

Minel J.-L. 2002. Filtrage sémantique de textes. Problèmes, conception et réalisation d'une plate-forme informatique. HDR, Université Paris-Sorbonne.

Minel J-L. 2003. *Filtrage sémantique. Du résumé à la fouille de textes*. Paris, Hermès.

Minel, J-L., Nugier, S. and Piat, G. 1997. How to appreciate the Quality of Automatic Text Summarization. EACL 97, Madrid, 25-30.