

Pensum, un système d'aide à la compréhension de cours à distance

Mathieu Loiseau, Damien Dupré, Philippe Dessus

▶ **To cite this version:**

Mathieu Loiseau, Damien Dupré, Philippe Dessus. Pensum, un système d'aide à la compréhension de cours à distance. EIAH' 2011, May 2011, Mons, Belgique. pp.287-299. hal-00598788

HAL Id: hal-00598788
https://hal.science/hal-00598788

Submitted on 7 Jun 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Pensum, un système d'aide à la compréhension de cours à distance

Mathieu Loiseau*,*, Damien Dupré**, Philippe Dessus*****

** LIDILEM*
UFR des Sciences du Langage
Université Stendhal, BP 25
38 040 Grenoble Cedex 9
mathieu.loiseau@msh-alpes.fr
*** Laboratoire Interuniversitaire de Psychologie, Université de Savoie,*
Chambéry et Université Pierre-Mendès-France, Grenoble
1251 av. Centrale
Université Pierre-Mendès-France, BP 47
38 040 Grenoble Cedex 9
Damien.Dupre@etu.upmf-grenoble.fr
**** Lab. sciences de l'éducation UPMF Grenoble-2 & UJF Grenoble-1*
1251 av. Centrale
Université Pierre-Mendès-France, BP 47
38 040 Grenoble Cedex 9
Philippe.Dessus@upmf-grenoble.fr

RÉSUMÉ. *Cet article a pour but de présenter Pensum, un système permettant à des étudiants à distance de produire des synthèses de cours dans le but de les comprendre, et d'en avoir des retours automatiques fondés sur une analyse sémantique. Nous décrivons une première utilisation de ce logiciel, auprès de 17 étudiants de 2e année de Master en sciences de l'éducation, ayant soit le rôle d'apprenant soit le rôle de tuteur, et analysons trois types de données : traces, bugs et questionnaire d'opinion. Les résultats montrent que les participants parviennent rapidement à utiliser les principales fonctionnalités de Pensum et que, s'il reste encore à le tester en situation réelle, son utilisabilité permet d'en faire un outil pouvant promouvoir l'apprentissage autorégulé de cours à distance.*

MOTS-CLÉS : *Production de synthèses de cours, Compréhension, LSA (Analyse sémantique latente), retours (feedback), validation, enseignement à distance*

1. Introduction

À l'université, la production écrite (notamment via les essais) est encore le passage obligé de toute évaluation de la compréhension d'un cours [VALENTI et al. 03]. Il existe de nos jours des outils d'aide à la production de texte de plus en plus sophistiqués, et les étudiants, notamment ceux à distance, peuvent utiliser ces outils pour corriger (semi-) automatiquement certaines caractéristiques de leurs productions (p. ex., liées à la présentation et la mise en page, l'orthographe ou la grammaire). Mais ce niveau de correction est très focalisé sur la forme et comporte deux principaux problèmes : il n'a tout d'abord qu'un effet faible sur ce type de performance en écriture [MACARTHUR 06] ; et la plupart des outils les plus récents d'aide aux étudiants sont en réalité plutôt une aide à l'enseignant (p. ex., les outils de détection de plagiat).

Dans un contexte d'enseignement à distance, les étudiants sont engagés dans un flux de travail qui, en règle générale, comprend les activités suivantes, dans lesquelles ils sont librement engagés. Tout d'abord, les étudiants peuvent *lire* un ensemble de documents de cours, qu'ils peuvent au besoin annoter, pour en fixer la compréhension. Ensuite, ils peuvent *écrire* des dissertations ou synthèses à partir de questions liées à ce cours, ou bien prendre des notes concernant ces derniers. Enfin, ils ont diverses interactions avec l'enseignant ou le tuteur pour une évaluation de la compréhension du cours, soit via courriel ou un forum.

Les problèmes liés à ce flux de travail sont assez nombreux, autant du côté des apprenants que des enseignants ou tuteurs. Les étudiants savent que les interactions avec leurs enseignants ou tuteurs vont être limitées, ce qui ne les encourage pas à demander des retours réguliers sur leur travail et les amène à s'apercevoir souvent trop tard (lors de l'évaluation) qu'ils n'ont pas compris tel ou tel point du cours. Ils peuvent de plus trouver parfois plus fructueux de recourir au plagiat, surtout s'ils comprennent mal le cours. Ensuite, ils attendent souvent longtemps les retours de leurs enseignants-tuteurs, qui considèrent souvent les demandes d'aide de la part des étudiants comme des intrusions, qu'ils doivent traiter en urgence, ce qui les amène à avoir une gestion non proactive des demandes des étudiants. Enfin, ils passent beaucoup de temps à corriger les travaux des étudiants en prenant en compte de très nombreux paramètres, ce qui alourdit leur charge de travail.

Ces problèmes ont amené des chercheurs à proposer de nouvelles activités de travail et de production pour essayer de les pallier (p. ex., la production de portfolios ou l'apprentissage fondé sur les problèmes). Une autre possibilité, que nous explorons, est de procurer à l'étudiant des retours sur sa production liée à la lecture le plus tôt possible (c'est-à-dire avant les interactions avec le tuteur ou l'enseignant). Cette solution a pour avantage de mettre l'étudiant directement en activité, et le laisse libre d'entreprendre à son rythme les deux boucles de lecture/écriture, en lui faisant tout de même produire des textes libres (de synthèses, d'essais) souvent négligés au cours de la scolarité [TROIA 07]. Il est donc intéressant de travailler à procurer à l'étudiant des outils se centrant sur des aspects d'assez haut niveau, utilisant par exemple des méthodes de traitement automatique de la langue, afin de procurer des retours immédiats sur des aspects de haut niveau des productions des étudiants. Dans cet article, nous présentons *Pensum*, un système permettant à un étudiant à distance d'être justement engagé dans une boucle de lecture/écriture à propos de cours, et

les premiers éléments d'une validation de ce logiciel en contexte de laboratoire, auprès d'étudiants de 2e année de Master professionnel en sciences de l'éducation.

2. Principes théoriques : Écrire des synthèses pour apprendre en autorégulation

Les principes théoriques qui ont présidé à la conception de *Pensum* sont liés au fait qu'il est important que l'étudiant, s'il souhaite comprendre un document du cours, réalise une activité qui remplisse au moins les quatre caractéristiques suivantes.

Tout d'abord, il doit être relativement libre dans son travail, et il doit lui être possible, à tout moment, de prendre une part active à son évaluation, et porter un regard réflexif sur ce qu'il a écrit, ou sur les jugements d'autrui sur son propre travail [BUTLER & WINNE 95]. Un système d'aide à l'apprentissage promeut *l'apprentissage autorégulé* (*self-regulated learning*) si et seulement s'il laisse l'apprenant libre de réaliser l'activité qu'il souhaite pendant la durée voulue et qu'à tout moment il puisse en changer. *Pensum* est justement organisé de manière à favoriser un apprentissage auto-régulé, en favorisant l'insertion de l'apprenant dans plusieurs *boucles* [ZIMMERMAN 00]. L'apprenant peut ainsi à tout moment passer d'une boucle dans laquelle il va lire tour à tour des documents du cours à une autre dans laquelle il va produire une synthèse de tout ou partie de ces documents (aucune contrainte particulière ne lui étant donnée quant aux caractéristiques de la synthèse). À tout moment, également, l'apprenant peut agir sur les retours proposés par *Pensum*. Comme le signale un auteur [TROIA 07, p. 136], l'autorégulation est bénéfique, entre autres, « parce qu'elle permet à l'apprenant d'être plus attentif aux qualités et limites de son écriture, et donc d'adopter des stratégies plus adéquates ».

Ensuite, son activité doit être suffisamment complexe pour enrichir ses connaissances [HÜBNER et al. 06] : par exemple *une synthèse*, qui permet de mettre en relation différentes parties du cours. Un apprenant, lorsqu'il apprend son cours, sera enclin à produire de l'écrit en vue de la compréhension de celui-ci. Ces écrits sont de complexité différente selon leur longueur et leur structure, et on pourrait les lister ainsi : – des annotations brèves et circonscrites ; – des résumés partiels d'un paragraphe ou une section ; – des synthèses plus complètes. *Pensum* place l'apprenant en position d'élaborer des synthèses pour l'apprentissage de cours selon l'approche « production de documents de sources multiples » [SEGEV-MILLER 04], activité qui peut, par sa complexité, lui permettre d'acquérir des connaissances dans le domaine choisi.

Cette activité sera préférentiellement réalisée à l'écrit (démarche d'écrire-pour-apprendre[KLEIN 99]), car l'écrit est l'une des activités centrales dans l'enseignement universitaire, qu'il soit en présence ou à distance. L'étudiant qui a recours à l'écrit pour noter, annoter son cours prend ainsi du recul sur ce cours et analyse sa propre compréhension.

Pour finir, il est nécessaire que l'étudiant puisse porter un regard réflexif sur son propre travail. De manière assez inattendue, le fait qu'aucun système automatisé ne puisse produire des retours fiables à 100 % pourrait permettre également ce regard. En effet, dès lors que l'on s'attaque à l'analyse sémantique, il est illusoire de penser pouvoir atteindre des résultats parfaits [ANTONIADIS 04]. Pourtant, comme le signale [LEBARBÉ 10], le Traitement automatique des langues doit encore « se détacher d'une image d'omnipotence répandue par les enthousiasmes de ses premiers pas (et par la littérature de science fiction) »

(*id*., p. 19). En effet, s'il est banal, dans un contexte de recherche, de souligner que les différents retours proposés par *Pensum* seront parfois éloignés de ceux d'un expert humain, ce ne sera pas nécessairement aussi évident pour un apprenant. Or la conscience des limites et des objectifs d'un système préside à l'adoption de ce dernier [MURRAY & BARNES 98] et à sa bonne utilisation [BAX 03] (notons que ces deux dernières publications proviennent du champ de l'Apprentissage des langues assisté par ordinateur (ALAO), mais les conclusions qu'elles présentent nous semblent tout à fait adaptables à notre problématique, pour ne pas dire généralisables aux EIAH en général). C'est dans l'optique de rendre accessible à l'utilisateur la faillibilité du feedback que nous avons décidé, pour cette version du système, d'ajouter une fonctionnalité de remise en question des retours. Outre les raisons évoquées et l'intérêt évident d'une telle fonctionnalité dans la perspective d'améliorer le système ultérieurement, il nous semble que son intérêt majeur est justement de permettre à l'apprenant d'arriver à porter ce regard réflexif sur son travail. Le simple fait que l'apprenant puisse critiquer, mettre en question, les retours qui lui sont faits lui permettront de pouvoir mieux analyser, de manière critique, ses écrits et, par là, la compréhension de ses cours. De plus, l'apprenant est sans doute mieux à même de pouvoir critiquer une machine plutôt que l'enseignant, ce qui rend cela plus aisé. Ainsi, tout retour élaboré par *Pensum* peut être contesté et justifié librement par l'étudiant.

3. *Pensum* : fonctionnalités et conception

3.1. Les trois types de feedback

La fonction première de *Pensum* est donc de proposer à l'apprenant un feedback automatique à la demande, qui lui permette une réflexion sur ses propres écrits. À l'heure actuelle, trois caractéristiques textuelles de l'écrit des apprenants (cohérence, importance, et hors-sujet) sont automatiquement analysées par *Pensum*. Toutes trois sont reliées à des critères importants issus de la psychologie de l'apprentissage de l'écrit ou d'un domaine particulier. Il a par exemple été montré que l'écriture d'un texte *cohérent* (i.e., dont les phrases et idées s'enchaînent correctement) est liée à la compréhension du domaine de ce dernier [TAPIERO 07] : *Pensum* indique à l'utilisateur ce qu'il considère comme des sauts de cohérence au moyen du symbole '‖'. Afin de garder une trace des différentes interactions, si l'apprenant conteste un tel feedback, ce symbole sera remplacé par '‖« (cf. partie synthèse de la figure 1). De plus, la prise en compte par l'apprenant des *principales idées* d'un cours est également importante, et *Pensum* lui signale quelles sont les phrases du cours qui ne sont pas encore sémantiquement présentes dans le texte de sa synthèse dans la partie cours de son interface. On constate, dans la figure 1 quatre couleurs pour les phrases du cours, celles-ci sont issues des interactions ayant eu lieu avec l'utilisateur. Une phrase indiquée en noir après un feedback a été considérée par le système comme prise en compte. Une phrase en rouge est considérée comme non prise en compte par l'apprenant, qui peut la contester de deux manières :

- la phrase n'est pas pertinente (cas de « l'annuaire… » dans la figure 1), elle sera ensuite affichée en gris ;
- la phrase est pertinente et a été prise en compte (l'utilisateur devra le justifier, cf. figure 2), elle sera affichée en bleu.

Comme les feed-back des versions précédentes sont conservés, une phrase qui a été considérée comme non pertinente sera indiquée comme telle, jusqu'à ce que l'apprenant revienne sur son jugement.

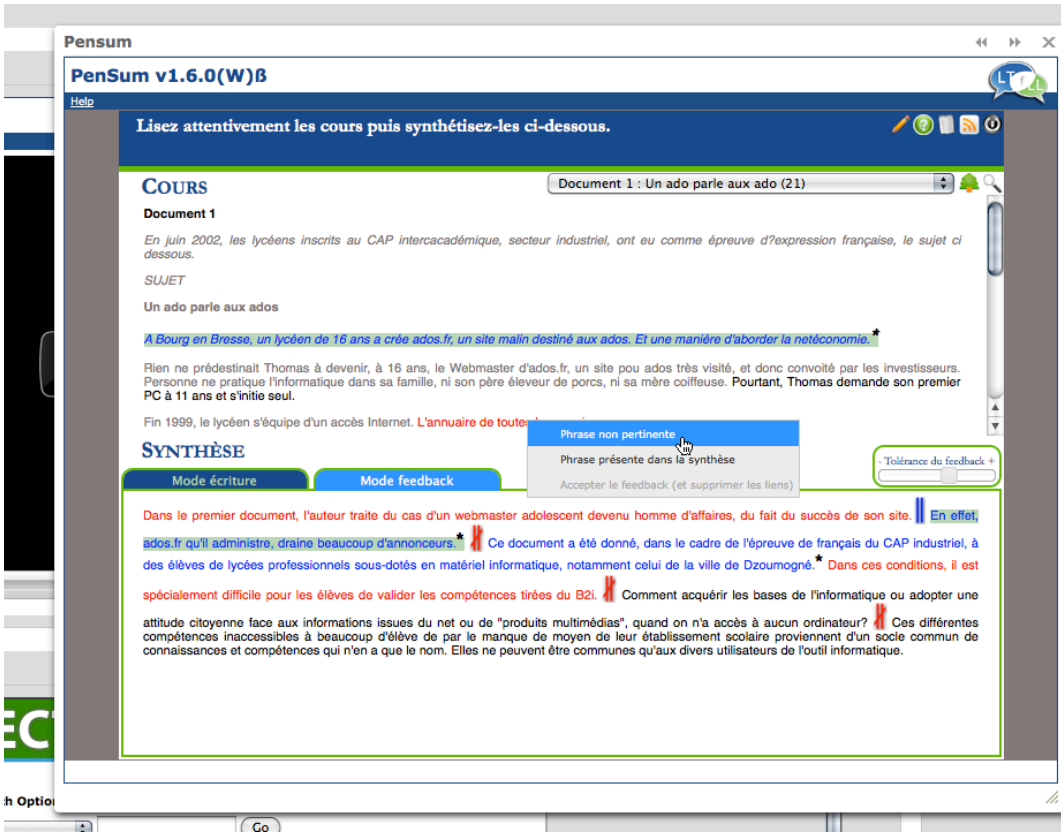


Figure 1. Interface de Pensum : l'utilisateur conteste un feedback « omission » qui concerne une phrase qui ne lui semble pas pertinente (version Widget)

Comme la capacité d'un étudiant ou élève à organiser et synthétiser correctement ses idées, sans réaliser de *hors-sujet*, est importante, *Pensum* les lui indique dans la partie « Synthèse » de l'interface. De la même manière, l'utilisateur peut accepter ou refuser un retour (dans ce dernier cas, la phrase est alors colorée en bleu). Il est à noter, enfin, que toutes les rétroactions de l'apprenant pour les trois types de feedback sont consignées et reportées dans les versions ultérieures de la synthèse (pour les phrases qui restent identiques).

3.2. L'Analyse sémantique latente pour produire des feed-back

Pour proposer ces rétroactions, *Pensum* s'appuie sur l'analyse sémantique latente (LSA), qui est une méthode statistique se fondant sur une théorie de l'induction et de représentation de la connaissance [LANDAUER & DUMAIS 97] et qui permet d'évaluer la similarité entre le sens de deux mots ou de deux portions de textes. Pour ce faire, LSA analyse un grand corpus et crée un espace vectoriel dans lequel chaque mot est représenté comme un point, en fonction des autres mots à proximité desquels il apparaît. LSA a été utilisé pour l'évaluation globale de dissertations selon différentes stratégies [KINTSCH et al. 07], mais notre objectif est ici de tenter de proposer un retour portant sur des entités plus élémentaires, en l'occurrence la phrase. Or, LSA permet non seulement de comparer la proximité

sémantique des mots, mais aussi des portions de texte, comme des paragraphes, des phrases ou d'autres groupes de mots.

Pensum utilise ces propriétés et s'appuie sur des travaux antérieurs [LEMAIRE et al. 05]. Le système propose donc de comparer les phrases des travaux des étudiants avec celles des textes-sources (détection du hors-sujet) et *vice versa* (phrases non prises en compte dans les productions des étudiants), ainsi que les phrases consécutives des synthèses (cohérence interphrase). Si le corpus servant de base à l'apprentissage a été bien construit, la méthode permet d'établir des similitudes entre des textes en surface très différents. Mais la puissance de LSA en est aussi la faiblesse : pour offrir les meilleurs résultats, l'entraînement doit être effectué avec un corpus généraliste (pour « apprendre » la langue), mais aussi des textes du domaine à traiter. LSA sera plus ou moins performant selon le texte utilisé et le corpus d'entraînement utilisé. Il est donc important de s'assurer que le feedback peut être réglé (figure1) et même remis en cause par l'apprenant, mais peut également « encourager le dialogue avec les enseignants et les pairs autour de l'apprentissage » [NICOL & MACFARLANE-DICK 06, p. 205]. C'est pourquoi nous offrons aux apprenants la possibilité de justifier leurs remises en question de certains feed-back (omission et hors-sujet, cf. figure 2).

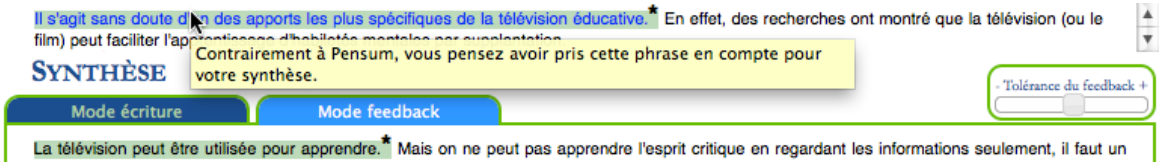


Figure 2. Feedback (omission) remis en question par l'apprenant avec justification (Pensum version Widget)

En effet, bien que l'interface d'administration de *Pensum* soit extrêmement rudimentaire, elle permet déjà de tirer parti de la conservation des versions successives des synthèses ainsi que des rétroactions de l'apprenant sur le feedback. Ainsi, un tuteur confronté à l'exemple ci-dessus pourra initier avec l'apprenant un échange, afin de savoir pourquoi il a remis en question le feedback proposé (alors qu'il ne semble *a priori* pas traiter l'essentiel de l'information de la phrase considérée).

4. Présentation de l'étude de validation

4.1. Historique des versions de Pensum

L'étude décrite ici est la première validation de *Pensum* (version 1.5), qui s'appuie sur la version 1 [VILLIOT-LECLERCQ et al. 10], et y ajoute toutes les fonctionnalités de contestation des feed-back ainsi que la sauvegarde de toutes les versions de chaque synthèse. L'autre mise à jour majeure entre la version 1 et la version 1.5 est l'utilisation d'une nouvelle implémentation de LSA. En effet, la version utilisée précédemment n'étant plus maintenue, elle nécessitait l'installation d'un système d'exploitation spécifique et suranné. La nouvelle implémentation [WILD 07], réalisée en *R* (http://www.r-project.org/), est légèrement moins performante en termes de rapidité d'exécution (qui n'a malgré tout pas été altérée, du fait d'une optimisation dans l'algorithme de calcul), mais beaucoup plus portable. Les modifications des versions ultérieures ne concernant pas les fonctionnalités

centrales du système (cf. section 3.1), nous proposons, dans cet article, les saisies d'écran de la version la plus récente.

4.2. Contexte

Le but de l'étude de validation en question est avant tout exploratoire. Il s'agit de détecter les éventuels *bugs* restants, de se rendre compte si le serveur gère correctement la montée en charge des requêtes, mais aussi de voir comment des utilisateurs-cibles peuvent l'utiliser et se l'approprier. Dix-sept étudiants (dont 15 femmes) de Master 2e année professionnel en sciences de l'éducation (moyenne d'âge : 33 ans) ont participé volontairement à cette étude et ont été dédommagés par une clé USB. Ce test a été réalisé dans le cadre d'un TP d'un cours sur les outils cognitifs d'aide à l'enseignement, à l'université Pierre-Mendès-France de Grenoble. Ils ont été répartis aléatoirement en deux groupes : – le groupe des « étudiants », dont la tâche était de réaliser, pendant environ 1 h, une synthèse d'un ensemble de documents de cours ; – le groupe des « tuteurs » qui a pour rôle de bien comprendre le contenu du cours (en en dégageant les idées principales) puis d'observer la fin de la rédaction de la synthèse de leur « étudiant », sans intervenir. Malgré son caractère artificiel, ce scénario permet de répliquer un contexte classique d'enseignement à distance.

4.3. Procédure

Pour commencer, les principaux buts de l'étude ont été présentés aux participants, ainsi qu'un avertissement sur le caractère optionnel de cette dernière, puis tous les participants ont suivi une présentation de quatre pages décrivant pas à pas le fonctionnement et les principales fonctionnalités de *Pensum*. Ensuite, les étudiants ont passé environ une heure à écrire la synthèse d'un cours composé de trois documents sur l'informatique éducative (nombre de mots du doc. 1 : 458, du doc. 2 : 1 315, du doc. 3 : 393, l'ensemble totalisant 4 pages), en utilisant *Pensum 1.5*. L'ensemble des documents (15 pages) était également reproduit sur papier. À tout moment les étudiants pouvaient reporter sur une feuille d'identification de *bugs* les problèmes rencontrés avec le logiciel (dont les colonnes étaient les suivantes : – But (ce que je cherchais à faire) ; – Problème rencontré ; – Solution trouvée (le cas échéant).

Pendant ce temps, leurs tuteurs avaient pour consigne – de lire les documents du cours et de surligner les passages importants ; – de suivre ensuite ce que leur étudiant faisait, sans intervenir. À la fin de la phase d'écriture de la synthèse, les tuteurs devaient lire extensivement la synthèse produite par leur étudiant, et répondre à quatre questions en utilisant une page spécifique composée de quatre champs de texte, le « bloc-notes ». Les questions du bloc-notes sont les suivantes : – La synthèse reprend-elle bien les éléments importants des différents cours ? ; – La synthèse contient-elle des éléments inutiles, peu importants ? ; – L'enchaînement des différentes phrases de la synthèse vous paraît-il correct ? ; – Note sur 20 et appréciation générale. Pour finir, les étudiants revenaient sur l'ordinateur pour lire ce qui avait été écrit par leurs tuteurs et une conversation s'engageait afin d'approfondir les commentaires. Pour finir, les participants ont rempli un questionnaire composé de questions ouvertes sur leur opinion à propos de *Pensum*.

4.4. Données recueillies

Trois types de données ont été recueillis afin de mesurer l'utilisabilité de *Pensum* : – *l'analyse des traces*, dans laquelle les actions des étudiants dans *Pensum* sont enregistrées (nombre de feed-back demandés, longueur de la synthèse sous ses différentes versions, types de feedback fournis, nombre de contestations de feedback, etc.) ; – l'*analyse des bugs* recueillis par les participants et l'expérimentateur ; ces indications ont ensuite été directement envoyées au programmeur de *Pensum* qui s'est chargé de les résoudre ; – *l'analyse de l'opinion des participants,* récupérée par le questionnaire, qui a permis de mettre en évidence diverses informations subjectives sur son utilisabilité.

5. Résultats

5.1. Analyse des traces et des bugs

Les traces des actions des étudiants dans leurs tâches dans *Pensum* nous permettent de comprendre leurs comportements. Le tableau 1 ci-dessous détaille les données des traces (les données de l'étudiant 6 ont été supprimées du fait la pauvreté de sa synthèse et de l'absence de feedback demandé). Les participants ont écrit une synthèse plutôt longue (ligne 1 du tableau). Il est à noter que les étudiants ont demandé de nombreux retours à *Pensum*, de manière régulière (ligne 2), et surtout en ajustant au besoin sa sévérité (ligne 3). Le feedback que les participants ont le plus pris en compte est celui de la cohérence entre phrases (ligne 8), vraisemblablement parce que c'est celui qui comporte le plus d'alertes (ligne 5), mais aussi qui est le plus aisé à remettre en question. Notons enfin que les participants ont peu justifié leur travail en faisant des liens cours-synthèse (ligne 9), peut-être parce que la procédure est plus difficile à mettre en œuvre.

Le détail du travail entrepris par le participant 7 nous permet de mieux comprendre comme *Pensum* peut être utilisé. Ce dernier a commencé à utiliser le feedback de *Pensum*, puis ne l'a plus utilisé pendant une longue période d'écriture. Sa dernière demande de feedback met au jour un nombre très important de phrases non cohérentes : il aurait été libre de choisir une stratégie de correction au fur et à mesure de l'écriture en demandant des feed-back plus réguliers. Il serait intéressant de savoir s'il s'agit d'une stratégie explicitement choisie et justifiable ou s'il s'agit d'une décision plus implicite. Le caractère exploratoire de cette étude et la durée de la passation nous empêchent de faire des analyses plus approfondies, qui seront réalisées dans une étude impliquant plus de participants et sur une durée plus longue, qui leur laisse plus de temps pour s'approprier l'outil. Toutefois, dans les conditions de l'expérimentation, très peu de problèmes ont été rapportés par les participants, ce qui ne permet pas d'évaluer la robustesse du système, mais indiquerait que les utilisateurs ne se trouvent pas désarmés face à l'interface et aux fonctionnalités, ce qui semble confirmé par leurs dires.

	Identifiant Participant	1	2	3	4	5	8	7	9	*M*	*EC*
1	Nb de mots de la synthèse finale	396	362	149	306	542	278	463	222	339,8	128,2
2	Nb de feedback demandés	10	9	5	5	7	7	2	11	7	3
3	Tolérance moyenne du feedback	1,07	0,93	1	0,87	0,93	1	1	0,96	1	0,1
4	Nb min de feedback de cohérence	6	5	2	1	1	3	1	2	2,6	1,9
5	Nb max de feedback de cohérence	11	10	3	6	11	7	18	11	9,6	4,5
6	Nb min de feedback de hors-sujet	1	3	1	1	1	1	1	1	1,3	0,7
7	Nb max de feedback de hors-sujet	5	3	3	2	3	9	3	6	4,3	2,3
8	Nb de feedback de cohérence remis en question	1	0	3	1	4	4	4	0	2,1	1,8
9	Nb de liens synthèse-cours ajoutés	0	3	0	1	3	0	0	2	1,1	1,3

Tableau 1. Traces de l'utilisation de Pensum dans la tâche de synthèse pour chaque participant

5.2. Analyse de l'opinion des participants

L'opinion des participants a été recueillie grâce à une source principale : le questionnaire d'opinion. Commençons par l'impression générale sur *Pensum.* Cinq participants se sont déclarés satisfaits de son utilisation. Pour eux, *Pensum* est « pratique », « intéressant », « interactif » et « constitue une aide » permettant aux étudiants de « comprendre leurs cours ». Parallèlement, 10 participants ont une impression d'ensemble mitigée. Ils trouvent le logiciel « intéressant », que c'est une « bonne idée », qu'il est « pratique » et « facile ». Mais pour chacun d'entre eux, le logiciel « doit être amélioré » et présente des limites notamment sur la « pertinence » et la compréhension du feedback. Enfin, deux participants montrent une opinion négative. Le premier participant est « déçu » et met en avant un « investissement cognitif supérieur » et une nécessité d'améliorer « l'interactivité ». Pour le second, le logiciel « semble encore peu abouti » et des améliorations sont possibles en termes d'affichage, d'interactivité et de qualité des feed-back.

Du point de vue de son utilisabilité, les participants sont quasi-unanimes (sauf 1) quant à la simplicité et la facilité à utiliser *Pensum* (« facile » d'utilisation, « facile à comprendre », « assez simple »). Les participants ont reporté quelques suggestions pour améliorer *Pensum*. De manière générale, ils suggèrent d'avoir 1) plus d'actions possibles sur la synthèse du texte, 2) une explication plus approfondie du feedback, et 3) des indications plus précises sur les feed-back de cohérence. On peut donc constater que les participants sont capables, après une heure d'utilisation, d'émettre des suggestions pertinentes vis-à-vis du feedback. Ce dernier point montre leur intérêt, mais aussi le fait qu'ils ont clairement compris le potentiel et les limitations de *Pensum*.

Selon 7 participants, *Pensum* a un intérêt élevé dans l'apprentissage grâce à son aide de « récapitulation », pour « revenir sur des éléments peu compris », « repérer des points importants », « travailler sur la compréhension » ou « l'élaboration de fiches ». Conjointement, 6 participants ont une réponse mitigée. *Pensum* « peut » ou « pourrait » les aider à apprendre et/ou synthétiser (ou plutôt pour résumer) leurs cours, mais des éléments négatifs sont une nouvelle fois avancés. Ainsi « l'articulation logique semble limitée », la « mise en page » (impossibilité d'annoter le texte, de surligner, de modifier la police, abréviations) ne permet pas par exemple « l'apprentissage visuel ». De plus, les feed-back ne sont pas affichés dans la fenêtre de rédaction. Enfin, 4 des participants considèrent que *Pensum* ne

pourrait pas leur permettre d'apprendre. Ils concèdent toutefois avoir une méthode de travail incompatible (besoin de s'approprier le texte sur papier par exemple).

6. Perspectives et conclusion

La plupart des *bugs* relevés par les utilisateurs étaient liés à l'interface et ont été facilement corrigés à partir de la version suivante (1.6), dont le développement a commencé au moment de cette vérification : dans le cadre du projet LTfLL, il était nécessaire que le système puisse être exécuté comme un *Widget* intégré à une plate-forme *Elgg* (http://www.elgg.org/). Du fait de cette nécessité, la version suivante, bien qu'entreprenant quelques améliorations issues de cette vérification (découpage de phrases, lisibilité de l'interface) propose avant tout une refonte complète de l'architecture monolithique de la version 1.5 :

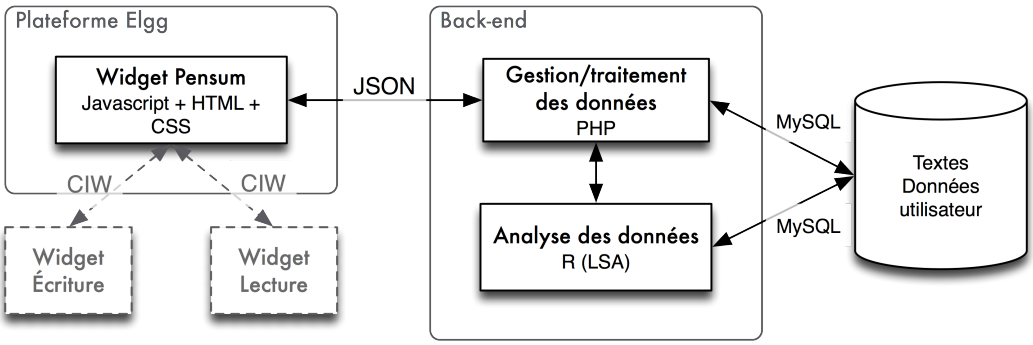


Figure 3. Architecture de Pensum v 1.6

Dans la version 1.6, tous les traitements sont effectués par un *back-end* va gérer tous les aspects de stockage et récupération des données, et appeler un web-service d'analyse des textes basé sur R-LSA. Le web-service récupère directement les données dans la base de données et génère un document XML, à son tour pris en charge par le module de gestion/traitement des données. L'interface du système est quant à elle conçue dans un module complètement indépendant qui utilise des technologies côté client afin de présenter les données à l'utilisateur et de gérer les interactions. Cette interface peut ensuite être utilisée de manière autonome, comme la version 1.5, mais également intégrée à une plateforme *Elgg*. Dans le cadre du projet LTfLL, le but de cette intégration était de faire communiquer *Pensum* avec d'autres widgets développés dans le cadre du projet. Mais cette communication inter-widgets (CIW, dans la figure 3) permettrait à terme d'envisager *Pensum* comme un widget de coordination et de gestion du feedback entre un widget d'écriture (comme *Multixinha*, http://marxjohnson.github.com/MultiXinha/) ou un widget d'affichage/prise de notes. Outre le fait d'accéder à des doléances des utilisateurs à moindre coût (p. ex., limites des fonctions d'édition), cette piste permet d'imaginer étendre les fonctionnalités et les objectifs de *Pensum* en intégrant celles offertes par ces *widgets* (comme la rédaction collaborative, grâce à *Multixinha*, par exemple).

Avant de pouvoir considérer ces pistes techniques, il reste encore à effectuer d'autres analyses pour déterminer si *Pensum* peut être un outil d'apprentissage de cours à distance performant (une validation dans un contexte réel est en cours d'analyse). Toutefois, cette première validation a permis, d'une part, de montrer qu'il fonctionne de manière

satisfaisante. Il sera également nécessaire de former mieux les utilisateurs à interpréter les rétroactions du logiciel, qui sont d'un type inhabituel pour eux. Cette formation pourrait traiter les trois points suivants : – une meilleure connaissance de la manière dont ces rétroactions sont produites, ce qui permettrait un retravail des synthèses plus efficace (p. ex., la notion de saut de cohérence, pour *Pensum*, ne concerne pas les connecteurs logiques, mais la proximité sémantique entre ces deux phrases) ; – montrer que le nombre important de rétroactions nécessite de les traiter en se donnant des priorités, – comme les rétroactions de *Pensum* comportent des indications factuelles plutôt que des conseils explicites (conseils qui sont préférés par les utilisateurs, mais sont très difficiles à produire), il conviendra de montrer que signaler une rupture de cohérence entre phrases, ce n'est pas indiquer comment la réduire. Pour reprendre un auteur [SADLER 98, p. 78], « on ne peut pas considérer qu'il suffise que les étudiants "reçoivent un feedback" pour qu'ils sachent quoi en faire ». Pour pouvoir évaluer précisément l'effet que l'usage du système peut avoir sur les pratiques et l'apprentissage des étudiants, il faudra, s'il s'avérait impossible de s'assurer qu'ils sachent toujours quoi faire des feed-back proposés, au moins leur laisser suffisamment de temps pour qu'ils acquièrent une compréhension du système qui les satisfasse assez pour stabiliser un minimum leurs usages.

Pour conclure, les fonctionnalités mises en œuvre dans *Pensum* promeuvent l'apprentissage autorégulé, qui est au cœur de la plupart des problématiques sur l'apprentissage de l'écrit, mais également celles sur l'apprentissage tout au long de la vie récemment préconisé dans la littérature sur l'enseignement à distance [LIN 01], même si, paradoxalement, il n'existe que peu de modèles et de systèmes qui s'en réclament [VOVIDES et al. 07].

Remerciements

Ce travail a été réalisé dans le cadre du projet européen LTfLL (*Language Technologies for Life-Long Learning*, 7e PCRD, ICT-STREP). La MSH-Alpes a également contribué aux bonnes conditions matérielles de ce travail. Nous remercions Stéphane Simonian d'avoir bien voulu nous communiquer le cours sur les technologies éducatives du CNED, dont une partie a servi de support à cette étude.

7. Bibliographie

[ANTONIONADIS 04] Antoniadis, G., « Les logiciels d'apprentissage des langues peuvent-ils ignorer le TAL ? », *Les cahiers de l'APLIUT,* vol. 23, n° 2, 2004, p. 82–97.

[BAX 03] Bax, S., « CALL–Past, present and future », *System,* vol. 31, n° 1, 2003, p. 13–28.

[BUTLER & WINNE 95] Butler, D. L., Winne, P. H., « Feedback and self-regulated learning: A theoretical synthesis », *Review of Educational Research,* vol. 65, n° 3, 1995, p. 245–281.

[HÜBNER et al. 06] Hübner, S., Nückles, M., Renkl, A., « Fostering the cycle of self-regulation in writing learning protocols », In G. Clarebout, J. Elen (Eds.), *Avoiding simplicity, confronting complexity. Advances in studying and designing (computer-based) powerful learning environments*, Sense Publishers, Rotterdam, 2006, p. 155–164.

[KINTSCH et al. 07] Kintsch, E., Caccamise, D., Franzke, M., Johnson, N., Dooley, S., « Summary Street: Computer-guided summary writing », In T. K. Landauer, D. McNamara, S. Dennis, W. Kintsch (Eds.), *Handbook of Latent Semantic Analysis*, Erlbaum, Mahwah, 2007, p. 263–277.

[KLEIN 99] Klein, P. D., « Reopening inquiry into cognitive processes in writing-to-learn », *Educational Psychology Review,* vol. 11, n° 3, 1999, p. 203–270.

[LANDAUER & DUMAIS 97] Landauer, T. K., Dumais, S. T., « A solution to Plato's problem : the Latent Semantic Analysis theory of acquisition, induction and representation of knowledge », *Psychological Review,* vol. 104, n° 2, 1997, p. 211–240.

[LEBARBÉ 10] Lebarbé, T., *Fonctions interdisciplinaires intrinsèques et extrinsèques du traitement automatique des langues*, Habilitation à diriger des recherches en sciences du langage. Grenoble, Université Stendhal, 2010.

[LEMAIRE et al. 05] Lemaire, B., Mandin, S., Dessus, P., Denhière, G., « Computational cognitive models of summarization assessment skills », In B. G. Bara, L. Barsalou, M. Bucciarelli (Eds.), *Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci' 2005)*, Erlbaum, Mahwah, 2005, p. 1266–1271.

[LIN 01] Lin, X., « Designing metacognitive activities », *Educational Technology Research & Development,* vol. 49, n° 2, 2001, p. 23–40.

[MACARTHUR 06] MacArthur, C. A., « The effects of new technologies on writing and writing processes », In C. A. MacArthur, S. Graham, J. Fitzgerald (Eds.), *Handbook of writing research*, Guilford Press, New York, 2006, p. 248–262.

[MURRAY & BARNES 98] Murray, L., Barnes, A., « Beyond the "wow" factor – Evaluating multimedia language learning software from a pedagogical viewpoint », *System,* vol. 26, n° 2, 1998, p. 249–259.

[NICOL & MACFARLANE-DICK 06] Nicol, D. J., Macfarlane-Dick, D., « Formative assessment and self-regulated learning: a model and seven principles of good feedback practice », *Studies in Higher Education,* vol. 31, n° 2, 2006, p. 199–218.

[SADLER 98] Sadler, D. R., « Formative assessment: revisiting the territory », *Assessment in Education: Principles, Policy & Practice,* vol. 5, n° 1, 1998, p. 77–84.

[SEGEV-MILLER 04] Segev-Miller, R., « Writing from sources: the effect of explicit instruction on college students' processes and products », *L1-Educational Studies in Language and Literature,* vol. 4, n° 1, 2004, p. 5–33.

[TAPIERO 07] Tapiero, I., *Situation models and levels of coherence*, Mahwah, Erlbaum, 2007.

[TROIA 07] Troia, G. A., « Research in writing instruction. What we know and what we need to know », In M. Pressley, A. K. Billman, K. H. Perry, K. E. Reffitt, J. Moorhead Reynolds (Eds.), *Shaping literacy achievement. Research we have, research we need*, Guilford, New York, 2007, p. 129–156.

[VALENTI et al. 03] Valenti, S., Neri, F., Cucchiarelli, A., « An overview of current research on automated essay grading », *Journal of Information Technology Education,* vol. 2, 2003, p. 319–330.

[VILLIOT-LECLERCQ et al. 10] Villiot-Leclercq, E., Mandin, S., Dessus, P., Zampa, V., « Helping students understand courses through written syntheses. An LSA-based online advisor », *Tenth IEEE Int. Conf. on Advanced Learning Technologies (ICALT 2010)*, Sousse (Tunisie), 2010.

[VOVIDES et al. 07] Vovides, Y., Sanchez-Alonso, S., Mitropoulou, V., Nickmans, G., « The use of e-learning course management systems to support learning strategies and to improve self-regulated learning », *Educational Research Review,* vol. 2, n° 1, 2007, p. 64–74.

[WILD 07] Wild, F., « An LSA package for R »*, Mini-Proceedings of the 1st European Workshop on Latent Semantic Analysis in Technology-Enhanced Learning*, Heerlen, Pays-Bas, 2007.

[ZIMMERMAN 00] Zimmerman, B. J., « Attaining self-regulation : A social cognitive perspective », In M. Boekaerts, P. R. Pintrich, M. Zeidner (Eds.), *Handbook of self-regulation*, Academic Press, San Diego, 2000, p. 13–41.