Using the Computer to Study the Dynamics of the Handwriting Processes

Gilles Caporossi[1], Denis Alamargot[2], and David Chesnet[2]

[1] GERAD and HEC Montréal, Canada
Gilles.Caporossi@hec.ca
[2] LACO CNRS and Université de Poitiers, France
{Denis.Alamargot,David.Chesnet}@univ-poitiers.fr

Abstract. In this paper, we present tools to help understanding the dynamics of cognitive processes involved in handwriting and in text composition. Three computer systems or programs used for this analysis are explained, the results obtained by their mean are exposed and their potential meaning discussed.

1 Introduction

For cognitive sciences, there is a great interest in understanding the writing processes. Indeed, text composition is an increasingly present human activity in our society. The historical development of computer has strengthened the use of keyboard but the forthcoming Tablet-PC and the well-diffused PDA give back a privileged role to handwriting for text composition and for data capture. Identifying the mental mechanisms underlying the handwriting activity brings up several outcomes.

1. At a fundamental level: to study the strategies of the writer in order to improve our knowledge of human rules to process the information.
2. At an educational level: to teach children and students to use expert strategies for writing and for composing.
3. At an ergonomic level: to design writing tools (systems for data capture, text processing, etc.) adapted to human strategies.
4. At an artificial intelligence level: to generate written code making the computer able to describe a database and to communicate with natural language.

According to cognitive psychology, writing is a complex activity implying transformation of a multidimensional knowledge structure (domain knowledge) into a linear sequence of words (the text). This transforming must respect linguistic conventions (linguistic knowledge: spelling) and communicative conventions (pragmatic knowledge: legibility, relevance). Several processes are involved in this transformation.

1. The planning process generates and organizes text content by retrieving domain knowledge from long-term memory or by encoding domain information from the environment (documentary sources, for instance).

E. Suzuki and S. Arikawa (Eds.): DS 2004, LNAI 3245, pp. 242–254, 2004.

2. The formulation process translates semantical representation into linguistic structures.
3. The revising process allows the writer to evaluate and to modify conceptual and linguistic characteristics of the text produced so far.
4. Finally, the execution process performs the written message by fulfilling graphomotor plans.

These processes have been well described in theoretical models of writing [1] and the functioning of each one has been studied in numerous experiments, involving various participants [16]. Nevertheless the question of the dynamics of the writing processes still remains difficult to answer. At least, writing is a dual processing activity. The writer encodes visual information as input (text produced so far; potential documentary sources) and executes graphomotor plans as output (to form letters on the screen or on the sheet of paper). The synchronization between input and output depends on the nature and on the number of writing processes engaged to assume the transformation of information. Those 'intermediary' writing processes can be fired simultaneously but due to the limited capacity of human cognitive system, some of them (mainly the most demanding in terms of cognitive resources) need to be postponed and engaged sequentially. Several mechanisms like metacognitive control [12] or processing capacity [10] have been evoked to explain why a process is fired (in terms of decisive criteria) and how it is fired (in parallel or sequentially). Indeed, the actual difficulty to understand such rules of writing dynamics is to precisely identify the moment when a process is engaged and the nature of this process. This scientific discovery may significantly be improved by the use of the machine.

2 Aim of the Research

The aim of this paper is to present a computer environment which allows the scientist to understand the engagement of writing processes and to discover the rules underlying this dynamics. If complete automation is a goal for research in artificial intelligence, Langley [14] and Valdés-Perès [19] also consider the computer as a collaborator for the researcher.

To describe the dynamics of writing processes, we ask a participant to compose a text from sources and we use the machine to analyze the temporal relationships between the activities of the eye (taking visual information from source, in the task environment) and of the pen (writing the text, making pauses, etc.). We argue that relationships between the eye and the pen constitute pertinent indicators of the writer's mental activity and notably of the writing processes engaged.

The use of two systems as well as data mining programs lead us to make the evidence of some previously suspected relations but also to discover some new knowledge about the writer behavior. As a first step, the *Eye and Pen* system records data while a participant compose an instructional text (directions required to build a turbine) from documentary sources (pictures and names of the different parts of the turbine). These data are extracted from a larger experimental study conducted with sixteen participants [2]. The second step consists in the use of some clustering programs in order to find different processing sequences associated to the handwriting processes. As a third step, the DyDA program is used for visualization and analysis of the results of the clustering programs as well as the visualization of the data.

3 Description of the Eye and Pen System

The Eye and Pen paradigm [8] was designed to study the dynamics of writing processes. Based on the synchronized measurements of ocular (the input) and graphomotor (the output) activities of the writer, the system provides a very fine-grained description of the temporal characteristics of written production. Eye and Pen relies on two main devices: a digitizing graphic tablet (to record spatial coordinates and pressure of the pen on the tablet surface) and an eye-tracker (to record eye movements). The whole is controlled by two PC type micro-computers. The first one, devoted to the eye movements acquisition sends the gaze position coordinates to the second PC which also simultaneously records the informations given by the graphic tablet. All these observations are stamped with a common base millisecond timing.

Eye and Pen data can be analyzed by a module which is an evolution of G-Studio software [9]. From the text written by the subject, and digitalized by the tablet, one may rebuild forward and backward on the screen, the trace leaved by the pen and the gaze position at the same time (synchronized events). Each event (movement or stop of the pen, saccades or fixations of the gaze) is numbered and may have a code assigned to select, sort and classify data.

4 Using Clustering to Study Writing Processes

In order to highlight the structure underlying the data, three clustering methods were used. The first one is the classical centroid based clustering model using sum of squared distance between each observation and the center of its cluster. As this model did not help finding any useful structure in the data, two other clustering models were tried. These two methods, based upon linear models are hyperplane clustering and clusterwise regression. All the three models are described below.

4.1 The Centroid Based Clustering Model

The centroid based model may be defined as follows:

$$Min \sum_{i=1}^{m} \sum_{k=1}^{K} z_{ik}(x_{ij} - c_{kj})^2 \tag{1}$$

$$Subject\,To$$

$$\sum_{k=1}^{K} z_{ik} = 1 \,\forall i = 1 \ldots m \tag{2}$$

$$z_{ik} \in \{0, 1\} \,\forall i = 1 \ldots m \,\forall k = 1 \ldots K. \tag{3}$$

Where $z_{ik} = 1$ if the observation i belongs to cluster k and equals 0 otherwise and x_{ij} represents the j^{th} coordinate of observation i. The centroid of cluster k's coordinates are computed using the following equations

$$c_{kj} = \frac{\sum_{i=1}^{m} z_{ik} x_{ij}}{\sum_{i=1}^{m} z_{ik}} \,\forall k = 1 \ldots K \,\forall j = 1 \ldots n. \tag{4}$$

The objective is to cluster the data in such a way that the sum of squares of distance between each observation and the centroid of its cluster is minimized. Equation (2) means that each observation belongs to exactly one cluster. The centroid based model is usually referred to as *k-means*. Although, *k-means* makes reference to an algorithm to obtain the clusters instead of just the model. We make this distinction here because if we consider the same problem, the way we search for the clusters is improved compared to the standard *k-means* algorithm.

4.2 The Hyperplane or *k-Plane* Clustering Model

As explained by Caporossi and Hansen [5] [6], a way to extract information from a database is by finding an hyperplane (relation) fitting the data. This method finds relations using the mathematics of the principal component analysis but instead of considering the largest eigenvalues of the variance-covariance matrix, one concentrates to the smallest. Should this smallest eigenvalue be close to 0, a relation is found. If this method was successful in graph theory, it also reveals a great potential in other fields when applied within a clustering scheme. Indeed it is possible that one single hyperplane does not fit the data but if two or more equations are considered, any observation is close enough to at least one of them. In this case, we have a more complex relation of the kind "either the relation A holds, or the relation B does". The *k-plane* algorithm proposed by Bradley and Mangasarian [4] uses the principle of *k-means* to handle this clustering problem. The *hyperplane* clustering problem may be formulated in the same way as the previous model, except that the objective function is:

$$Min \sum_{i=1}^{m} \sum_{k=1}^{K} z_{ik} d_{ik}. \tag{5}$$

Where dik is the distance between the hyperplane corresponding to cluster k and the observation i. The hyperplane corresponding to cluster k is deduced from the last eigenvector (corresponding to the smallest eigenvalue) of the variance-covariance matrix associated to cluster k.

4.3 The Clusterwise Regression Model

The main difference between *k-plane* clustering and the clusterwise regression is that a dependent variable that one needs to predict is used in the last case.

Another difference is that the error corresponding to a given observation is not computed in the same way for both models although this last difference is not of great importance in practice.

The clusterwise regression constraints the model to involve a chosen variable in its model. As first exposed by Späth [18] in 1979, in the clusterwise regression model, each cluster is represented by a regression model fitting the corresponding data and each observation is assigned to the cluster that best fits it. The clusterwise regression model may also be written in the same way as the previous ones except that the objective function is [15]:

$$Min \sum_{i=1}^{m} \sum_{k=1}^{K} z_{ik}(y_i - \sum_{j=1}^{n} b_{jk}x_{ij} + b_{0k})^2 \tag{6}$$

where the variables b_{jk} are regression coefficients for the cluster k.

4.4 Optimization Algorithm Used for Clustering the Data

For clustering data using the centroid model, the *k-means* algorithm is usually used. In the case of hyperplane clustering, the *k-plane* algorithm proposed by Bradley and Mangasarian uses the same scheme. In the case of the clusterwise regression, Späth [18] proposes a different algorithm taking advantage of the possible update of the models. All these algorithms thus lead to a local optimum and none explicitly consider routines to get out of it (except multistart).

The *alternate descent* algorithm upon which are based *k-means* and *k-plane* is the following:

1. Randomly choose the cluster corresponding to each data.
2. For each cluster, compute the best model *i.e.,* the centroid, the parameters of the hyperplane that best fits the data or the regression parameters, depending on the model used.
3. If an observation best fits another cluster *i.e.,* it is closer to another centroid or to another model, move it to that cluster. If at least one observation was moved, go back to Step 2 otherwise, the descent is complete.

This *alternate descent* algorithm converges quickly but the solution so found highly depends on the initial solution used, which makes it a bad optimization technique as it often misses the global optimum. Alone or within a *multistart* scheme, it leads to rather poor results on which one cannot rely enough to identify the models underlying the data. On the other hand, is is much faster than other descent methods we know for this problem.

As the main reason for the use of a clustering algorithm in discovery science is to find a partition of the data that leads to interpretation and allows the construction of hypothesis, the optimization algorithm must be fast and reliable. For this reason, in the context of a project with the Bell Laboratories, an efficient framework for optimizing clustering problems was developed using a heuristic based upon the Variable Neighborhood Search (VNS) [17] [13].

The Variable Neighborhood Search Algorithm

Repeat until the stopping condition is met:

1. Set $k = 1$;
2. Until $k = k_{max}$, repeat the following steps
   (a) (*shaking*) generate a partition x' at random from the k^{th} neighborhood of x (i.e., $x' \in N_k\, X(x)$):
   (b) (*descent*) Apply alternate local search with x' as initial solution: denote by x'' the local optimum obtained;
   (c) (*improvement or continuation*) If the solution x'' so obtained is better than the best known one x, move there ($x \leftarrow x''$) and continue the search within $N_1(x)(k = 1, l = 1)$; otherwise set $l \leftarrow l + 1$. If $l = l_{max}$ then set $k \leftarrow k + 1$ and $l \leftarrow 1$.

In the case of clustering, shaking in the k^{th} neighborhood is done by choosing k times 2 clusters at random and randomly reassign their observations to one or the other cluster. This shaking is a local perturbation that only affects few clusters. By the rules of VNS, if the local search does not succeed after l_{max} shakings and local searches, the shaking magnitude (k) is increased by 1. The stopping criterion may either be $k = k_{max}$ or the total cpu time used.

5 Description of the DyDA Program

In order to easily understand the Eye and Pen data, the first task achieved was to build a specific program that allows a visualization of the data in high dimension with an accurate representation of the time component. The main purpose of DyDA is to provide the researcher a graphical interface that helps understanding the whole process underlying the data. To achieve this goal, the program needs

– to take the dynamical aspect into account,
– to show many features at the same time,
– to show correspondence between different variables.

5.1 Using DyDA to Have a Dynamic Overview of the Process

As the first interest for the researcher is a spatial view of the data, one may, as illustrated on Figure 1, plot position of the pen and position of the eye on the same window.

An animation showing the evolution of the eye and pen position during time improves the reading of the data but is still difficult to follow. The alternative we propose here is to continuously show a "slice" of data corresponding to one or few seconds so that a reasonable amount of information is displayed at the same time. Wholes seconds of the process may then be understood by a single look as we can see on a screen capture represented on Figure 2.

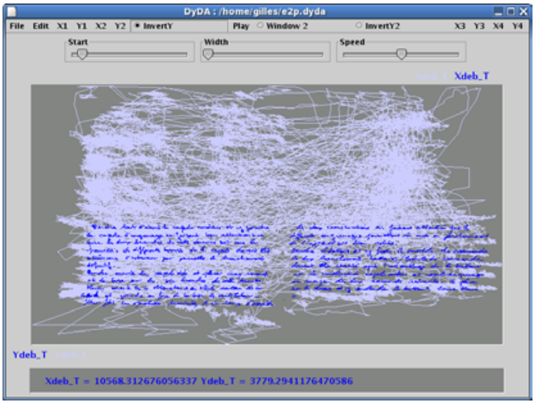

Fig. 1. View of the main window of DyDA showing the movements of the eye and pen.

5.2 Increasing the Dimension of the View and Exhibiting Correspondence Between Variables

In order to have a good understanding of the dynamics of writing, at each time, the researcher needs to see the position and pressure of the pen, the position of the eye and eventually some combination of them. As it would be difficult to show pressure or any different information on the same window as the spatial representation, a second window was added to DyDA. On that second window may be drawn speed and the pressure of the pen. The right window on Figure 2, shows a slice of the writing and the corresponding trajectory of the eye while the left window displays pressure against the speed of the pen. This figure clearly

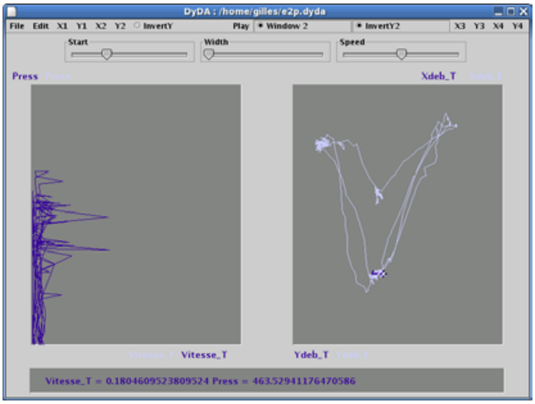

Fig. 2. A "slice" with pressure against speed of the pen curve on the first window and the corresponding written sequence as well as the position of the eye on the second one.

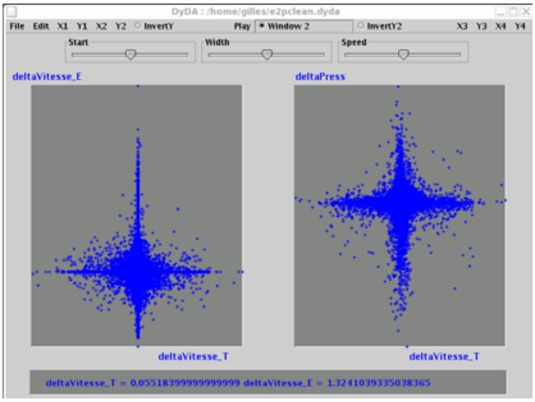

Fig. 3. variation of the pen speed against variation of the eye speed or variation of pressure.

shows that the pressure is rather high but varies while the pen is slow but still writing. On the right window we notice that the writer does not look at his writing (this is a parallel sequence).

To exhibit the relation between the different points representing the same data, the user may select and highlight some data on the main curve of the first window and see the corresponding points highlighted on the other curves. as illustrated on Figure 4.

6 Application Examples and New Results

The analysis of the data was achieved in two steps; the DyDA program alone was first used to have low level (almost mechanical) overview of data. Clustering was then applied to have a deeper understanding of the various activities involved at a higher level (more related to cognitive activities).

6.1 First Overview of the Data with DyDA

Showing Dependencies Between Eye and Pen Variations

Some intuition about the process came from drawing variation of eye or pen speed as well as pressure of the pen one against the other. Indeed, Figure 3 shows the variation of the eye speed (left window) and variation of pressure (on the right window) against the variation of the pen speed. This figure suggests that variation of the pen speed very rarely occurs simultaneously with the variation of eye speed or pressure.

Our attention was thus driven to the study of the variations of speed and pressure. The analysis of the dynamic process indicates that the eye acceleration usually occurs in three cases:

– when a parallel sequence occurs, in which case the pen speed does not vary much, even if it slows down a little,
– when the pen is paused and the eye looks away,
– and when the pen is about to be repositioned. In this case, the eye moves first to the future position of the pen and this last one only moves after.

Exhibiting Sequences of Parallel Processing

In most studies, the presence of parallel processes during writing is more discussed than objectively demonstrated [7]. One of the interest of the Eye and Pen paradigm is to bring out specific periods during which ones a new visual information is process in parallel with graphomotor execution. During these periods, the transcription of a visual information is accomplished while the eye is looking for a new information to be translated [2]. By choosing to highlight data with large distance between eye and pen (*i.e.* the pen is out of the parafoveal vision) and with pen speed greater than 0 (the pen is writing), it is possible to exhibit so called parallel sequences. The Figure 4 shows how DyDA may be used to attest those parallel sequences. Few seconds are needed to exhibit them precisely. Once these sequences are selected, the user may decide to activate the animation and see in real time the process of writing with the selected sequences highlighted. As long as the selection is not erased, it is possible to switch to the single window view or to display other variables, eventually while the animation is still running.

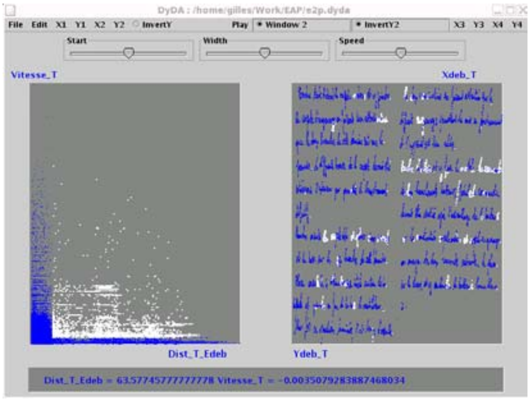

Fig. 4. DyDA with parallel sequences highlighted.

6.2 Classifying Different Phases of Visual Activities During Writing

To identify cognitive activities related to the handwriting task, we first needed to "clean" the data. Indeed, most variance of the raw data refers to rather mechanical components. For example, nystagmus, the constant eye movement around the watched object is not of great interest at a cognitive level. Indeed all eye movements with velocity smaller than 30 degrees per second were aggregated into a single point: the barycenter of all the corresponding eye positions. This threshold of 30 degrees per second is commonly used by researchers working on that topic to define the so called fixations reflecting the position the participant is willing to look at (for example, see [11]). A now important dimension in the data is the duration of each fixation as well as the distance the eye has achieved between fixations. As each observation is associated to a given clock time but not to a period of time, or sequence, the euclidian metric is appropriate. Indeed this would not be the case if each observation was associated to a sequence, as it is the case in hand writing recognition [20][3].

The first attempt to find information in the data with clustering methods was achieved using the centroid based model. Unfortunately, it did not provide any understandable clusters. The second step was to use *hyperplane* or *k-plane* clustering. If the results were more encouraging, the only information extracted by this mean was of the kind: either the pen (or the eye) is stopped, or its speed is almost constant. As a third attempt we used the clusterwise regression model. To use such a method, one needs a dependent variable. The distance achieved by the eye being related to the kind of composing activity conducted by the writer we decided to try clusterwise regression with eye distance as dependent variable. After running the program for 2 to 7 clusters, we noticed that each additional cluster, up to 4, yields an important error reduction. However, this is no more the case when increasing further this number of clusters, suggesting that the structure of the data consists in 4 clusters.

Conducted for each cluster, the analysis with DyDA of the temporal characteristics of visual and of graphomotor movements indicates four kinds of eye behavior.

1. Moving away from the pen with high speed, the eye can next stay relatively static, making few and long fixations on specific part of the sources before finally quickly joining the pen. This behavior should attest the writer's necessity to encode and to generate a new piece of content which could be immediately translated.
2. Still far from the pen, the eye can also explore, in a very dynamic way, the different parts of the source, by doing numerous fixations, moving with a very high speed and browsing on a large distance. This second kind of behavior could underly an important planning activity occurring during a long writing pause (the writer explores the source to plan a large part of text to be produced). It could also be due to a revision activity that consists in verifying agreement between the text produced so far and the source's information. In this case, the eye quickly browses the source as well as the text (to read). Two periods corresponding to such a revision may be described in the protocol.
3. During graphomotor execution, the eye is generally close to the pen (except during a period of parallel processing). In this case, the writer accomplishes long and static fixations, moving slowly to rejoin regularly the pen. This behavior can be interpreted as an ocular control of graphomotoric execution.

4. In some cases, an unexpected phenomenon occurs. When the eye comes back from an intensive source exploration, or when the pen moves quickly from the end to the beginning of a line, the eye can make a long fixation, relatively far from the pen executing the trace, but also far from source's information. This behavior could correspond to a peripheral (or parafoveal) monitoring of the trace. Here it is interesting to note that data-mining could be helpful to bring out two kinds of relationships between eye and graphomotor execution: a local control and a peripheral monitoring.

7 Conclusion and Future Work

The Eye and Pen device and DyDA software represent key elements to study the dynamics of writing from source. They constitute a powerful computer environment able to catch very fine grained phenomena related to ocular and graphomotor activities. They also provide to the researcher an easy-to-use tool helping to understand the temporal and the spatial relationships between the behavioral indicators. Used in association with Eye and Pen and DyDA, data-mining and discovery methods offer interesting perspective regarding the nature of recorded data. In our study, data-mining is used to infer the upstream cognitive processes. By now, it has been used to categorize different temporal patterns in the graphomotoric execution of written words and one main result can already be stressed: clusterwise regression appears to be very relevant to segment and categorize the flow of writing from source. By isolating four different relationships between the visual input and the graphomotoric output of writing, the clusterwise regression allows us to interpret four different patterns in the writing activity. One of them, the fourth cluster, appears to be original and its cognitive signification to be precised. Mostly descriptive, these first results must be reinforced and validated by systematical investigations.

The future work will aim, first, at analyzing the dynamics of the four writing processes configurations, that is to say the order in which these configurations occur all along the text composition; second at comparing these sequences in various contexts either involving different participants or different kinds of writing tasks. Indeed, further data mining techniques such as dynamic time warping will then need to be used to handle correctly the temporal dimension and its distortion from one writer to another.

In every cases, these issues give a great insight of the possibilities given by the machines to explore new dimensions in human cognitive activities.

Acknowledgments

The authors thank C. Ros - from laboratory LaCo-CNRS - Université de Poitiers and C. Dansac from laboratory LTC-CNRS Université de Toulouse for their contribution on the data analysis. This work was supported by NSERC - Canada and ACI-MSHS - France.

References

1. D. Alamargot and L. Chanquoy. *Through the models of writing.* Dordrecht-Boston-London: Kluwer Academic Publishers, 2001.
2. D. Alamargot, C. Dansac, and D. Chesnet. Parallel processing around pauses: A conjunct analysis of graphomotor and eye movements during writing instructions. In M. Torrance, D. Galbraith, and L. v. Waes, editors, *Recents developpements in writing-process research (Vol. 2).* Dordrecht-Boston-London: Kluwer Academic Press, In press.
3. C. Bahlmann and H. Burkhardt. The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(5), 2004.
4. P.S. Bradley and O.L. Mangasarian. k-plane clustering. *Journal of Global Optimization*, 16:23–32, 2000.
5. G. Caporossi and P. Hansen. Finding Relations in Polynomial Time. In *Proceedings of the XVI International Joint Conference on Artificial Intelligence*, pages 780–785, 1999.
6. G. Caporossi and P. Hansen. Variable Neighborhood Search for Extremal Graphs: 5. Three Ways to Automate Finding Conjectures. *Discrete Mathematics*, 276:81–94, 2004.
7. L. Chanquoy, J.N. Foulin, and M. Fayol. Temporal management of short text writing by children and adults. *Cahiers de Psychologie Cognitive*, 10(5):513–540, 1990.
8. D. Chesnet and D. Alamargot. Analyses en temps réel des activités oculaires et graphomotrices du scripteur: intérêt du dispositif 'eye and pen'. *L'Année Psychologique*, In press.
9. D. Chesnet, F. Guillabert, and E. Espéret. G-studio: un logiciel pour l'étude en temps réel des paramètres temporels de la production écrite. *L'Année Psychologique*, 94:115–125, 1994.
10. J. Grabowski. Writing and speaking: Common grounds and differences. toward a regulation theory of written language production. In *The Science of Writing: Theory, methods, individual differences and applications*, pages 73–91. Mahwah (NJ): Laurence Erlbaum Associated, 1996.
11. Z.M. Griffin and K. Bock. What the eyes say about speaking. *Psychological Science*, 11(4):274–279, 2000.
12. D.J. Hacker. Comprehension monitoring of written discourse across early-to-middle adolescence. *Reading and Writing*, 9(3):207–240, 1997.
13. P. Hansen and N. Mladenović. Variable neighborhood search: Principles and applications. *European Journal of Operations Research*, 130:449–467, 2001.
14. P. Langley. The computer-aided discovery of scientific knowledge. *Proceeding of the First International Conference on Discovery Science*, 1998.
15. Kin-Nam Lau, Pui-Iam Leung, and Ka-Kit Tse. A mathematical programming approach to clusterwise regression. *European Journal of Operations Research*, 116:640–652, 1999.
16. C.M. Levy and S. Ransdell. *The science of writing: Theories, methods, individual differences, and applications.* Hillsdale (NJ): Laurence Erlbaum Associated, 1996.
17. N. Mladenović and P. Hansen. Variable neighborhood search. *Computers and Operations Research*, 24:1097–1100, 1997.
18. H. Spaeth. Clusterwise linear regression. *Computing*, 22:367–373, 1979.

19. R.E. Valdés-Peréz. Principles of Human Computer Collaboration for Knowledge Discovery in Science. *Artificial Intelligence*, 107:335–346, 1999.
20. M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh. Indexing multi-dimensional time-series with support for multiple distance measures. In Lise Getoor, Ted E. Senator, Pedro Domingos, and Christos Faloutsos, editors, *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2003.