MEDITE – A Unilingual Text Aligner for Humanities Applications to Textual Genetics and to the Edition of Text Variants

Jean-Gabriel Ganascia

LIP6 – University Pierre et Marie Curie (Sorbonne Universités)
BC 169, 4, place Jussieu, 75252, Paris Cedex 05, France
Jean-Gabriel.Ganascia@lip6.fr

Abstract

MEDITE is a powerful text comparison software that is the result of a collaboration between literary and AI scholars. It is based on string alignment techniques. The main difficulty comes from the necessity to detect displacements, which precludes the use of classical string alignment algorithms. This paper aims at introducing to the MEDITE tool. It is divided into six parts. The first describes the algorithmic difficulties of large string alignment in the context of textual genetics, because of the need to detect displacements. The second part shows the way in which the MEDITE algorithm overcomes those difficulties. The third compares the efficiency of different string alignment algorithms including MEDITE. The fourth presents the MEDITE interface, which facilitates browsing into text variants. The fifth explains what textual genetic is and how MEDITE can be useful in this field. The sixth explicates how it has been used for the edition of text variants. Lastly, the paper briefly concludes by providing details on the MEDITE software availability, on its future evolutions and on the perspectives it opens.

Introduction

MEDITE is a powerful text comparison software that is the result of a collaboration between literary and AI scholars.

More precisely, funded by the CNRS information society program, the EDITE[1] project made the ITEM - Institut des Textes et Manuscrits Modernes - collaborate with the ACASA team (Agents Cognitifs et Apprentissage Symbolique Automatique) that is member of the LIP6 laboratory – Laboratoire d'Informatique de Paris VI (Fenoglio & Ganascia, 2008). In other words, MEDITE is the result of an interdisciplinary collaboration between literary scholars and computer scientists.

MEDITE, which means *Machine EDITE*, has originally been achieved to solve the needs of textual genetic criticism, but it now appears to be useful in many literary applications (scholar publishing, automatic translation, computerized epistemology, etc.).

From an algorithmic point of view, it is nothing more than an efficient uni-lingual aligner. However, the problem is not easy to solve. The main difficulty comes from the need to detect possible textual displacements, which precludes the use of classical edit-distance string alignment algorithms based on standard editions, i.e. insertion, deletion and replacement. As we shall see in the following, there does not exist any optimal solution to this problem, which requires trade-off and heuristics.

This paper aims at introducing to the MEDITE tool. Apart the introduction and the conclusion, it is divided into six parts. The first describes the algorithmic difficulties of large string alignment when considering textual displacements. The second part shows the way in which the MEDITE algorithm overcomes those difficulties. The third compares the efficiency of different string alignment algorithms including MEDITE. The fourth presents the MEDITE interface, which facilitates browsing into text variants. The fifth explains what textual genetic is and how MEDITE can be useful in this field. The sixth explicates how it has been used for the edition of text variants. Lastly, the paper briefly concludes by providing details on the MEDITE software availability, on its future evolutions and on the perspectives it opens.

1 EDITE is an acronym for *Edition Diachronique et Interprétative de Textes d'Ecrivain*, which makes an implicit reference to the notion of "edit-distance"

The Problem and its Difficulties

The unilingual text alignment is the process that compares two texts written in the same language and that, based on the results of this comparison, arranges them in a way that puts in evidence their similarities and their differences. Defined as such, the text alignment task requires two steps that are firstly the mechanical comparison of texts and secondly their presentation in a manner that emphasizes their similarities and differences.

The automatic comparison of texts is a generic problem to the solution of which many algorithms were designed. Each of them has its own characteristics and is dedicated to specific tasks. Among them, the MEDITE algorithm was specially conceived for unilingual text alignment in literary studies and, more specifically, in textual genetics. To understand its peculiarities, let us mention three other applications of text alignment.

One of the oldest text comparison algorithm is the famous *diff* (Hunt & McIlroy, 1976), which compares two files line by line and outputs a list of inserted and deleted lines. This program is issued from the community of computer scientists who created the Unix operating system. This approach is restricted to the grain of the line, which means that intra-line modifications are not adequately addressed. For instance, the addition or deletion of a character in a line leads to the deletion of this line and to the addition of a new line. If such a solution is acceptable for code source it isn't for our purpose for at least three reasons: firstly the notion of line does not play the same role in literary texts – except, maybe in poetry – than in code source; secondly, the line is a too coarse grain for literary studies; thirdly, the displacement of text blocs is not detected.

Modern approaches in machine translation, what is currently called "statistical translation", are based on automatic bilingual text alignment. This involves comparing manually translated texts to their original version to automatically improve machine translation. Most of the multilingual text alignment rely on Machine

Learning techniques (Manning & Schtze, 1999) that train statistical models from bilingual reference corpus such as french-english Hansards. This approach is not relevant for us, because in our case there does not exist any relevant aligned corpus. In addition, our goal is to detect modifications of texts, while in case of machine translation, each word or expression in the first language must match a similar unit in the second.

The most relevant approaches come from bioinformatics, where DNA, i.e. sequence of nucleic acids, or proteins, i.e. sequence of amino acids, are automatically aligned. With the unilingual text alignment as with the alignment of biological macromolecules, sequences are written with the same alphabet and the alignment is based on the character pairing. Different alignment types exist in bioinformatics, somme are local (Smith & Waterman, 1981) while other are global (Needleman &Wunsch, 1970). More precisely, the first look similar regions in biological sequences, while the latter try to match the complete sequence. In our particular case, i.e. in the case of textual genetics, the only interesting techniques are the global ones, because we want to identify the set of transformations between two versions of the same text. However, local techniques may also be valuable in literary criticism, for instance when we want to identify plagiarisms, reuse of contemporary texts or pastiches. Despite those similarities between biological sequence alignment and unilingual text alignment, the techniques developed by bioinformaticians cannot be reused as is for at least two reasons. One the one hand, the text alignment requires basic linguistic knowledge about the role punctuation signs or the limits of words, which is irrelevant for macromolecules alignment. On the other hand, it appears crucial to detect textual displacements for solving the needs of textual genetics, while this does not seem to be so essential to bioinformatics. This last point is critical, because the classical text alignment algorithms don't succeed in efficiently managing textual displacements and because there is no optimal solution to this problem, which necessitates the introduction of heuristics and tradeoff.

Lastly, in addition to these algorithmic difficulties of the unilingual text alignment process, it is needed to present the aligned texts in a way that facilitates both their reading and the browsing through the text transformations. As we shall see in the following, it means that the visual presentation of the texts needs to be completed by interactive functionalities that help the pairing of textual blocks to their counterparts.

MEDITE Alignment Algorithm

MEDITE is built on an original sequence alignment algorithm, which is based on the edit-distance with moves conceptual frame. Let us first recall that the notion of edit-distance has been introduced by mathematicians to formalize string alignment problems. It is based on the notion of edition, that designates a local transformation in a sequence, for instance, the deletion or the insertion of a character or a word, or the replacement of one element by another. To each of those operators is associated a cost. The notion of edit-distance (Crochemore & Rytter 1994; Sankoff & Kruskal, 1983) is based on the minimal cost of editions that allows the transformation from a sequence to another. As a consequence, computing the edit distance between two sequences is equivalent to align the text, because it leads to associate to each element of a sequence its "cheaper" counterpart, with respect to the transformation operator costs.

The traditional edit-distance problem is based on a restricted set of editions called the standard set and that contains the *insertion*, the *deletion* and the *replacement*. However, besides those three standard operators, there exist other operators that may appear to be meaningful in various applications. For instance, when rewriting natural language texts, moving a chunk of text from one place to another doesn't seem as costly as the sum of the deletion cost and the insertion cost of this chunk, as the standard edit-distance would suggest. Therefore, adding a *move* operator to the standard set of edition could be suggested. Unfortunately, when adding the move operator to the set of editions, the algorithm that computes the exact edit-distance becomes too complex to be computed in an efficient manner. As a consequence, to solve this problem, we have introduced heuristics techniques that allow to approximate efficiently the optimal text alignment while taking into account the possibility of moving chunks of text.

These heuristics are based on a three steps algorithm, which is called "alignment of fragments".

The first step of this algorithm detects invariant character blocks (i.e. the "fragments"). Then a second step distinguishes, among the invariant character blocks, those which are displaced, i.e. moved. Lastly, a third step identifies the deleted, inserted and replaced blocks that are between the unmoved invariant blocks. Note that, using this algorithm, a fragment of text may appear to be both displaced and inserted or deleted.

Let us now precise the way those different algorithmic steps are achieved in the MEDITE program.

The first one detects Maximal Exact Matches (MEM) that are the maximal exact homologies between the two texts, i.e. the homologies that can't be extended to the left or to the right without losing identity. To achieve this step, the algorithm extracts all the maximal homologies between the two texts by building a generalized suffix tree over the whole two sequences. Usually, a minimum size parameter is chosen by the user (by default, we restrict to homologies greater then 4-characters long), but this is not problematic. The real problem we encounter is due to the existence of overlapping maximal homologies. Let us consider, for instance, the alignment of the two following strings: *"Il a avalé"* and *"Il avala"*. There exist two maximal homologies between those two strings, which are |*Il a*| and | *aval*|. However, those homologies are not disjoint: the substring | *a*| ends the first and begins the second, which means that they overlap each other. It is then necessary to solve this problem in order to obtain a proper partition of the whole sequences in disjoint blocks. For this, we use a heuristics based on natural language properties: if the overlapping block contains separators (i.e. punctuation or white), it is better to cut it on one of them. For instance, in our example, the hyphenation may appear on three positions (that are |↑ *a*|, | ↑*a*| and | *a*↑|), which gives the three following sets of non overlapping homologies: (1) |*Il*| and | *aval*|, (2) |*Il* | and |*aval*| and (3) |*Il a*| and |*val*|. This example shows that, while the

hyphenations 1 and 2 are correct, the hyphenation 3 isn't, because it leads to unnecessarily cut a word.

Among the non overlapping MEMs that are built in step 1, some are in the same order in both texts, while others are displaced. The first are called invariants while the second are said to be moved blocks. The distinction between invariant and moved blocks is obtained by browsing the space of all possible alignments with a classical optimization procedure, the A* algorithm, which minimizes the alignment cost function. This function takes into account the total size of displaced chunks and, for each of them, the length of the gaps between its two relative positions in the two texts. Note that there is no exact solution to this step, which depends on compromises among different preference criteria, e.g. the way gaps influence the cost of displacements of lengthy blocks. This whole process is then applied recursively between each pair of aligned invariant blocks in order to detect smaller blocks and to avoid a masking phenomena that precludes an optimal alignment.

To make more clear what we mean by this masking phenomena, let us consider the two following sentences: “Ce matin le chat observa de petits oiseaux dans les arbres.” and “Le chat était en train d’observer des oiseaux dans les petits arbres ce matin. Il observa les oiseaux pendant deux heures.”. Let us now suppose that we extract the MEMs without taking into account neither the punctuation, neither the blanks, nor the letter case, i.e. the difference between lower and upper cases. We obtain the seven following MEMs: “ce matin”, “le chat”, “observa”, “de”, “petits”, “oiseaux dans les”, “arbres”. We emphasize the invariant in the two previous sentences by underlying the common blocks: “Ce matin le chat observa de petits oiseaux dans les arbres.” and “Le chat était en train d’observer des oiseaux dans les petits arbres ce matin. Il observa les oiseaux pendant deux heures.”, which leads to the following optimal alignment that is emphasized in bold: “Ce matin **le chat** observa **de** petits **oiseaux dans les arbres**.” and “**Le chat** était en train d’observer **de**s **oiseaux dans les** petits **arbres** ce matin. Il observa les oiseaux pendant deux heures.”

However, it appears that some repeated strings are not detected, while they should. It is the case with the first occurrence of the string |observ| and with the second occurrence of |oiseaux| in the second sentence. The recursive call facilitates this detection and gives the following alignment – emphasized with bold characters – that is clearly better that the previous, because it now aligns the string |observ|: “Ce matin **le chat observa de** petits **oiseaux dans les arbres**.” and “**Le chat** était en train d’**observ**er **de**s **oiseaux dans les** petits **arbres** ce matin. Il observa les oiseaux pendant deux heures.” This phenomenon is called a “masking effect”, because the homology on |oberva| hides the alignment on the first occurrence of the substring |observ|.

Finally, as deleted, inserted and replaced blocks are non repeated blocks, they are deduced from the alignment of invariant blocks obtained in the step two.

The resulting software (Bourdaillet & Ganascia, 2007) is able to find in some minutes moves between two versions of a 500 pages novel, and very robustly, even if there are a lot of differences between the two versions.

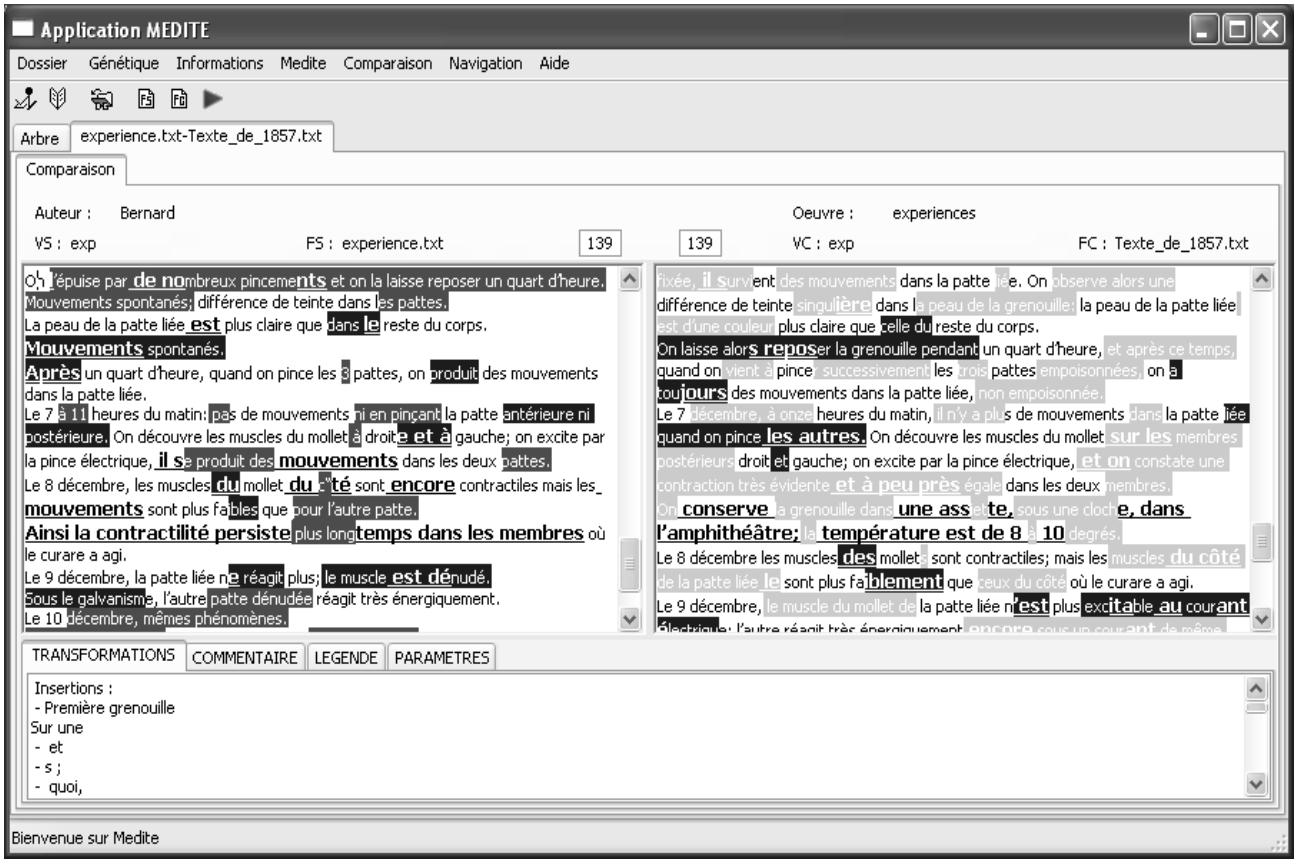

Figure 1: comparison of two Claude Bernard's text with MEDITE

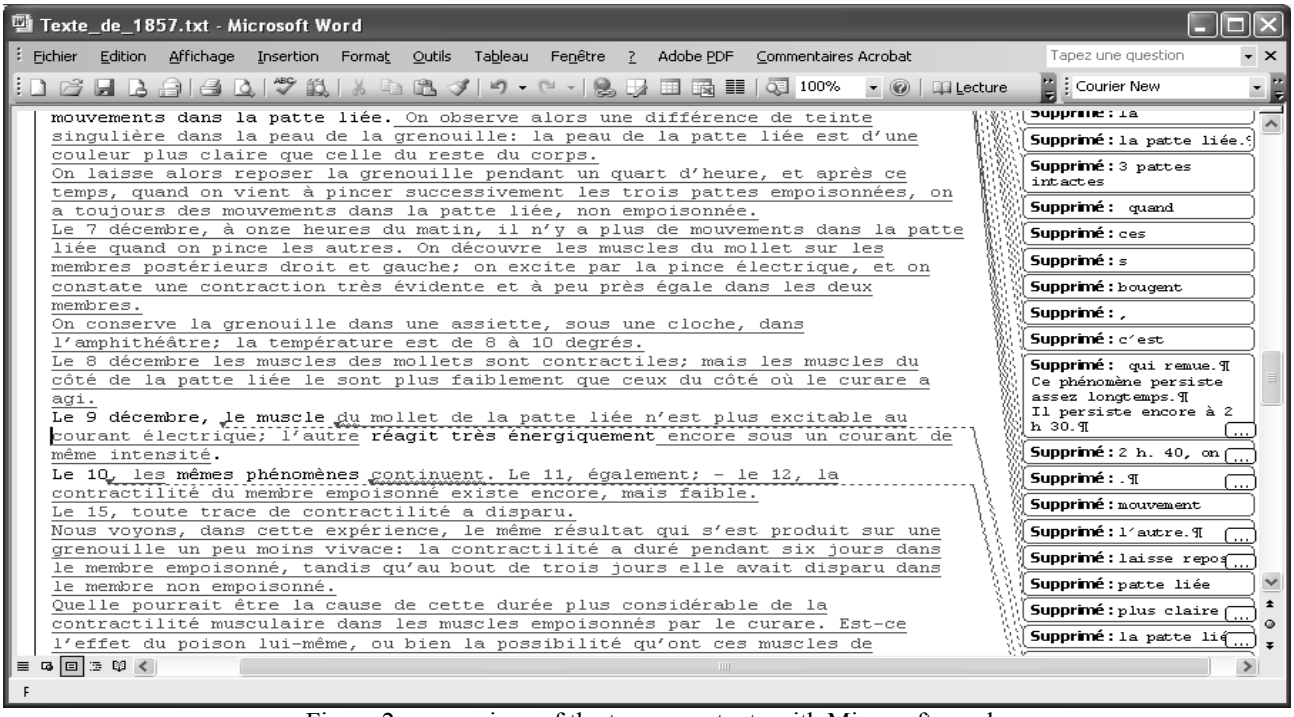

Figure 2: comparison of the two same texts with Microsoft word

Comparison with Other Text Alignment Algorithms

MEDITE has been compared with other version comparison tools, the most famous being the one inside Microsoft Word. None of them was able to align correctly hard texts and to overcome masking phenomena but MEDITE.

An example of comparison of these two texts using MEDITE is given in Figure 1 and using Microsoft Word in Figure 2. It can be seen that MEDITE identifies considerably more invariants (in black and white) between the two texts than Word, resulting in a better alignment.

MEDITE has then been systematically compared with six aligners, the most commonly used being the one present in Microsoft Word. For each application, three file comparisons were made, where three points were tested (identified with capitalized letters below).

The first comparison is between two versions of a short story by Pascal Quignard entitled "Le Chant des Enfants du Marais". Small modifications of some characters were introduced throughout the text. Lexical words were changed, misspellings corrected and words moved. The goal was to find such modifications. Paragraphs must be aligned (A); word modifications must be found (B) and character modifications must be found (C).

The second comparison is between a news agency dispatch and an article which is rather different but derived from it. Two paragraphs were kept with some internal modifications, and the remaining text was replaced completely by another one. The two paragraphs must be aligned (D); modifications inside these paragraphs must be found (E) and similar lexical words must be found (F).

The third comparison is the one described in the beginning of this section. Two texts from Claude Bernard, one from its experiment notebooks and the second from a scientific paper were aligned. This task is very hard because the existing content remained the same but the form changed and new content was inserted. Paragraphs must be aligned (G); word groups must be aligned (H) and isolated words must be aligned (I).

The results of this experiment are presented in Table 1. Paragraph alignment (A) is correct for all the applications. Only four applications detect word changes in test (B) and only MEDITE and Compare It detect character changes (C). The others detect character changes as word changes, whereas often only one or two characters have been modified. By contrast MEDITE focuses on the modified characters.

For the second comparison, only DiffDoc and MEDITE align the two paragraphs (D) and find small internal modifications (E). All the other applications fail to detect this. This test is useful because the longest invariant sequence is 752 characters long for two texts of 14 Ko and 18 Ko, and so represents about 5% the size of each file. As it doesn't change, we could except all software to find it but only two of them do. Because the theme of the two texts is related, common lexical words are used in the remainder of the texts but only MEDITE aligns them correctly (F).

	A	B	C	D	E	F	G	H	I	Total
MEDITE	1	1	1	1	1	1	1	1	1	9
DiffDoc	1	1	0	1	1	0	1	1	0	6
Word	1	1	0	0	0	0	1	1	0	4
Compare It	1	1	1	0	0	0	0	0	0	3
Visual Comparer	1	0	0	0	0	0	1	0	0	2
Araxis Merge	1	0	0	0	0	0	0	0	0	1
Beyond Compare	1	0	0	0	0	0	0	0	0	1

Table 1: comparison of different aligners

The third comparison is the hardest one. Paragraphs are aligned correctly only by DiffDoc, MEDITE and Word (G). Several word groups are aligned by DiffDoc and Word but a lot are missed (H). We know they are missed because MEDITE detects them. As DiffDoc and Word miss numerous word groups, they miss isolated word changes whereas MEDITE aligns them pairwise correctly (I). The absence of these alignment anchors results in a bad alignment because a lot of information is not discovered and it impacts on the readability of the alignment. Our result can be viewed in Figure 1. The less the texts are aligned the less the visualization is good. In earlier versions of MEDITE we had similar problems but the introduction of recursion in our algorithm enabled us to address them.

None of the applications except MEDITE detects moved blocks, though we have already said that this is crucial for philology. For source code comparison, this is still the case. Detecting that a code line has been moved from one function to another is an important piece of information. It is also important for any natural language text, because it makes possible to detect rearrangements of ideas, for instance.

The MEDITE Interface

For results visualization, the two texts are presented side-by-side in a two panel GUI (cf. Figure 1 and figure 3) . Deletions, insertions and replacements are highlighted in specific colors that can be parametrized by the users. Moves are underlined, which enables to visualize moves inside insertions or deletions. Invariants stay black on white. In addition to text visualization, pairwise aligned blocks are internally linked together. A simple mouse click aligns them side-by-side. It goes the same with displaced blocks, which can be align side-by-side by just clicking on one of the two moved blocks. Those two functionalities help readers to browse through aligned texts and to interact with alignments, which is particularly useful in case of displacements.

In addition, a panel presents all the transformations (i.e. deletions, insertions, replacements and moves), which can help to make statistics and to better understand the nature of the transformations. For instance, it could be possible to establish a typology of transformations, e.g. to characterize the text rewriting as a contraction or inversely as an amplification.

Lastly, the MEDITE interface allows to parametrize the alignment, for instance to fix the minimal size of MEMs, to decide not to take care of punctuation, diacritic signs or letter case, etc.

Use of MEDITE for Textual Genetics

Textual genetic criticism is a discipline that studies drafts led by authors during the writing process (deBiasi, 2000; Hay, 1979). MEDITE's first aim was to align linearized transcriptions of such drafts (usually, two texts) in order to identify invariants and differences between them. To characterize those differences, we reuse the four operators previously identified by textual genetics (Ganascia, Fenoglio & Lebrave, 2005): *deletion*, *insertion*, *replacements* and *displacements*. Those operators correspond exactly to the set of editions used in MEDITE, i.e. to the standard set {deletion, insertion, replacement} completed with the move operator. As previously shown, MEDITE automatically identifies the sequence of applications of these editions that transforms the first version of the text into the second.

Usually, this task is done manually by textual geneticists, which is tedious, boring, costly and impossible to achieve on long texts. As a consequence, the use of MEDITE helps to exhaustively enumerate all the differences between the two versions of texts, which was practically impossible before, without computers.

We have compared the transformations obtained with MEDITE and the transformations that where manually annotated by textual geneticists. It appears that they were very close, even if some open questions remain. For instance, the intra-words transformations are not always relevant; sometimes they are needed, sometimes not. It goes the same with the displacements of small words or of small expressions. Lastly, the arbitration between on the one hand the combination of a deletion and an insertion and on the other hand a replacement is not an easy task to automate. As a consequence, the last version of MEDITE allows to retouch the transformations in case of needs.

It strangely appears that for texts with a lot of repetitions, existing aligners (version comparison tools) failed to perform correct alignments. This is due to the masking phenomena mentioned above, which appear when the pairing of two text blocks hides and therefore avoids the pairing of other identical blocks. MEDITE addresses this problem by the above mentioned recursive step.

Publications of Text Variants

MEDITE is now used by philologists in textual genetic criticism and epistemologists in understanding ideas' history (Ganascia, 2008; Ganascia & Debru 2007). It

enables them to study longer texts and to discover, more systematically, transformations between authors' draft. They can then establish diachronic corpus of an author's work.
Today, MEDITE is also used to facilitate the publication of the texts of different published versions. It greatly alleviates the establishment of the textual apparatus by automatically highlighting the differences between versions. Simultaneously, it allows the presentation of the differences, which was practically impossible before, with classical printing edition. For the sake of illustration, the MEDITE software was used by scholars of Lausanne University to publish the new edition[2] of the Charles Ferdinand Ramuz's works by Slatkine. Under the supervision of Daniel Maggetti, the editors of this work wanted to integrate to the publication the different variants of the Ramuz's novels. However, it was both too expensive and inconvenient to envisage it manually. More precisely, the price of a manual edition would be unaffordable, because it is a too long and fastidious task to track the differences between the different version of the novels, especially to detect the displacements. So, with MEDITE, a digital publication renders possible what would be too expensive for a classical publication.
In addition, the presentation of the differences between versions is very difficult with a classical paper publication because it is particularly embarrassing to highlight the transformations. It requires, for instance, to add footnotes, which is cumbersome and difficult to read. Furthermore, when there are more than two versions, it is insufficient to present side by side one version on even pages and the second on odd pages. As a consequence, the reader cannot easily read in parallel the different versions without having to turn pages, which is very inconvenient. The MEDITE interface makes it very easy for the reader to navigate through the comparison between different versions of the text. It means that an electronic version is added to the paper publication of Ramuz's work by Slatkine under the form of a CD-rom.

Conclusion & Perspective

An existing version of MEDITE is freely available[3]. Interest readers who have questions or who would like specific options can also send me an email.
The experience with MEDITE shows the added value, for the humanity studies, of a new type of hybrid edition combining electronic support to classical paper books. We are now working on a new hybrid edition of Balzac work with Pierre Glaudes and Andrea del Lungo, which will include the comparison between the different publications of the novels. It will undoubtedly constitute an improvement of classical Balzac publications. The CNRS program PHOEBUS helps us to explore the technical aspects of this work in progress, which is yet in a preliminary phase. In particular, the interface has to be augmented to make it possible to show more than two versions, with more than two columns.
But the improvements will not be limited to the only interface. We would like also to take into account the variant languages into the text alignment, to focus on the relevant variants, without taking into account the typographic or spelling differences. Lastly, we plan to use text alignment techniques for intertextual studies, for the detection of self-rewriting – or self-plagiarism – in an author work. All those perspectives show the incredible enrichments of textual analysis due to the use digital editions, among which the MEDITE software constitutes an illustrative concrete contribution.

2 A few articles have mentioned this event in May 2011. Cf. http://www.myscience.ch/wire/ramuz_edition_papier_et_cd_rom-2011-unil & http://sigales.hypotheses.org/132

3 It is possible to download it on my website at the following URL: http://ganascia.name/Medite_Project.

Acknowledgements

I am indebted to Rudolf Mahrer who initiated the use of MEDITE in the Ramuz's publication and who patiently helped to correct the result of the MEDITE software. I am also indebted to Julien Bourdaillet who contributed to greatly improve MEDITE during his PhD thesis.

Bibliographical References

Bourdaillet J., Ganascia J.-G., "MEDITE: A Unilingual Textual Aligner."Proc. of the 5th International Conference on Natural Language Processing (FinTAL 2006), LNAI vol.4139, pp. 458-469. Turku, Suomi.

Bourdaillet J., Ganascia J.-G. : "Practical block sequence alignment with moves", 1st International Conference on Language and Automata Theory and Applications (LATA), LNCS, Tarragona, Espagne (2007)

Crochemore M, Rytter W., *Text Algorithms*, "Approximate pattern matching", (1994), pp. 237-251

de Biasi, P.M., *La génétique des textes*. Nathan Université (2000)

Fenoglio I., Ganascia J.-G.: "Le logiciel MEDITE: approche comparative de documents de genèse", in L'édition du manuscrit - De l'archive de création au scriptorium électronique, Aurèle Crasson, Academia A| B Bruylant, col. Au coeur des textes, n°10, pp. 209-228, (2008).

Ganascia J.-G., "In silico' Experiments : Towards a Computerized Epistemology", in Newsletter on Philosophy and Computers, Piotr Boltuc (ed), American Philosophical Association Newsletters, 07 (2), 11-15, Spr. 2008, ISSN 1067-9464

Ganascia, J.-G., Debru, C.: CYBERNARD: A Computational Reconstruction of Claude Bernard's Scientific Discoveries, Model-Based Reasoning in Science, Technology, and Medicine, vol. 64, Li, Ping, pp. 497-510, Springer Verlag Ed. (ISBN : 978-3-540-71985-4) (2007)

Ganascia J.G., Fenoglio I., Lebrave J-L, Manuscrits, genèse et documents numérisés. EDITE : une étude informatisée du travail de l'écrivain, revue Document numérique, special issue on « temps et document » (2005)

Gusfield D., *Algorithms on Strings, Trees and Sequences: Computer Sciences and Computer Biology*, Cambridge University Press (1997).

Hay, L., ed., *Essais de critique génétique.* Flammarion, coll. Textes et Manuscrits (1979)

Hunt, J.W., McIlroy, M.D., An Algorithm for Differential File Comparison. Technical Report CSTR 41, Bell Laboratories, Murray Hill, NJ (1976)

Manning C.D., Schtze H., "Foundations of Statistical Natural Language Processing", MIT Press, (1999).

Smith T.F., Waterman M.S., "Identification fo Common Molecular Subsequences", Journal of Molecular Biology, 147, (1981), pp. 195-197

Needleman S., Wunsch C., "A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins", Journal of Molecular Biology, 48(3), (1970), 443-452.

Sankoff D., Kruskal J.B., *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison*, Addison-Wesley, Reading, Mass., (1983)

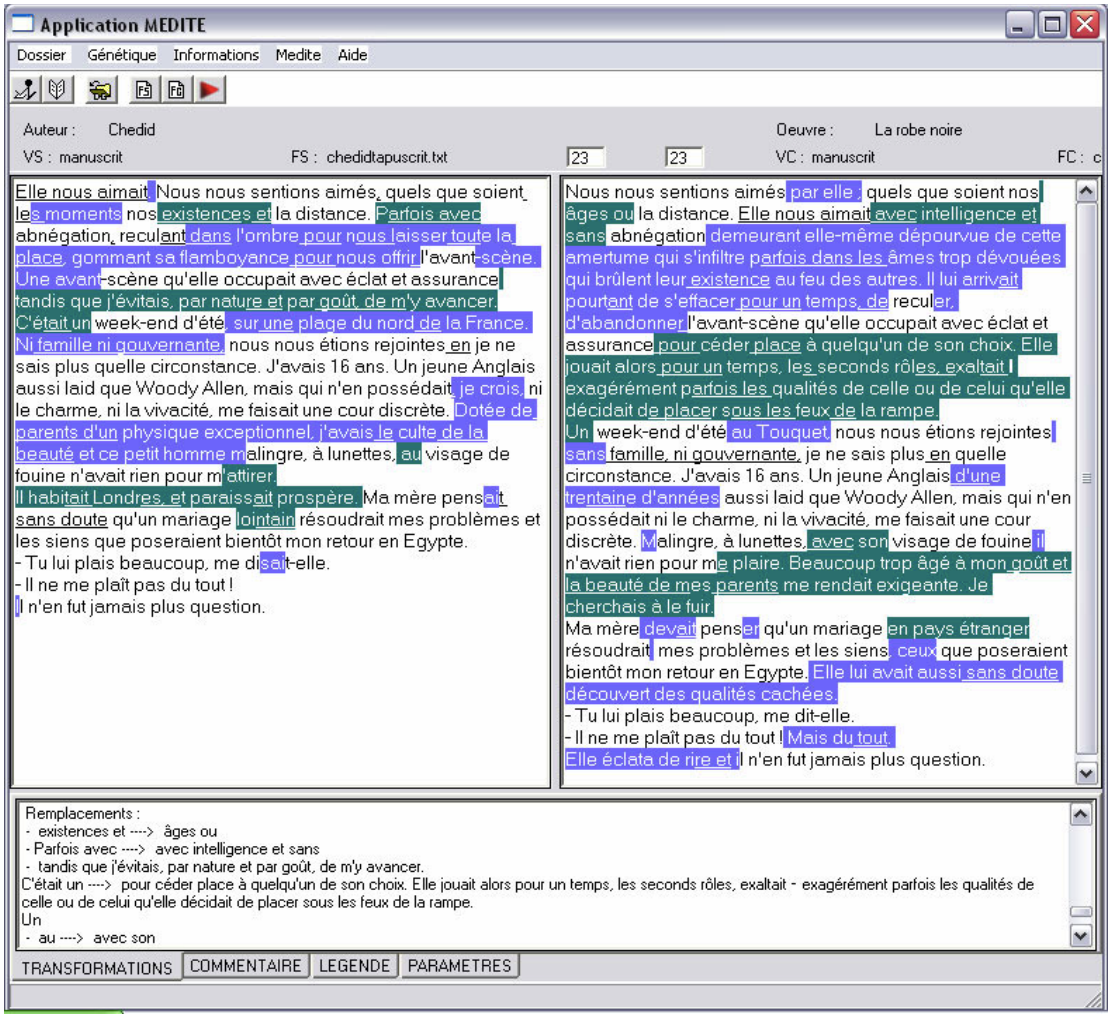

Figure 1: an Illustration of the MEDITE Interface